AXIOMTEK

CEM100

AMD G-Series APU
COM Express™ Type 2
Basic Module

User's Manual

Disclaimers

This manual has been carefully checked and believed to contain accurate information. Axiomtek Co., Ltd. assumes no responsibility for any infringements of patents or any third party's rights, and any liability arising from such use.

Axiomtek does not warrant or assume any legal liability or responsibility for the accuracy, completeness or usefulness of any information in this document. Axiomtek does not make any commitment to update the information in this manual.

Axiomtek reserves the right to change or revise this document and/or product at any time without notice.

No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Axiomtek Co., Ltd.

CAUTION

If you replace wrong batteries, it causes the danger of explosion. It is recommended by the manufacturer that you follow the manufacturer's instructions to only replace the same or equivalent type of battery, and dispose of used ones.

April 2012, Version A1

Printed in Taiwan

ESD Precautions

Computer boards have integrated circuits sensitive to static electricity. To prevent chipsets from electrostatic discharge damage, please take care of the following jobs with precautions:

- Do not remove boards or integrated circuits from their anti-static packaging until you are ready to install them.
- Before holding the board or integrated circuit, touch an unpainted portion of the system unit chassis for a few seconds. It discharges static electricity from your body.
- Wear a wrist-grounding strap, available from most electronic component stores, when handling boards and components.

Trademarks Acknowledgments

Table of Contents

This page is intentionally left blank.

Chapter 1
Introduction

The CEM100 is a new COM Express™ Type 2 Basic Module to support BGA type AMD G-Series processors. It integrates AMD FCH A55E chipset which supports the most updated high speed I/O, like PCI-Express gen 2 at 5GT/s and SATA-600 at 6Gb/s. The CEM100 does fully comply with COM Express™ Type 2 specification. It provides 8 lanes of PCIe, gigabit ethernet, HD audio interface, VGA/LVDS LCD and 2 configurable digital display ports for more flexible digital display options.

1.1 Features

- AMD G-series BGA processors
  Product Configurations:

CEM100DG-T56N	TPM
CEM100DG-T40R	TPM

- AMD FCH A55E
- 2 DDR3 1066MHz SO-DIMMs support up to 8GB memory capacity
- 8 lanes of PCIe gen 2
- 4 SATA-600 with RAID 0, 1, 5, 10
- 8 USB 2.0 ports
- TPM

1.2 Specifications

- **CPU**
  - AMD G-Series BGA processors.

- **Chipset**
  - AMD FCH A55E chipset.

- **BIOS**
  - American Megatrends Inc. BIOS.
  - 16Mbit SPI Flash, DMI, Plug and Play.
  - RPL/PXE Ethernet Boot ROM, customized default saving features, LPC-free supported, uses SPI type flash memory.

- **System Memory**
  - Two 204-pin DDR3 SO-DIMM slots.
  - Maximum up to 8GB DDR3 1066MHz memory.

- **TPM**
  - Trusted Platform Module compatible with TPM1.2 Main and PC Client specification based on AMD LPC Bus Interface.

- **Expansion Interface**
  - Four PCI-Express x1 (gen 2) thru APU for discrete graphics or general purpose PCI Express (1 x4 or 4 x1).
  - Four PCI-Express x1 (gen 2) thru FCH general purpose PCI Express.

- **USB Interface**
  - Eight USB ports comply with USB Spec. Rev. 2.0.

- **SATA Interface**
  - Four SATA 6Gb/s ports supported through COM Express™ connector.

***Note**: Users upgrading a computer system with a larger hard disk drive may have problems using more than 2.2 terabytes (TB) of disk space. This size limit is based on limits introduced by the master boot record (MBR) partitioning method.*
Users can install the OS on a MBR formatted disk under 2.2TB. The OS will use the MBR disk as the boot drive, eliminating the MBR limitation.

- **Video**
  - ATi Radeon HD6310/HD6250 integrated in APU T56N/T40R.
  - CRT interface supports up to QXGA (2048 x 1536).
  - 18/24-bit single/dual channel LVDS interface.
  - Two digital display ports supporting HDMI / DVI / DisplayPort.

- **Ethernet**
  - One 10/100/1000 Base-T provided by Broadcom BCM50610 PHY with integrated boot ROM.

- **Audio**
  - HD link interface to baseboard for external codec.

- **Power Management**
  - ACPI (Advanced Configuration and Power Interface).

- **Form Factor**
  - Basic module 125mm x 95mm.

1.3 Utilities Supported

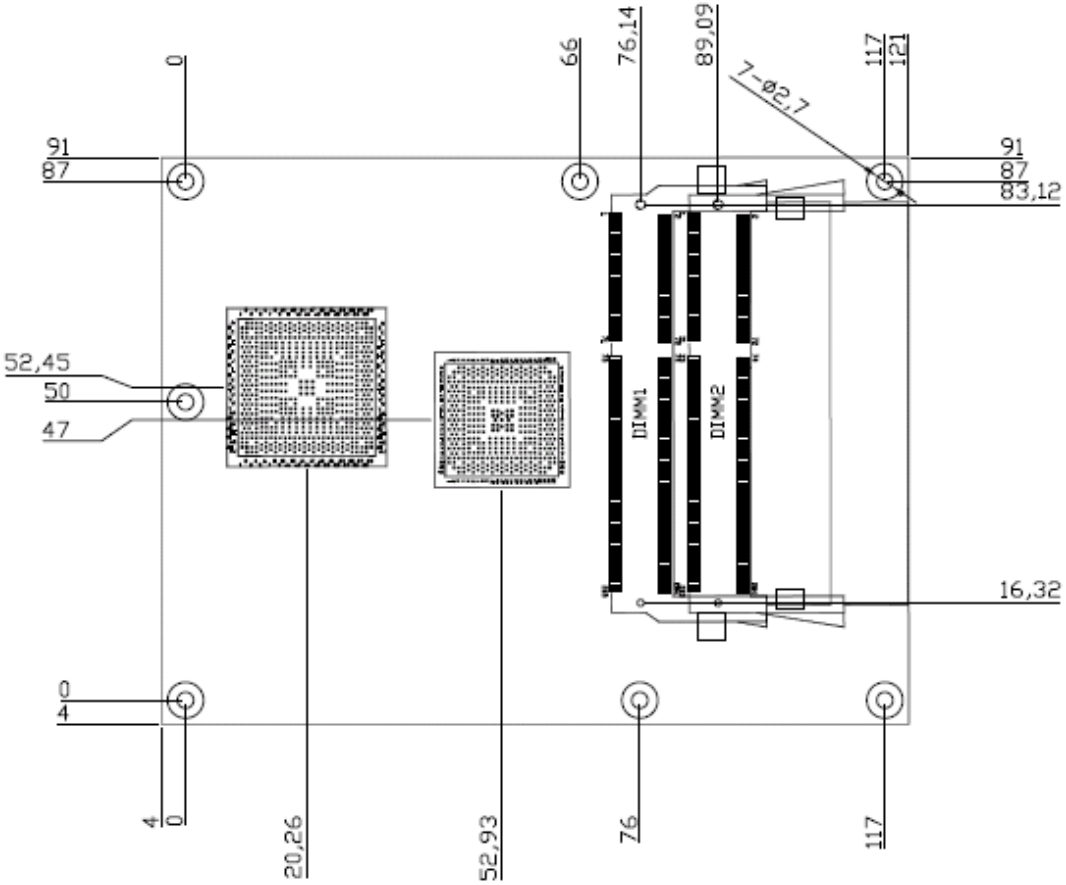
- AMD A55E utility
- Chipset and Graphic driver
- Ethernet driver
- TPM utility
- Audio driver

Note: ***All specifications and images are subject to change without notice.***

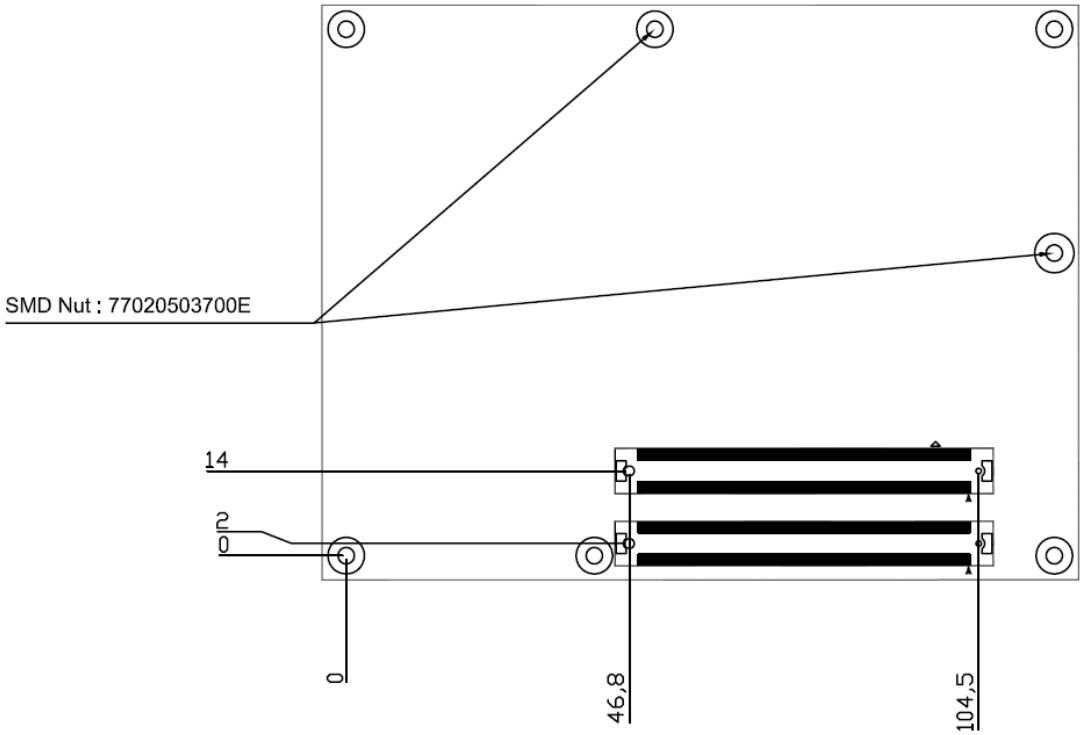
This page is intentionally left blank.

Chapter 2
Module and Pin Assignments

2.1 Module Dimensions and Fixing Holes

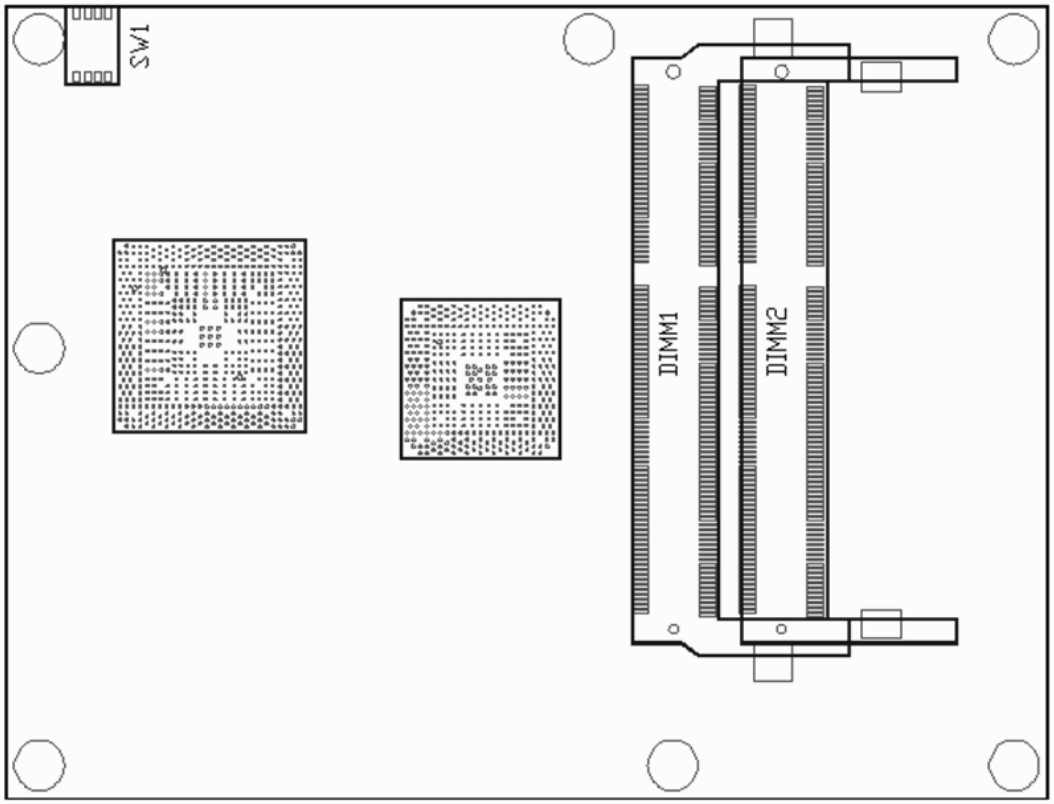


Top Side

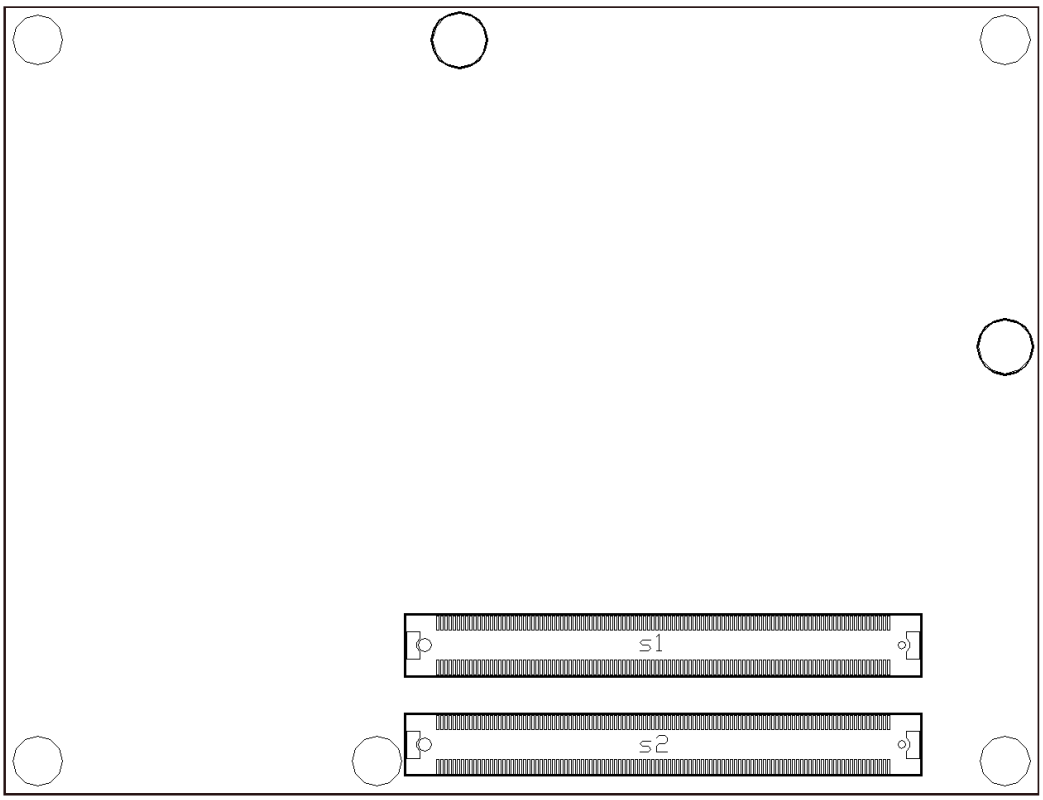


Bottom Side

2.2 Module Layout



Top Side



Bottom Side

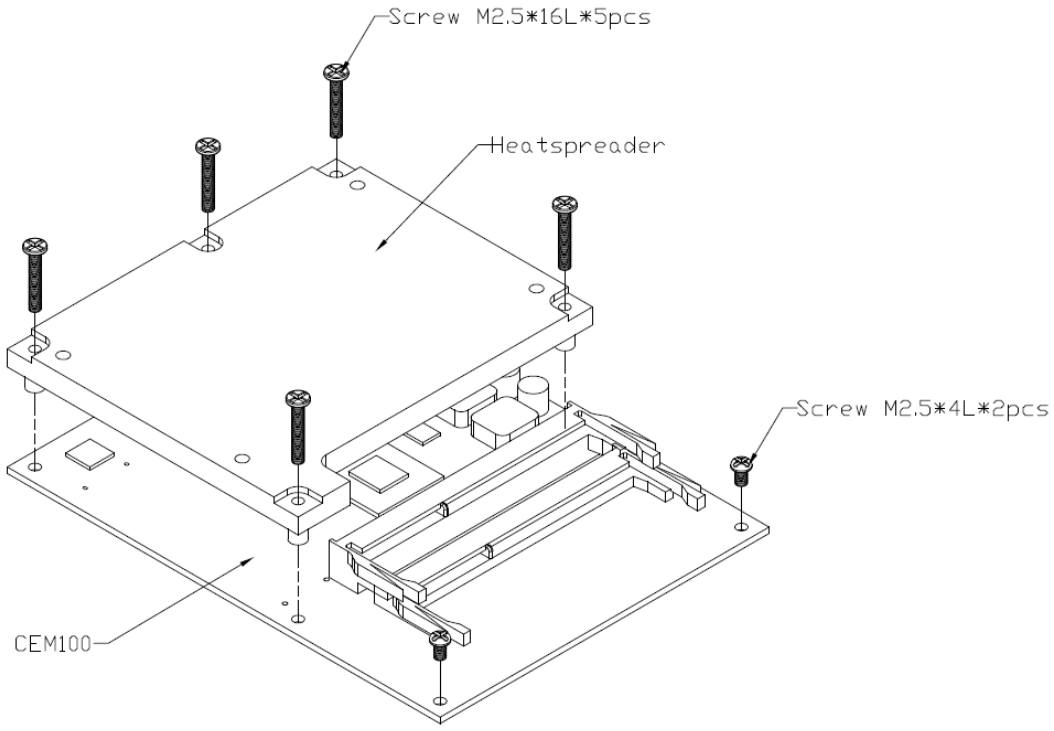
2.3 Installing Heat spreader

For thermal dissipation, a heat spreader enables the CEM100's components to dissipate heat efficiently. All heat generating components are thermally conducted to the heat spreader in order to avoid hot spots. Follow steps below to install heat spreader.

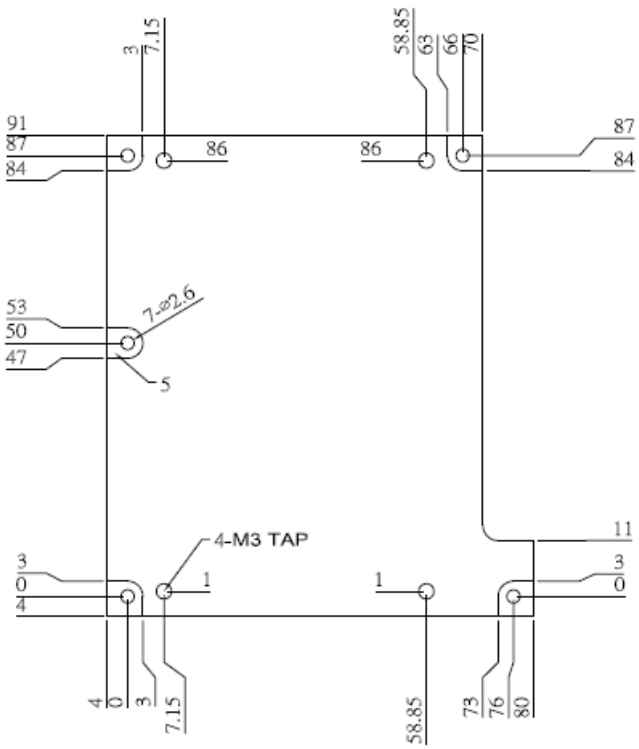
1. There is a protective plastic covering on the thermal pads. This must be removed before the heat spreader can be mounted.

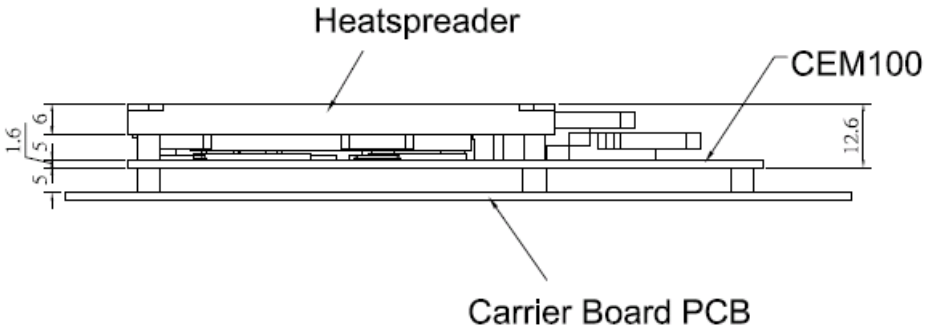
2. Each heat spreader is designed for a specific CEM module. The thermal pads on the heat spreader are designed to make contact with the necessary components on the CEM module. When mounting the heat spreader you must make sure that the thermal pads on the heat spreader make complete contact (no space between thermal pad and component) with the corresponding components on the CEM module. This is especially critical for CEM modules that have higher CPU speeds (for example 1.0GHz or more) to ensure that the heat spreader acts as a proper thermal interface for cooling solutions.

3. This CPU module has seven assembly holes for installing heat spreader plate. Align and use screws (M2.5x16Lx5pcs and M2.5x4Lx2pcs) to secure the heat spreader plate to the CEM100. Be careful not to over-tighten the screws.



4. If you need additional information for thermal solution, please refer to image below to design your system's cooling.





2.4 Switch Setting (SW1)

If SW1-1 is enabled for AC power input, the system will be automatically power on without pressing soft power button. If SW1-1 is disabled for AC power input, it is necessary to manually press soft power button to turn on the system. This function is similar to the feature of power on after power failure, which is controlled by hardware circuitry instead of BIOS.

Properly configure switch setting on the CEM100 to meet your application purpose. See table below for detailed information.

Function	Setting
Enable auto power on	SW1-1 ON
Disable auto power on (Default)	SW1-1 OFF
Display port 0 is Display Port	SW1-2 ON
Display port 0 is LVDS (Default)	SW1-2 OFF
Enable IDE function (Default)	SW1-3 ON
Disable IDE function	SW1-3 OFF
DDR3 memory voltage is 1.35V	SW1-4 ON
DDR3 memory voltage is 1.5V (Default)	SW1-4 OFF

SW1

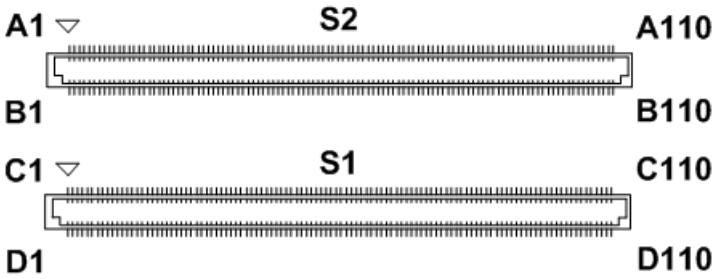
2.5 Connectors

Signals go to the other parts of the system through connectors. Loose or improper connection might cause problems, please make sure all connectors are properly and firmly connected. Here is a summary table which shows all connectors on the hardware.

Connector	Description
DIMM1	DDR3 SO-DIMM Connector
DIMM2	DDR3 SO-DIMM Connector
S1	COM Express™ Connector
S2	COM Express™ Connector

2.5.1 COM Express™ Connectors (S1 and S2)

The S1 and S2 are for connecting COM Express™ module and COM Express™ baseboard. The following table shows pin assignments for the 220-pin COM Express™ connectors.



Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
A1	GND (FIXED)	B1	GND (FIXED)	C1	GND (FIXED)	D1	GND (FIXED)
A2	GBE0_MDI3-	B2	GBE0_ACT#	C2	IDE_D7	D2	IDE_D5
A3	GBE0_MDI3+	B3	LPC_FRAME#	C3	IDE_D6	D3	IDE_D10
A4	GBE0_LINK100#	B4	LPC_AD0	C4	IDE_D3	D4	IDE_D11
A5	GBE0_LINK1000#	B5	LPC_AD1	C5	IDE_D15	D5	IDE_D12
A6	GBE0_MDI2-	B6	LPC_AD2	C6	IDE_D8	D6	IDE_D4
A7	GBE0_MDI2+	B7	LPC_AD3	C7	IDE_D9	D7	IDE_D0
A8	GBE0_LINK#	B8	LPC_DRQ0#	C8	IDE_D2	D8	IDE_REQ
A9	GBE0_MDI1-	B9	LPC_DRQ1#	C9	IDE_D13	D9	IDE_IOW#
A10	GBE0_MDI1+	B10	LPC_CLK	C10	IDE_D1	D10	IDE_ACK#
A11	GND (FIXED)	B11	GND (FIXED)	C11	GND (FIXED)	D11	GND (FIXED)
A12	GBE0_MDI0-	B12	PWRBTN#	C12	IDE_D14	D12	IDE_IRQ
A13	GBE0_MDI0+	B13	SMB_CK	C13	IDE_IORDY	D13	IDE_A0
A14	GBE0_CTREF	B14	SMB_DAT	C14	IDE_IOR#	D14	IDE_A1
A15	SUS_S3#	B15	SMB_ALERT#	C15	PCI_PME#	D15	IDE_A2
A16	SATA0_TX+	B16	SATA1_TX+	C16	PCI_GNT2#	D16	IDE_CS1#
A17	SATA0_TX-	B17	SATA1_TX-	C17	PCI_REQ2#	D17	IDE_CS3#
A18	SUS_S4#	B18	SUS_STAT#	C18	PCI_GNT1#	D18	IDE_RESET#
A19	SATA0_RX+	B19	SATA1_RX+	C19	PCI_REQ1#	D19	PCI_GNT3#
A20	SATA0_RX-	B20	SATA1_RX-	C20	PCI_GNT0#	D20	PCI_REQ3#
A21	GND (FIXED)	B21	GND (FIXED)	C21	GND (FIXED)	D21	GND (FIXED)
A22	N.C	B22	N.C	C22	PCI_REQ0#	D22	PCI_AD1
A23	N.C	B23	N.C	C23	PCI_RESET#	D23	PCI_AD3
A24	SUS_S5#	B24	PWR_OK	C24	PCI_AD0	D24	PCI_AD5
A25	N.C	B25	N.C	C25	PCI_AD2	D25	PCI_AD7
A26	N.C	B26	N.C	C26	PCI_AD4	D26	PCI_C/BE0#
A27	BATLOW#	B27	WDT	C27	PCI_AD6	D27	PCI_AD9
A28	ATA_ACT#	B28	AC_SDIN2	C28	PCI_AD8	D28	PCI_AD11
A29	AC_SYNC	B29	AC_SDIN1	C29	PCI_AD10	D29	PCI_AD13
A30	AC_RST#	B30	AC_SDIN0	C30	PCI_AD12	D30	PCI_AD15
A31	GND (FIXED)	B31	GND (FIXED)	C31	GND (FIXED)	D31	GND (FIXED)
A32	AC_BITCLK	B32	SPKR	C32	PCI_AD14	D32	PCI_PAR
A33	AC_SDOUT	B33	I2C_CK	C33	PCI_C/BE1#	D33	PCI_SERR#
A34	N.C	B34	I2C_DAT	C34	PCI_PERR#	D34	PCI_STOP#
A35	THRMTRIP#	B35	THRM#	C35	PCI_LOCK#	D35	PCI_TRDY#
A36	USB6-	B36	USB7-	C36	PCI_DEVSEL#	D36	PCI_FRAME#
A37	USB6+	B37	USB7+	C37	PCI_IRDY#	D37	PCI_AD16
A38	USB_6_7_OC#	B38	USB_4_5_OC#	C38	PCI_C/BE2#	D38	PCI_AD18
A39	USB4-	B39	USB5-	C39	PCI_AD17	D39	PCI_AD20
A40	USB4+	B40	USB5+	C40	PCI_AD19	D40	PCI_AD22
A41	GND (FIXED)	B41	GND (FIXED)	C41	GND (FIXED)	D41	GND (FIXED)
A42	USB2-	B42	USB3-	C42	PCI_AD21	D42	PCI_AD24
A43	USB2+	B43	USB3+	C43	PCI_AD23	D43	PCI_AD26
A44	USB_2_3_OC#	B44	USB_0_1_OC#	C44	PCI_C/BE3#	D44	PCI_AD28
A45	USB0-	B45	USB1-	C45	PCI_AD25	D45	PCI_AD30
A46	USB0+	B46	USB1+	C46	PCI_AD27	D46	PCI_IRQC#
A47	VCC_RTC	B47	EXCD1_PERST#	C47	PCI_AD29	D47	PCI_IRQD#
A48	EXCD0_PERST#	B48	EXCD1_CPPE#	C48	PCI_AD31	D48	PCI_CLKRUN#
A49	EXCD0_CPPE#	B49	SYS_RESET#	C49	PCI_IRQA#	D49	N.C
A50	LPC_SERIRQ	B50	CB_RESET#	C50	PCI_IRQB#	D50	PCI_CLK
A51	GND (FIXED)	B51	GND (FIXED)	C51	GND (FIXED)	D51	GND (FIXED)
A52	*PCIE_TX0+	B52	*PCIE_RX0+	C52	PEG_RX0+	D52	PEG_TX0+
A53	*PCIE_TX0-	B53	*PCIE_RX0-	C53	PEG_RX0-	D53	PEG_TX0-
A54	GPI0	B54	GPO1	C54	N.C	D54	PEG_LANE_RV#
A55	N.C	B55	N.C	C55	PEG_RX1+	D55	PEG_TX1+

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
A56	N.C	B56	N.C	C56	PEG_RX1-	D56	PEG_TX1-
A57	GND	B57	GPO2	C57	N.C	D57	N.C
A58	N.C	B58	N.C	C58	PEG_RX2+	D58	PEG_TX2+
A59	N.C	B59	N.C	C59	PEG_RX2-	D59	PEG_TX2-
A60	GND (FIXED)	B60	GND (FIXED)	C60	GND (FIXED)	D60	GND (FIXED)
A61	PCIE_TX3+	B61	PCIE_RX3+	C61	PEG_RX3+	D61	PEG_TX3+
A62	PCIE_TX3-	B62	PCIE_RX3-	C62	PEG_RX3-	D62	PEG_TX3-
A63	GPI1	B63	GPO3	C63	RSVD	D63	RSVD
A64	PCIE_TX2+	B64	PCIE_RX2+	C64	RSVD	D64	RSVD
A65	PCIE_TX2-	B65	PCIE_RX2-	C65	PEG_RX4+	D65	PEG_TX4+
A66	GND	B66	WAKE0#	C66	PEG_RE4-	D66	PEG_TX4-
A67	GPI2	B67	WAKE1#	C67	RSVD	D67	GND
A68	PCIE_TX1+	B68	PCIE_RX1+	C68	PEG_RX5+	D68	PEG_TX5+
A69	PCIE_TX1-	B69	PCIE_RX1-	C69	PEG_RX5-	D69	PEG_TX5-
A70	GND(FIXED)	B70	GND(FIXED)	C70	GND(FIXED)	D70	GND(FIXED)
A71	LVDS_A0+	B71	LVDS_B0+	C71	PEG_RX6+	D71	PEG_TX6+
A72	LVDS_A0-	B72	LVDS_B0-	C72	PEG_RX6-	D72	PEG_TX6-
A73	LVDS_A1+	B73	LVDS_B1+	C73	SDVO_DATA	D73	SDVO_CLK
A74	LVDS_A1-	B74	LVDS_B1-	C74	PEG_RX7+	D74	PEG_TX7+
A75	LVDS_A2+	B75	LVDS_B2+	C75	PEG_RX7-	D75	PEG_TX7-
A76	LVDS_A2-	B76	LVDS_B2-	C76	GND	D76	GND
A77	LVDS_VDD_EN	B77	**LVDS_B3+	C77	RSVD	D77	IDE_CBLID#
A78	**LVDS_A3+	B78	**LVDS_B3-	C78	PEG_RX8+	D78	PEG_TX8+
A79	**LVDS_A3-	B79	LVDS_BKLT_EN	C79	PEG_RX8-	D79	PEG_TX8-
A80	GND(FIXED)	B80	GND(FIXED)	C80	GND(FIXED)	D80	GND(FIXED)
A81	LVDS_A_CK+	B81	LVDS_B_CK+	C81	PEG_RX9+	D81	PEG_TX9+
A82	LVDS_A_CK-	B82	LVDS_B_CK-	C82	PEG_RX9-	D82	PEG_TX9-
A83	LVDS_I2C_CK	B83	LVDS_BKLT_CTRL	C83	RSVD	D83	RSVD
A84	LVDS_I2C_DAT	B84	VCC_5V_SBY	C84	GND	D84	GND
A85	GPI3	B85	VCC_5V_SBY	C85	PEG_RX10+	D85	PEG_TX10+
A86	KBD_RST#	B86	VCC_5V_SBY	C86	PEG_RX10-	D86	PEG_TX10-
A87	KBD_A20GATE	B87	VCC_5V_SBY	C87	GND	D87	GND
A88	PCIE0_CK_REF+	B88	RSVD	C88	PEG_RX11+	D88	PEG_TX11+
A89	PCIE0_CK_REF-	B89	VGA_RED	C89	PEG_RX11-	D89	PEG_TX11-
A90	GND (FIXED)	B90	GND (FIXED)	C90	GND (FIXED)	D90	GND (FIXED)
A91	RSVD	B91	VGA_GRN	C91	PEG_RX12+	D91	PEG_TX12+
A92	RSVD	B92	VGA_BLU	C92	PEG_RX12-	D92	PEG_TX12-
A93	GPO0	B93	VGA_HSYNC	C93	GND	D93	GND
A94	RSVD	B94	VGA_VSYNC	C94	PEG_RX13+	D94	PEG_TX13+
A95	RSVD	B95	VGA_I2C_CK	C95	PEG_RX13-	D95	PEG_TX13-
A96	GND	B96	VGA_I2C_DAT	C96	GND	D96	GND
A97	VCC_12V	B97	TV_DAC_A	C97	RSVD	D97	N.C
A98	VCC_12V	B98	TV_DAC_B	C98	PEG_RX14+	D98	PEG_TX14+
A99	VCC_12V	B99	TV_DAC_C	C99	PEG_RX14-	D99	PEG_TX14-
A100	GND (FIXED)	B100	GND (FIXED)	C100	GND (FIXED)	D100	GND (FIXED)
A101	VCC_12V	B101	VCC_12V	C101	PEG_RX15+	D101	PEG_TX15+
A102	VCC_12V	B102	VCC_12V	C102	PEG_RX15-	D102	PEG_TX15-
A103	VCC_12V	B103	VCC_12V	C103	GND	D103	GND
A104	VCC_12V	B104	VCC_12V	C104	VCC_12V	D104	VCC_12V
A105	VCC_12V	B105	VCC_12V	C105	VCC_12V	D105	VCC_12V
A106	VCC_12V	B106	VCC_12V	C106	VCC_12V	D106	VCC_12V
A107	VCC_12V	B107	VCC_12V	C107	VCC_12V	D107	VCC_12V
A108	VCC_12V	B108	VCC_12V	C108	VCC_12V	D108	VCC_12V
A109	VCC_12V	B109	VCC_12V	C109	VCC_12V	D109	VCC_12V
A110	GND (FIXED)	B110	GND (FIXED)	C110	GND (FIXED)	D110	GND (FIXED)

Chapter 3
Hardware Description

3.1 Microprocessor

The CEM100 supports BGA type AMD G-Series processors, which enables your system to operate under Windows® 7 and Linux environments. The system performance depends on the microprocessor. You must install the heat spreader or cooler carefully and properly to prevent damage.

3.2 BIOS

The CEM100 uses AMI Plug and Play BIOS with a single 16Mbit SPI Flash.

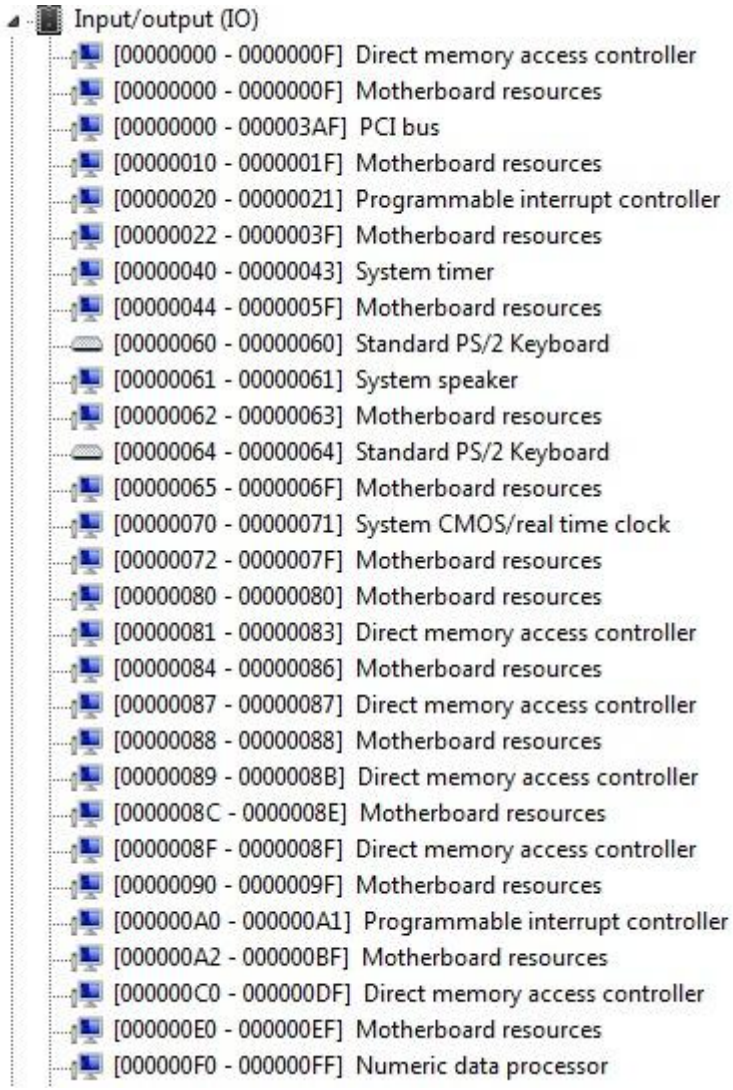
3.3 System Memory

The CEM100 supports two 204-pin DDR3 1066MHz SO-DIMM slots for a maximum memory of 8GB DDR3 SDRAMs. The memory module can come in sizes of 1GB, 2GB and 4GB.

3.4 I/O Port Address Map

The BGA type AMD G-Series processors communicate via I/O ports. Total 1KB port addresses are available for assigning to other devices via I/O expansion cards.

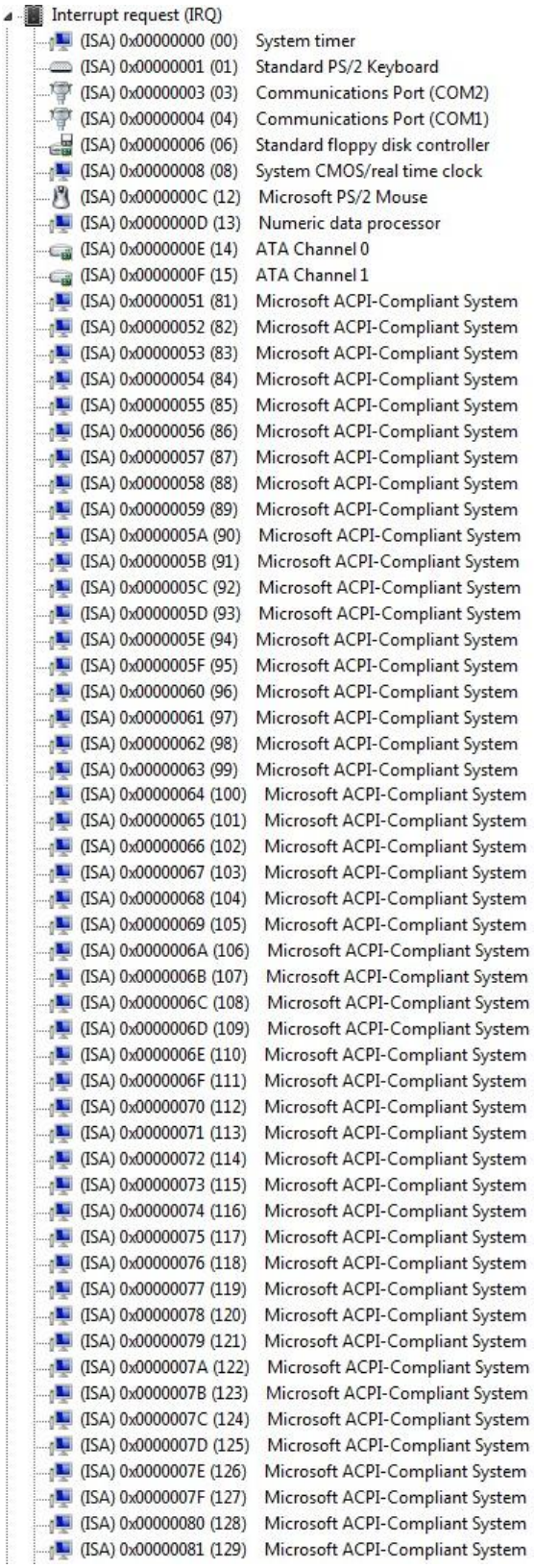
The I/O port addresses (with CEB94000 baseboard under Windows® 7) are as follows:



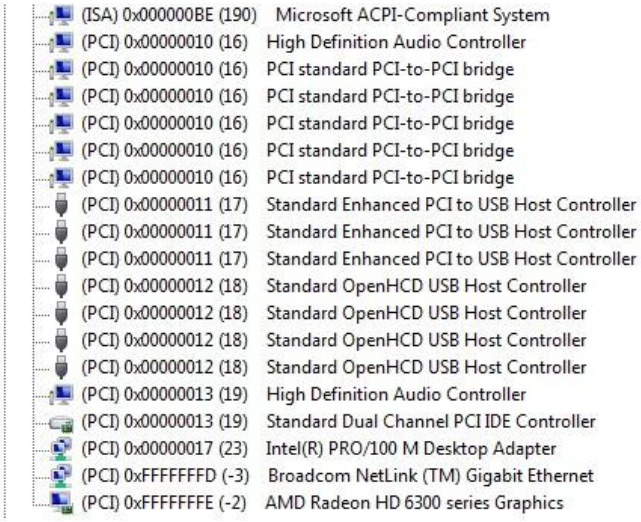
- [00000170 - 00000177] ATA Channel 1
- [000001F0 - 000001F7] ATA Channel 0
- [00000280 - 0000028F] Motherboard resources
- [000002F8 - 000002FF] Communications Port (COM2)
- [00000376 - 00000376] ATA Channel 1
- [00000378 - 0000037F] Printer Port (LPT1)
- [000003B0 - 000003BB] AMD Radeon HD 6300 series Graphics
- [000003B0 - 000003DF] PCI bus
- [000003C0 - 000003DF] AMD Radeon HD 6300 series Graphics
- [000003E0 - 00000CF7] PCI bus
- [000003F0 - 000003F5] Standard floppy disk controller
- [000003F6 - 000003F6] ATA Channel 0
- [000003F7 - 000003F7] Standard floppy disk controller
- [000003F8 - 000003FF] Communications Port (COM1)
- [0000040B - 0000040B] Motherboard resources
- [000004D0 - 000004D1] Motherboard resources
- [000004D6 - 000004D6] Motherboard resources
- [00000800 - 0000089F] Motherboard resources
- [00000900 - 0000090F] Motherboard resources
- [00000910 - 0000091F] Motherboard resources
- [00000B20 - 00000B3F] Motherboard resources
- [00000C00 - 00000C01] Motherboard resources
- [00000C14 - 00000C14] Motherboard resources
- [00000C50 - 00000C51] Motherboard resources
- [00000C52 - 00000C52] Motherboard resources
- [00000C6C - 00000C6C] Motherboard resources
- [00000C6F - 00000C6F] Motherboard resources
- [00000CD0 - 00000CD1] Motherboard resources
- [00000CD2 - 00000CD3] Motherboard resources
- [00000CD4 - 00000CD5] Motherboard resources
- [00000CD6 - 00000CD7] Motherboard resources
- [00000CD8 - 00000CDF] Motherboard resources
- [00000D00 - 0000FFFF] PCI bus
- [0000D000 - 0000DFFF] PCI standard PCI-to-PCI bridge
- [0000E000 - 0000E03F] Intel(R) PRO/100 M Desktop Adapter
- [0000E000 - 0000EFFF] ATI I/O Communications Processor PCI Bus Controller
- [0000F000 - 0000F0FF] AMD Radeon HD 6300 series Graphics
- [0000F100 - 0000F10F] Standard Dual Channel PCI IDE Controller
- [0000F150 - 0000F15F] Standard Dual Channel PCI IDE Controller
- [0000F160 - 0000F163] Standard Dual Channel PCI IDE Controller
- [0000F170 - 0000F177] Standard Dual Channel PCI IDE Controller
- [0000F180 - 0000F183] Standard Dual Channel PCI IDE Controller
- [0000F190 - 0000F197] Standard Dual Channel PCI IDE Controller
- [0000FE00 - 0000FEFE] Motherboard resources

3.5 Interrupt Controller (IRQ) Map

The IRQ mapping list (with CEB94000 baseboard under Windows® 7) is shown as follows:

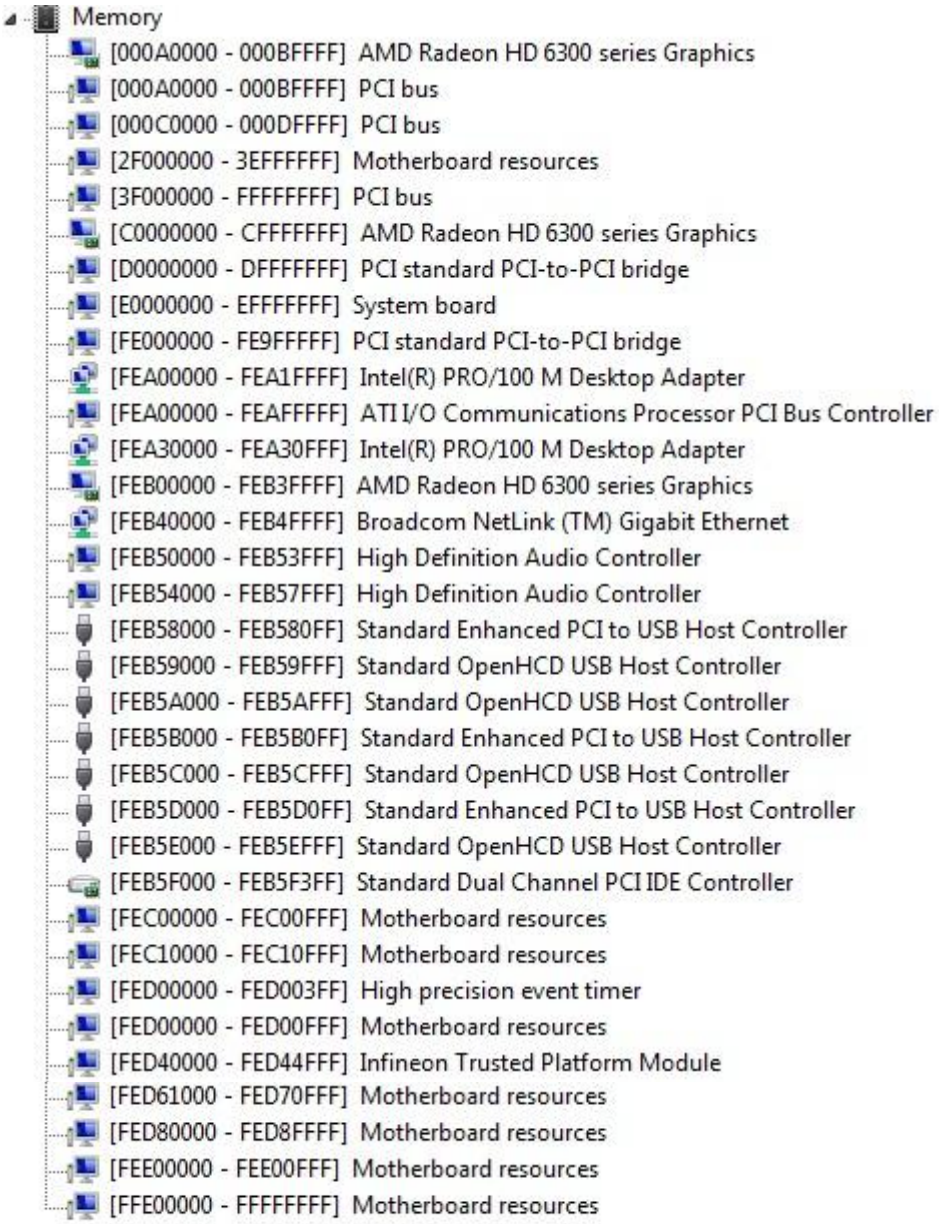


- (ISA) 0x00000082 (130) Microsoft ACPI-Compliant System
- (ISA) 0x00000083 (131) Microsoft ACPI-Compliant System
- (ISA) 0x00000084 (132) Microsoft ACPI-Compliant System
- (ISA) 0x00000085 (133) Microsoft ACPI-Compliant System
- (ISA) 0x00000086 (134) Microsoft ACPI-Compliant System
- (ISA) 0x00000087 (135) Microsoft ACPI-Compliant System
- (ISA) 0x00000088 (136) Microsoft ACPI-Compliant System
- (ISA) 0x00000089 (137) Microsoft ACPI-Compliant System
- (ISA) 0x0000008A (138) Microsoft ACPI-Compliant System
- (ISA) 0x0000008B (139) Microsoft ACPI-Compliant System
- (ISA) 0x0000008C (140) Microsoft ACPI-Compliant System
- (ISA) 0x0000008D (141) Microsoft ACPI-Compliant System
- (ISA) 0x0000008E (142) Microsoft ACPI-Compliant System
- (ISA) 0x0000008F (143) Microsoft ACPI-Compliant System
- (ISA) 0x00000090 (144) Microsoft ACPI-Compliant System
- (ISA) 0x00000091 (145) Microsoft ACPI-Compliant System
- (ISA) 0x00000092 (146) Microsoft ACPI-Compliant System
- (ISA) 0x00000093 (147) Microsoft ACPI-Compliant System
- (ISA) 0x00000094 (148) Microsoft ACPI-Compliant System
- (ISA) 0x00000095 (149) Microsoft ACPI-Compliant System
- (ISA) 0x00000096 (150) Microsoft ACPI-Compliant System
- (ISA) 0x00000097 (151) Microsoft ACPI-Compliant System
- (ISA) 0x00000098 (152) Microsoft ACPI-Compliant System
- (ISA) 0x00000099 (153) Microsoft ACPI-Compliant System
- (ISA) 0x0000009A (154) Microsoft ACPI-Compliant System
- (ISA) 0x0000009B (155) Microsoft ACPI-Compliant System
- (ISA) 0x0000009C (156) Microsoft ACPI-Compliant System
- (ISA) 0x0000009D (157) Microsoft ACPI-Compliant System
- (ISA) 0x0000009E (158) Microsoft ACPI-Compliant System
- (ISA) 0x0000009F (159) Microsoft ACPI-Compliant System
- (ISA) 0x000000A0 (160) Microsoft ACPI-Compliant System
- (ISA) 0x000000A1 (161) Microsoft ACPI-Compliant System
- (ISA) 0x000000A2 (162) Microsoft ACPI-Compliant System
- (ISA) 0x000000A3 (163) Microsoft ACPI-Compliant System
- (ISA) 0x000000A4 (164) Microsoft ACPI-Compliant System
- (ISA) 0x000000A5 (165) Microsoft ACPI-Compliant System
- (ISA) 0x000000A6 (166) Microsoft ACPI-Compliant System
- (ISA) 0x000000A7 (167) Microsoft ACPI-Compliant System
- (ISA) 0x000000A8 (168) Microsoft ACPI-Compliant System
- (ISA) 0x000000A9 (169) Microsoft ACPI-Compliant System
- (ISA) 0x000000AA (170) Microsoft ACPI-Compliant System
- (ISA) 0x000000AB (171) Microsoft ACPI-Compliant System
- (ISA) 0x000000AC (172) Microsoft ACPI-Compliant System
- (ISA) 0x000000AD (173) Microsoft ACPI-Compliant System
- (ISA) 0x000000AE (174) Microsoft ACPI-Compliant System
- (ISA) 0x000000AF (175) Microsoft ACPI-Compliant System
- (ISA) 0x000000B0 (176) Microsoft ACPI-Compliant System
- (ISA) 0x000000B1 (177) Microsoft ACPI-Compliant System
- (ISA) 0x000000B2 (178) Microsoft ACPI-Compliant System
- (ISA) 0x000000B3 (179) Microsoft ACPI-Compliant System
- (ISA) 0x000000B4 (180) Microsoft ACPI-Compliant System
- (ISA) 0x000000B5 (181) Microsoft ACPI-Compliant System
- (ISA) 0x000000B6 (182) Microsoft ACPI-Compliant System
- (ISA) 0x000000B7 (183) Microsoft ACPI-Compliant System
- (ISA) 0x000000B8 (184) Microsoft ACPI-Compliant System
- (ISA) 0x000000B9 (185) Microsoft ACPI-Compliant System
- (ISA) 0x000000BA (186) Microsoft ACPI-Compliant System
- (ISA) 0x000000BB (187) Microsoft ACPI-Compliant System
- (ISA) 0x000000BC (188) Microsoft ACPI-Compliant System
- (ISA) 0x000000BD (189) Microsoft ACPI-Compliant System


(ISA) 0x000000BE (190) Microsoft ACPI-Compliant System
(PCI) 0x00000010 (16) High Definition Audio Controller
(PCI) 0x00000010 (16) PCI standard PCI-to-PCI bridge
(PCI) 0x00000010 (16) PCI standard PCI-to-PCI bridge
(PCI) 0x00000010 (16) PCI standard PCI-to-PCI bridge
(PCI) 0x00000010 (16) PCI standard PCI-to-PCI bridge
(PCI) 0x00000010 (16) PCI standard PCI-to-PCI bridge
(PCI) 0x00000011 (17) Standard Enhanced PCI to USB Host Controller
(PCI) 0x00000011 (17) Standard Enhanced PCI to USB Host Controller
(PCI) 0x00000011 (17) Standard Enhanced PCI to USB Host Controller
(PCI) 0x00000012 (18) Standard OpenHCD USB Host Controller
(PCI) 0x00000012 (18) Standard OpenHCD USB Host Controller
(PCI) 0x00000012 (18) Standard OpenHCD USB Host Controller
(PCI) 0x00000012 (18) Standard OpenHCD USB Host Controller
(PCI) 0x00000013 (19) High Definition Audio Controller
(PCI) 0x00000013 (19) Standard Dual Channel PCI IDE Controller
(PCI) 0x00000017 (23) Intel(R) PRO/100 M Desktop Adapter
(PCI) 0xFFFFFFFD (-3) Broadcom NetLink (TM) Gigabit Ethernet
(PCI) 0xFFFFFFFE (-2) AMD Radeon HD 6300 series Graphics

3.6 Memory Map

The memory mapping list (with CEB94000 baseboard under Windows® 7) is shown as follows:



This page is intentionally left blank.

Chapter 4
AMI BIOS Setup Utility

The AMI BIOS provides users with a built-in setup program to modify basic system configuration. All configured parameters are stored in a battery-backed CMOS to save the setup information whenever the power is turned off. This chapter provides users with detailed description about how to set up basic system configuration through the AMI BIOS setup utility.

4.1 Starting

To enter the setup screens, follow the steps below:

1. Turn on the computer and press the <Del> key immediately.
2. After you press the <Del> key, the main BIOS setup menu displays. You can access the other setup screens from the main BIOS setup menu, such as the Advanced and Chipset menus.

It is strongly recommended that you should avoid changing the chipset's defaults. Both AMI and your system manufacturer have carefully set up these defaults that provide the best performance and reliability.

4.2 Navigation Keys

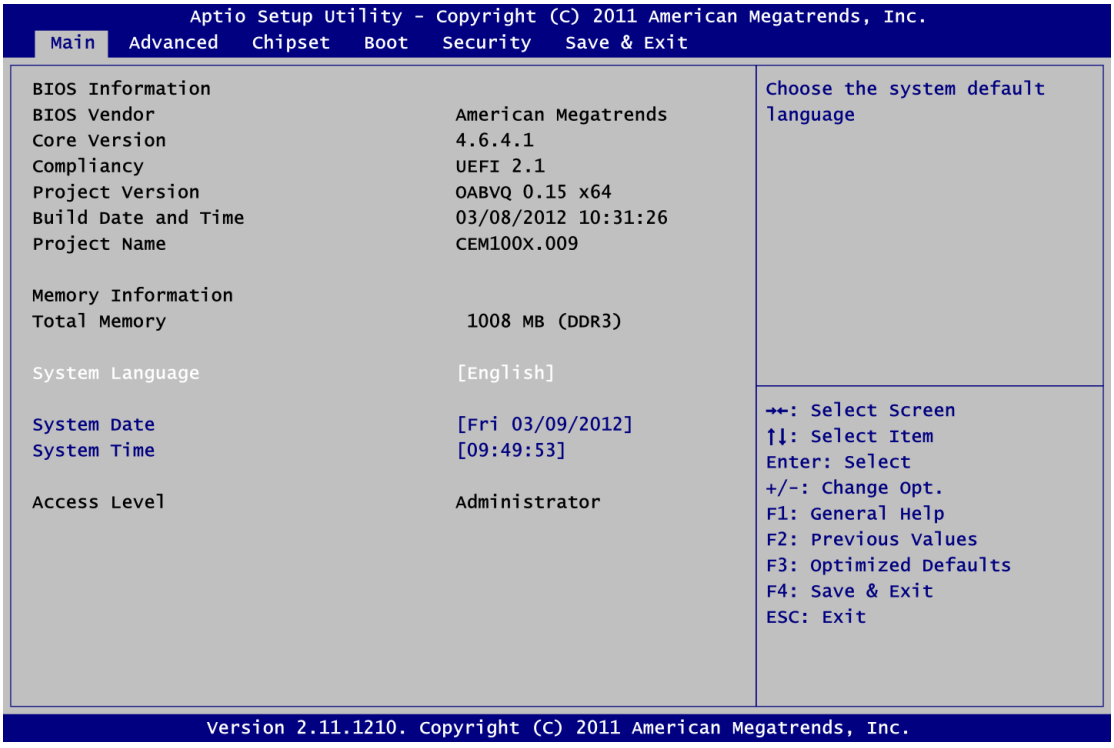
The BIOS setup/utility uses a key-based navigation system called hot keys. Most of the BIOS setup utility hot keys can be used at any time during the setup navigation process. These keys include <F1>, <F2>, <Enter>, <ESC>, <Arrow> keys, and so on.

Note: ***Some of the navigation keys differ from one screen to another.***

Hot Keys	Description
→← Left/Right	The Left and Right <Arrow> keys allow you to select a setup screen.
↑↓ Up/Down	The Up and Down <Arrow> keys allow you to select a setup screen or sub-screen.
+– Plus/Minus	The Plus and Minus <Arrow> keys allow you to change the field value of a particular setup item.
Tab	The <Tab> key allows you to select setup fields.
F1	The <F1> key allows you to display the General Help screen.
F2	The <F2> key allows you to Load Previous Values.
F3	The <F3> key allows you to Load Optimized Defaults.
F4	The <F4> key allows you to save any changes you have made and exit Setup. Press the <F4> key to save your changes.
Esc	The <Esc> key allows you to discard any changes you have made and exit the Setup. Press the <Esc> key to exit the setup without saving your changes.
Enter	The <Enter> key allows you to display or change the setup option listed for a particular setup item. The <Enter> key can also allow you to display the setup sub- screens.

4.3 Main Menu

When you first enter the setup utility, you will enter the Main setup screen. You can always return to the Main setup screen by selecting the Main tab. System Time/Date can be set up as described below. The Main BIOS setup screen is shown below.



System Language
Use this item to choose the system default language.

System Date/Time
Use this option to change the system time and date. Highlight System Time or System Date using the <Arrow> keys. Enter new values through the keyboard. Press the <Tab> key or the <Arrow> keys to move between fields. The date must be entered in MM/DD/YY format. The time is entered in HH:MM:SS format.

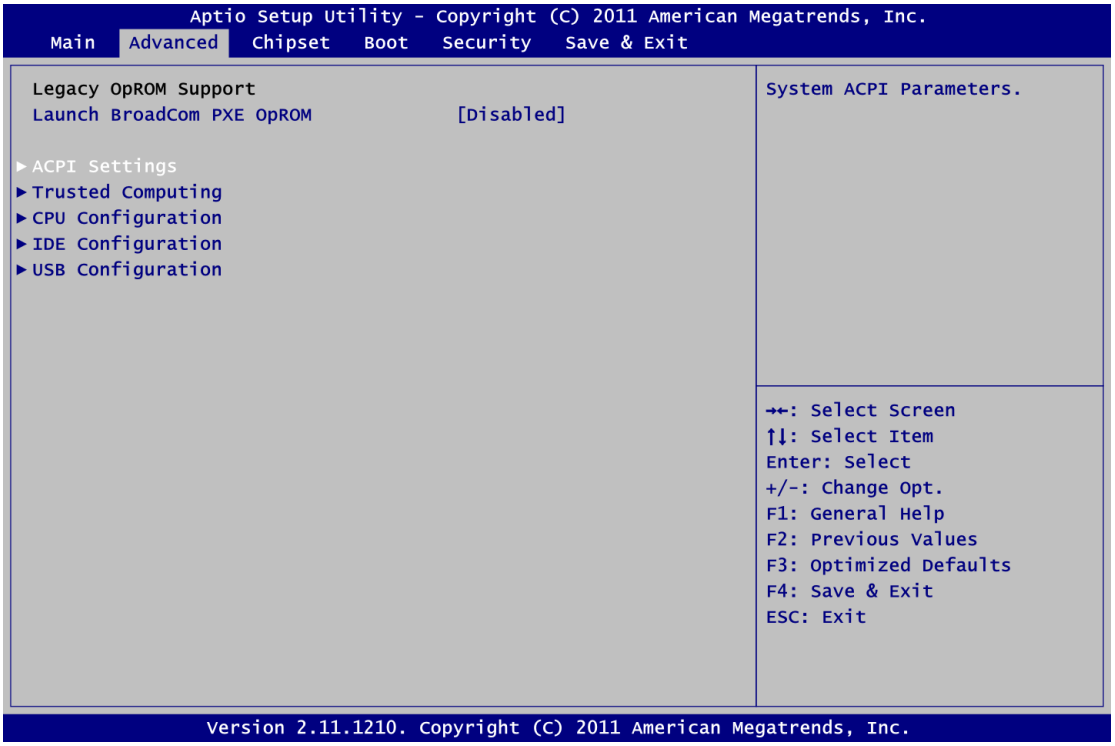
4.4 Advanced Menu

Launch BroadCom PXE OpROM
Use this item to enable or disable the boot ROM function of the onboard LAN chip when the system boots up.

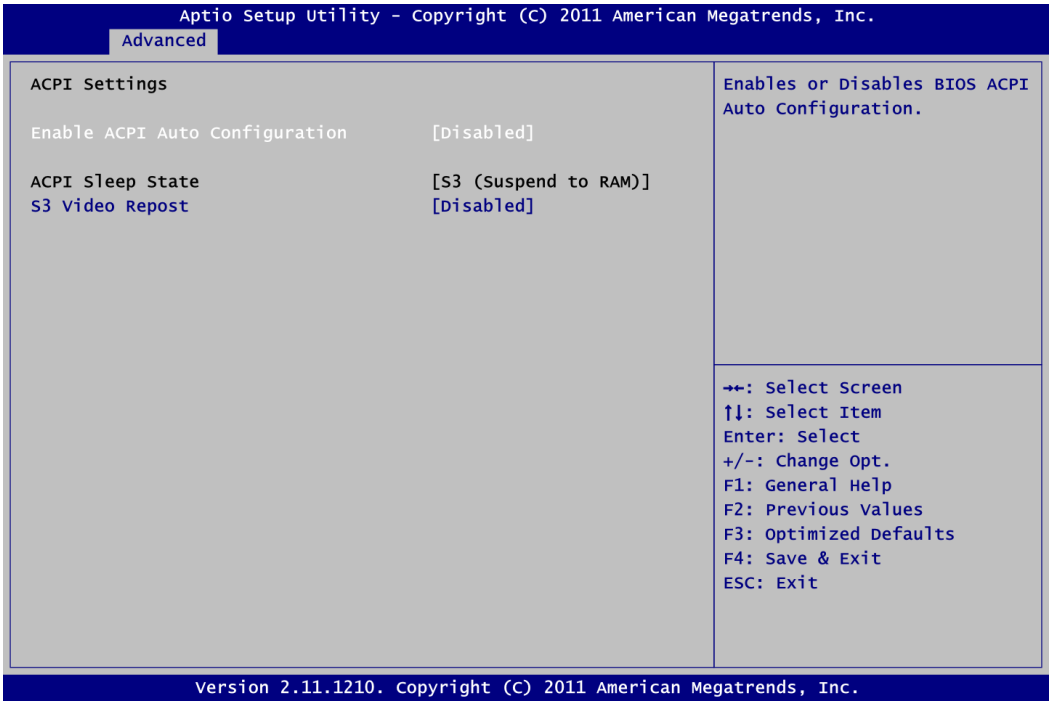
The Advanced menu also allows users to set configuration of the CPU and other system devices. You can select any of the items in the left frame of the screen to go to the sub menus:

- ▶ ACPI Settings
- ▶ Trusted Computing
- ▶ CPU Configuration
- ▶ IDE Configuration
- ▶ USB Configuration

For items marked with "▶", please press <Enter> for more options.



- **ACPI Settings**
  You can use this screen to select options for the ACPI configuration, and change the value of the selected option. A description of the selected item appears on the right side of the screen.

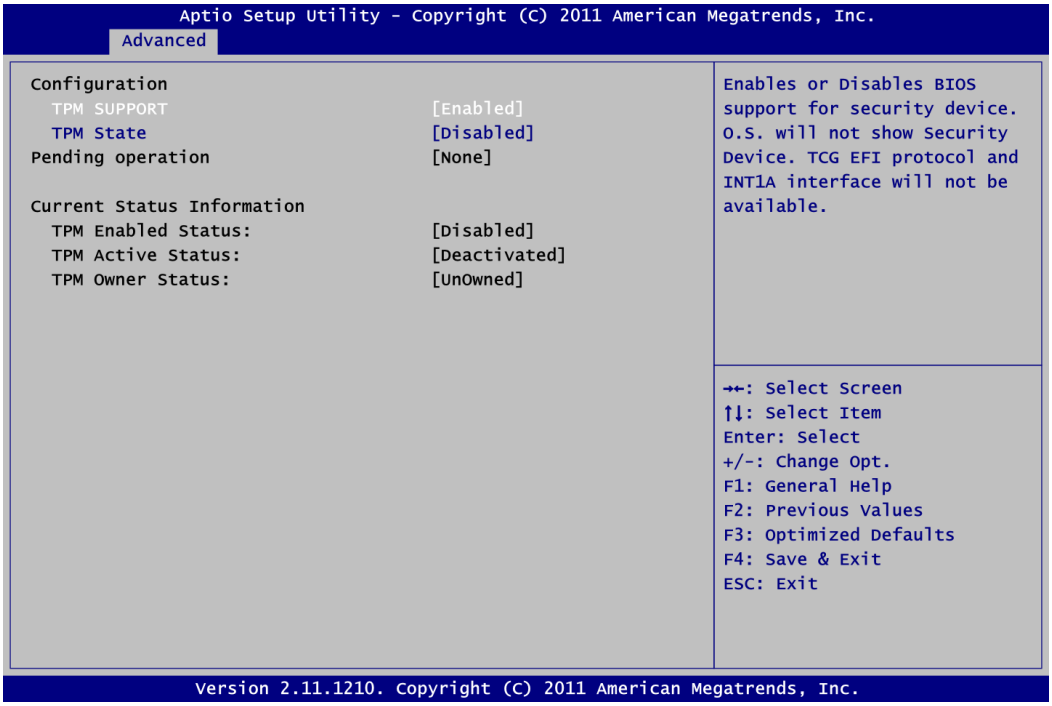


Enable ACPI Auto Configuration
Use this item to enable or disable BIOS ACPI auto configuration.

ACPI Sleep State
Allow you to select the Advanced Configuration and Power Interface (ACPI) sleep state. Here are the options for your selection; Suspend Disabled, S1 (CPU Stop Clock) and S3 (Suspend to RAM).

S3 Video Repost
Enable or disable S3 video repost.

- **Trusted Computing**
  You can use this screen for TPM configuration. It also shows current TPM status information.

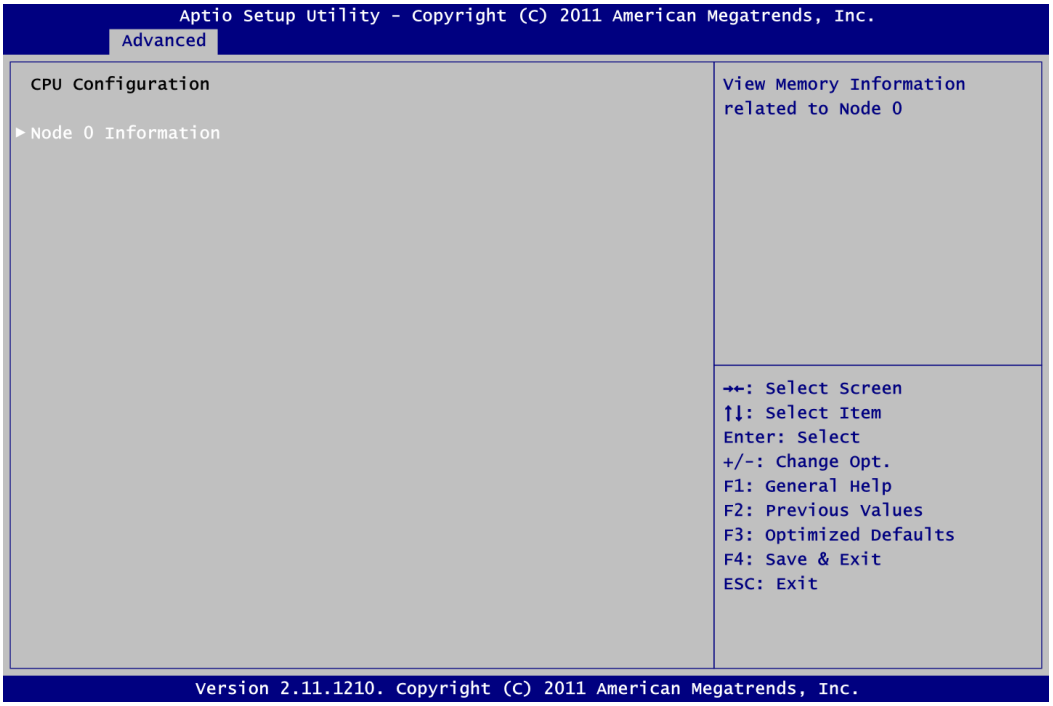


TPM Support
Allow you to enable or disable TPM support.

TPM State
Allow you to enable or disable TPM state.

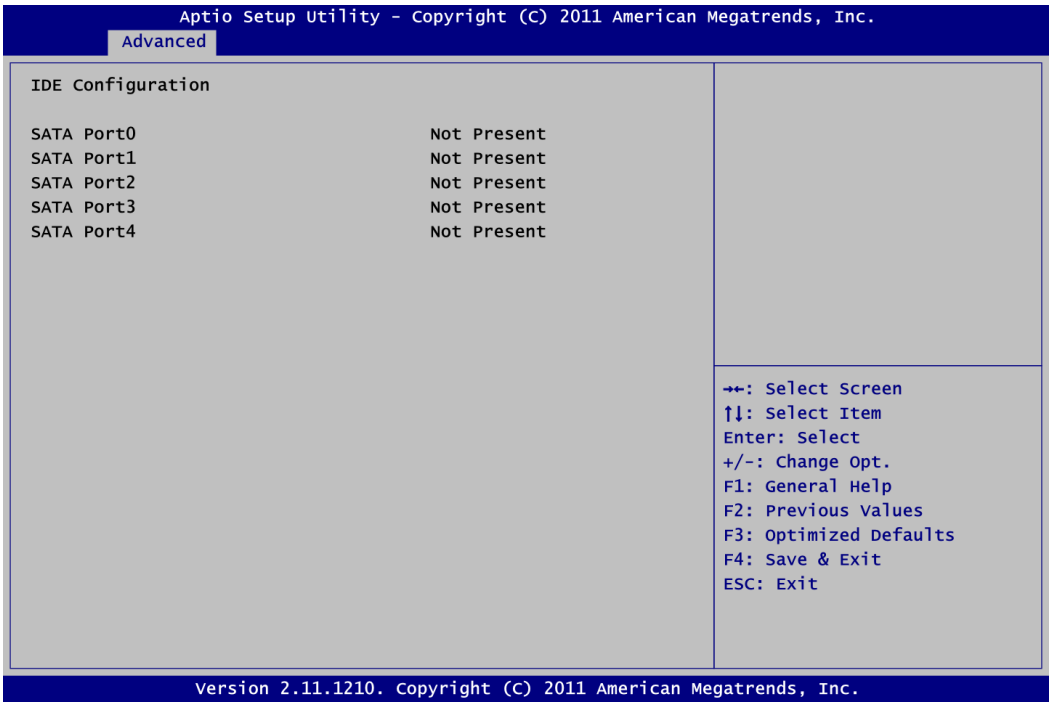
Current Status Information
Display current TPM status information.

- **CPU Configuration**
  This screen shows the CPU Configuration, and you can change the value of the selected option.

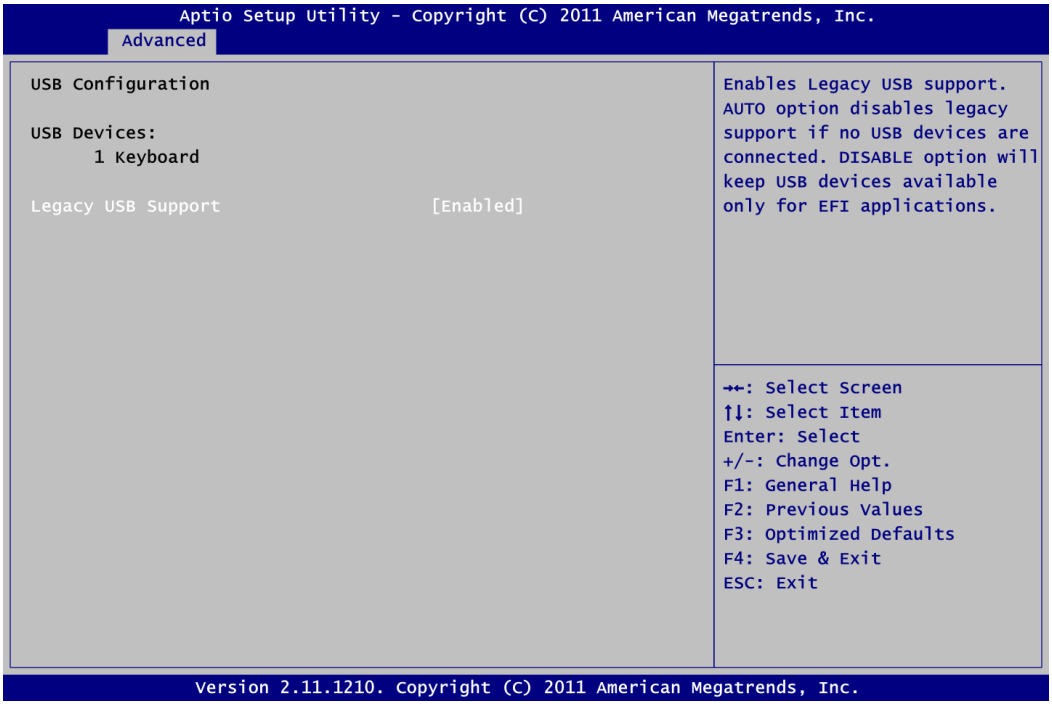


Node 0 Information
View memory information related to Node 0.

- **IDE Configuration**
  In the IDE Configuration menu, you can see the currently installed hardware in the SATA ports. During system boot up, the BIOS automatically detects the presence of SATA devices.



- **USB Configuration**
  You can use this screen to select options for the USB Configuration, and change the value of the selected option. A description of the selected item appears on the right side of the screen.



USB Devices
Displays all detected USB devices.

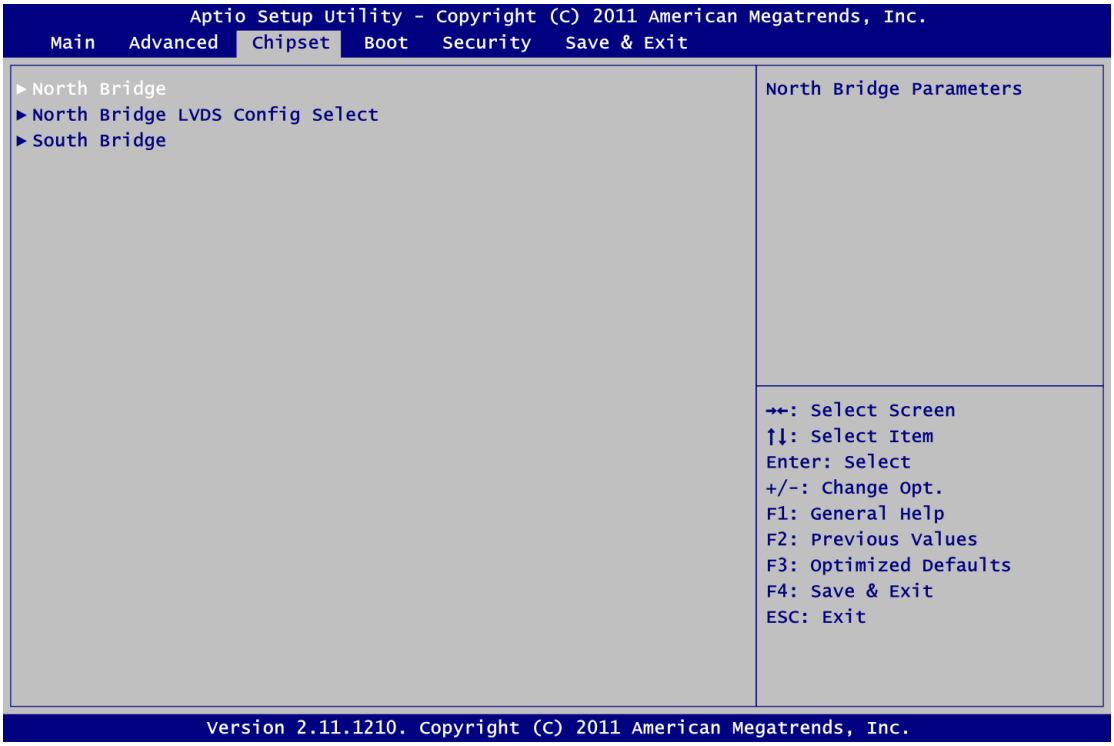
Legacy USB Support
Use this item to enable or disable support for USB device on legacy operating system. The default setting is Enabled. Auto option disables legacy support if no USB devices are connected. Disable option will keep USB devices available only for EFI applications.

4.5 Chipset Menu

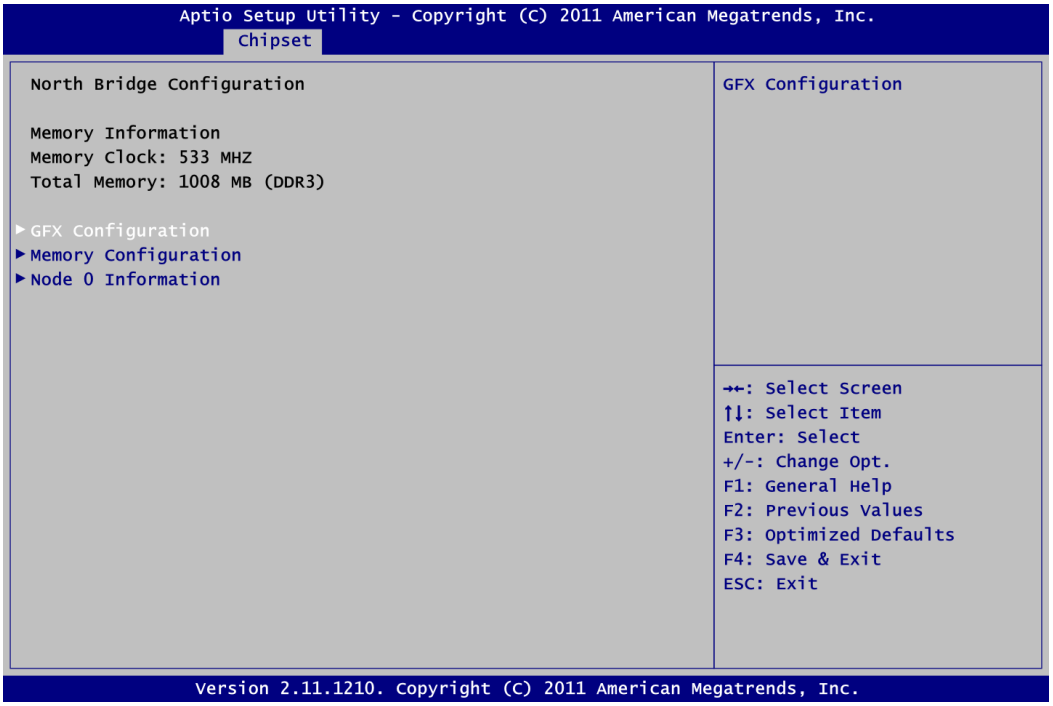
The Chipset menu allows users to change the advanced chipset settings. You can select any of the items in the left frame of the screen to go to the sub menus:

- ► North Bridge
- ► North Bridge LVDS Config Select
- ► South Bridge

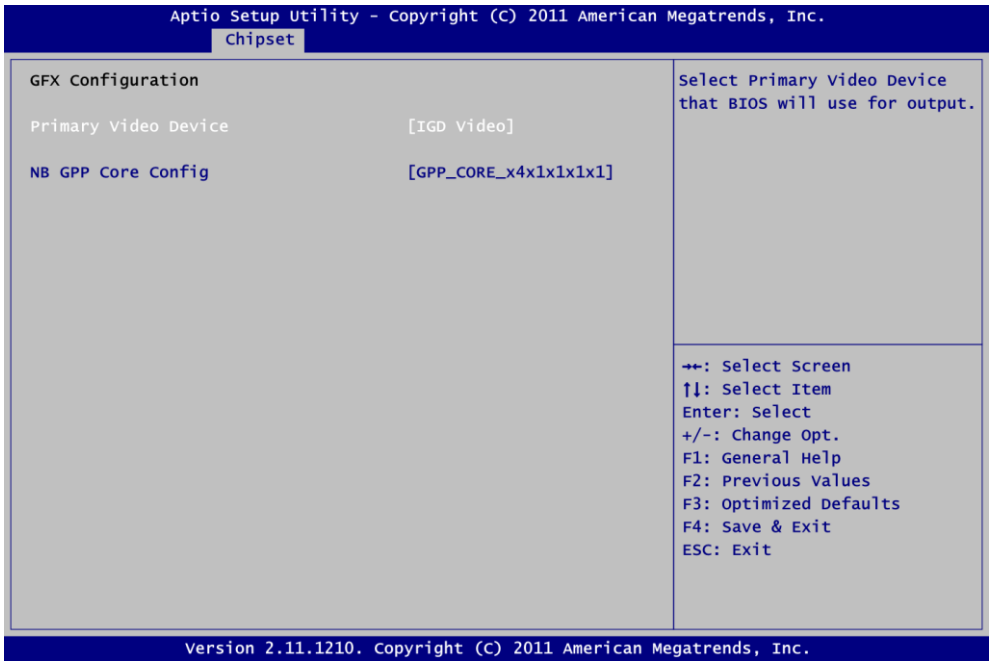
For items marked with "►", please press <Enter> for more options.



- **North Bridge Configuration**
  This screen allows users to configure parameters of North Bridge chipset. For items marked with "►", please press <Enter> for more options.



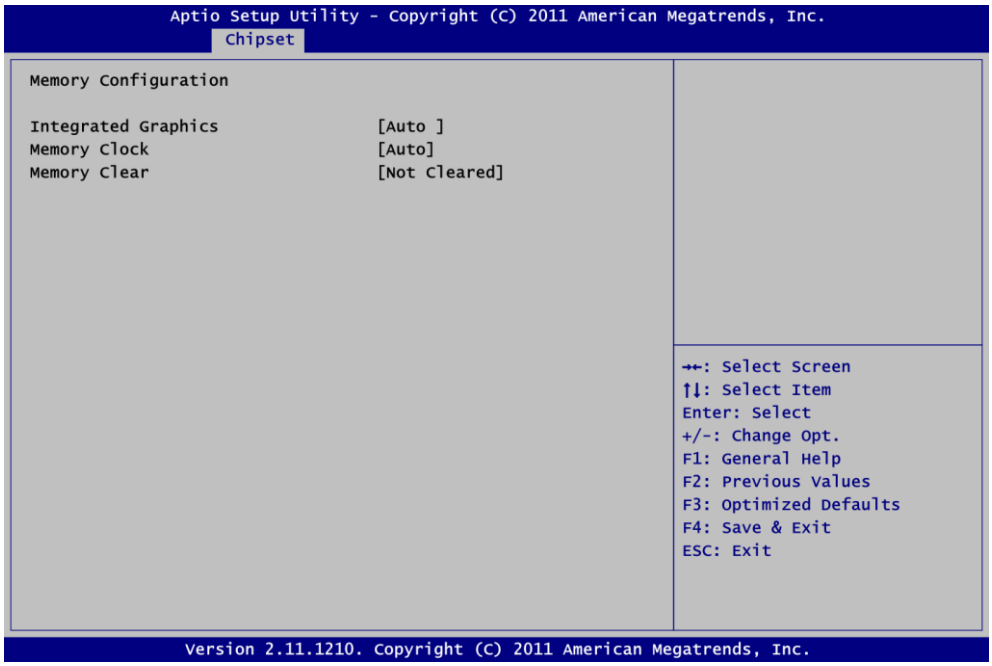
- **GFX Configuration**



Primary Video Device
Select primary video device that BIOS will use for output. Default option is IGD Video.

NB GPP Core Config
This item is for NB GPP Core configuration.

- **Memory Configuration**

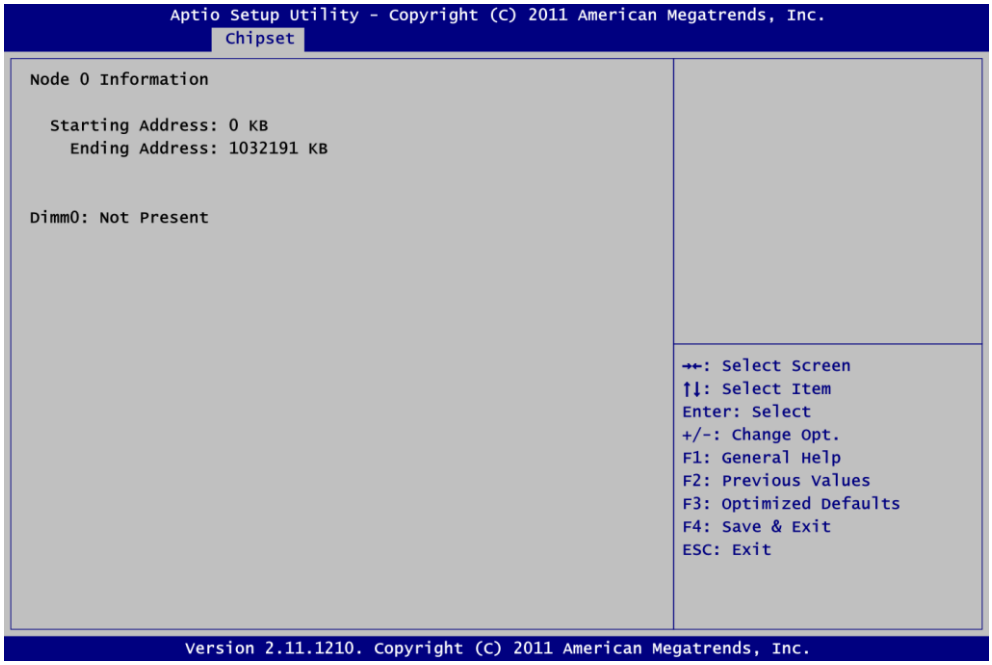


Integrated Graphics
The Integrated Graphics controller configuration is set to Auto.

Memory Clock
This item allows user to select different memory clock.

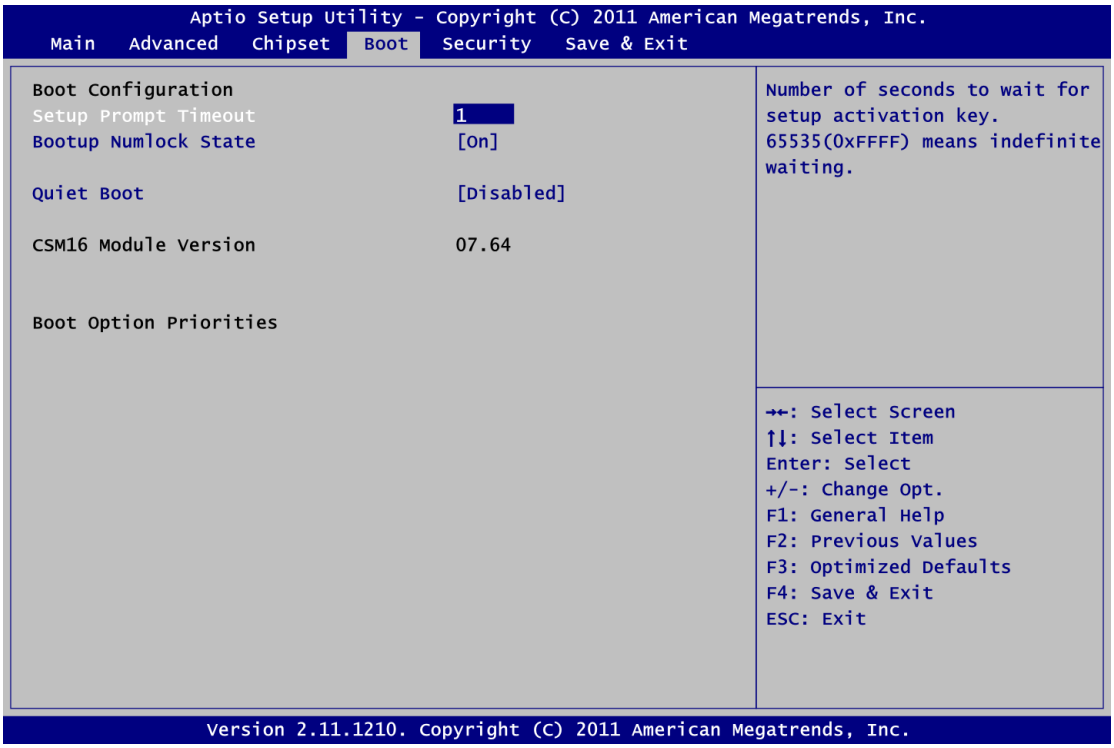
Memory Clear
This is for memory clear functionality control.

- **Node 0 Information**
This screen provides user with the information of current using DDR3 SDRAM.



4.6 Boot Menu

The Boot menu allows users to change boot options of the system.



- **Setup Prompt Timeout**
  Number of seconds to wait for setup activation key. 65535(0xFFFF) means indefinite waiting.

- **Bootup NumLock State**
  Use this item to select the power-on state for the keyboard NumLock.

- **Quiet Boot**
  Select to display either POST output messages or a splash screen during boot-up.

- **Boot Option Priorities**
  Specify the boot device priority sequence from the available devices.

4.7 Security Menu

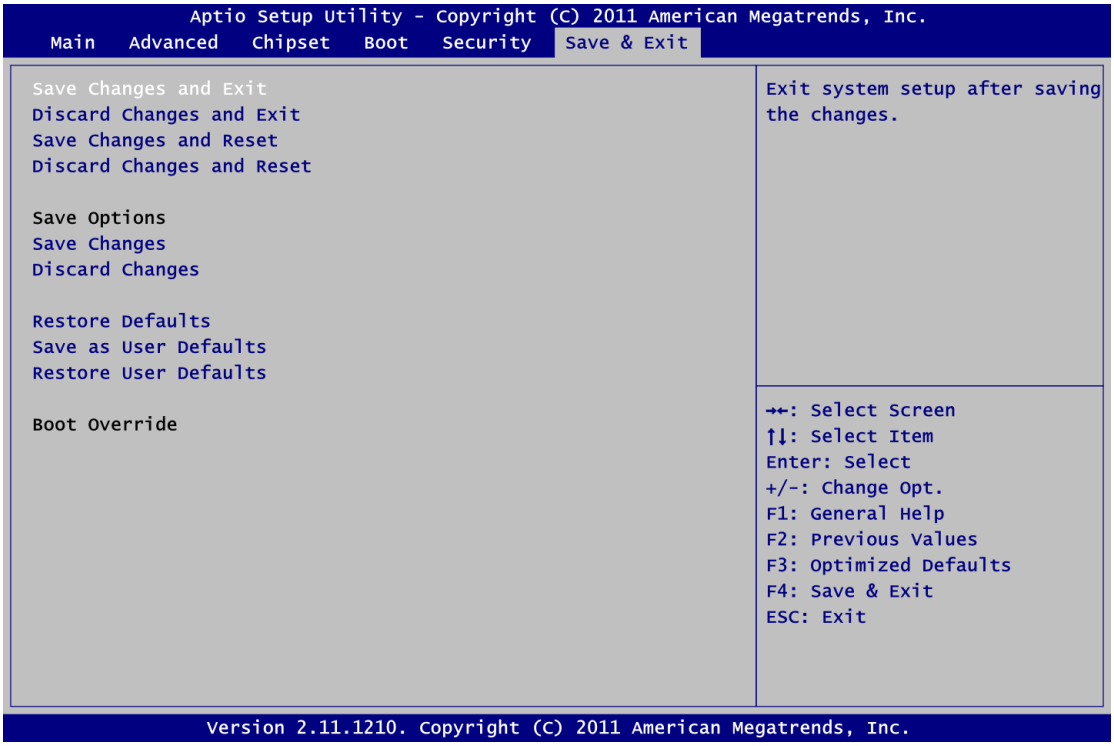
The Security menu allows users to change the security settings for the system.

- **Administrator Password**
  This item indicates whether an administrator password has been set (installed or uninstalled).

- **User Password**
  This item indicates whether an user password has been set (installed or uninstalled).

4.8 Save & Exit Menu

The Save & Exit menu allows users to load your system configuration with optimal or fail-safe default values.



- **Save Changes and Exit**
  When you have completed the system configuration changes, select this option to leave Setup and return to Main Menu. Select Save Changes and Exit from the Save & Exit menu and press <Enter>. Select Yes to save changes and exit.

- **Discard Changes and Exit**
  Select this option to quit Setup without making any permanent changes to the system configuration and return to Main Menu. Select Discard Changes and Exit from the Save & Exit menu and press <Enter>. Select Yes to discard changes and exit.

- **Save Changes and Reset**
  When you have completed the system configuration changes, select this option to leave Setup and reboot the computer so the new system configuration parameters can take effect. Select Save Changes and Reset from the Save & Exit menu and press <Enter>. Select Yes to save changes and reset.

- **Discard Changes and Reset**
  Select this option to quit Setup without making any permanent changes to the system configuration and reboot the computer. Select Discard Changes and Reset from the Save & Exit menu and press <Enter>. Select Yes to discard changes and reset.

- **Save Changes**
  When you have completed the system configuration changes, select this option to save changes. Select Save Changes from the Save & Exit menu and press <Enter>. Select Yes to save changes.

- **Discard Changes**
  Select this option to quit Setup without making any permanent changes to the system configuration. Select Discard Changes from the Save & Exit menu and press <Enter>. Select Yes to discard changes.

- **Restore Defaults**
  It automatically sets all Setup options to a complete set of default settings when you select this option. Select Restore Defaults from the Save & Exit menu and press <Enter>.

- **Save as User Defaults**
  Select this option to save system configuration changes done so far as User Defaults. Select Save as User Defaults from the Save & Exit menu and press <Enter>.

- **Restore User Defaults**
  It automatically sets all Setup options to a complete set of User Defaults when you select this option. Select Restore User Defaults from the Save & Exit menu and press <Enter>.

- **Boot Override**
  Select a drive to immediately boot that device regardless of the current boot order.

Appendix A
Watchdog Timer and GPIO

About Watchdog Timer

Software stability is major issue in most application. Some embedded systems are not watched by human for 24 hours. It is usually too slow to wait for someone to reboot when computer hangs. The systems need to be able to reset automatically when things go wrong. The watchdog timer gives us solution.

The watchdog timer is a counter that triggers a system reset when it counts down to zero from a preset value. The software starts counter with an initial value and must reset it periodically. If the counter ever reaches zero which means the software has crashed, the system will reboot.

About GPIO

The onboard GPIO (general input and output) has 8 bits (GPI0~3 and GPO0~3). In default, all pins are pulled high with +3.3V level (according to main power). The BIOS default settings are 4 inputs and 4 outputs where all of these pins are set to 1. Use these GPIO signals to control cash drawers and sense warning signals from an Uninterrupted Power System (UPS), or perform store security control.

Sample Program

Programming sample code (from CEM100 FINTEK F75111R):

```
/*---------------- INCLUDE FILE ----------------------*/
#include <stdio.h>
#include <conio.h>
#include <bios.h>
#include <dos.h>

#define UCHAR   unsigned char
#define UINT    unsigned int
#define SMIOBASE      0x0B00
/******************
       SMIOBASE can get from PCI device Bus-0,Device-31, Function-3
       Register 20h~23h, the value is the IO base address.
*******************/
#define SM_REG        (SMIOBASE+3)
#define SM_ADDR       (SMIOBASE+4)
#define SM_DATA       (SMIOBASE+5)
#define SM_CMD        (SMIOBASE+2)
#define SM_STATUS     (SMIOBASE+0)
#define SM_byteAccess 0x48
#define Device_Addr   0x6E  // depend on hardware designed Low:0x9C, High:0x6E
UCHAR _read_smbus(UCHAR ,UCHAR);
void _write_smbus(UCHAR,UCHAR, UCHAR);
#define F75111_CHIPID           0x0003
#define F75111_VENDORID         0x3419
#define WDTOUT10_CntlReg1       0x34
#define WDTOUT10_2S_bit         0x04    //bit 2
#define WDTOUT10_OINV_bit       0x02    //bit 1
#define WDTOUT10_Status_bit     0x01    //bit 0 , write 1 for clear status
#define WDTOUT10_CntlReg2       0x35
#define WDTOUT10_Enable_bit     0x80    //bit 7
```

```
#define WDTOUT10_PTIME          0x7f    //bit 0~6

/* ------------------ MAIN PROGRAM ------------------*/
main()
{
UCHAR   xch,xch2;
UINT    chipid=0, vendorid=0;
       //Check the Chip ID information
       xch=_read_smbus(Device_Addr,0x5a);     //Chip ID 1
       xch2=_read_smbus(Device_Addr,0x5b);    //Chip ID 2
       chipid=((UINT)xch2 << 8) + (UINT)xch;
       xch=_read_smbus(Device_Addr,0x5d);     //Vendor ID 1
       xch2=_read_smbus(Device_Addr,0x5E);    //Vendor ID 2
       vendorid=((UINT)xch2 << 8) + (UINT)xch;
       if (F75111_CHIPID != chipid || F75111_VENDORID != vendorid )
         { printf("!!! Not found F75111 chip !!!\n");
           exit(-1);
         }
       printf("=== Found F75111 chip ===\n");
       printf("=== GPIO Output Test ===\n");

       //set GPIO3x  direction
       printf("Set F75111 GPIO3x pin is output direction\n");
       _write_smbus(Device_Addr,0x40,0x0f);   //GPIO3x Output direction

       //set GPIO3x output level or plus
       printf("Set F75111 GPIO3x pin is output Level\n");
       _write_smbus(Device_Addr,0x43,0);      //GPIO3x Level control

       //write GPIO3x data
       printf("Write GPIO3x data is 0x0A\n");
       printf("GPO0=0, GPO1=1, GPO2=0, GPO3=1\n");
       _write_smbus(Device_Addr,0x41,0x0a);
       printf("Please check the GPO level and hit any key to continue\n");
       getch();

       //write GPIO3x data
       printf("Write GPIO3x data is 0x05\n");
       printf("GPO0=1, GPO1=0, GPO2=1, GPO3=0\n");
       _write_smbus(Device_Addr,0x41,0x05);
       printf("Please check the GPO level and hit any key to continue\n");
       getch();


       //set GPIO10,11,12 used
       printf("=== GPIO Input Test ===\n");
       printf("Set F75111 GPIO1x is used GPIO function\n");
       xch=_read_smbus(Device_Addr,0x03);
       xch &= 0xE0 ;
       _write_smbus(Device_Addr,0x03,xch);    //set Pin GPIO10/11/12 used
       _write_smbus(Device_Addr,0x04,0);      //set Pin GPIO1x used

       printf("Set F75111 GPIO10,11,12,13 is input function\n");
       _write_smbus(Device_Addr,0x10,0x00);   //set GPIO1x input direction

       printf("Set F75111 GPIO10,11,12,13 is Level mode\n");
       _write_smbus(Device_Addr,0x13,0x00);   //set GPIO1x Level Control

       xch=_read_smbus(Device_Addr,0x12);     //read GPIO1x Status
       printf("Read the GPI0,1,2,3 input data is %02X\n",xch);
       printf("Please Change the GPIx input and hit any key to continue\n");
       getch();
       xch=_read_smbus(Device_Addr,0x12);     //read GPIO1x Status
       printf("Read the GPI0,1,2,3 input data is %02X\n",xch);
       printf("Please Change the GPIx input and hit any key to continue\n");
       getch();
       xch=_read_smbus(Device_Addr,0x12);     //read GPIO1x Status
       printf("Read the GPI0,1,2,3 input data is %02X\n",xch);
       /*****************************************/
       printf("===== WatchDogTimer Test =====\n");
       printf("Set WDTOUT10 pin used\n");
       _write_smbus(Device_Addr,0x01,0x20);   //Pin1 config

       printf("Set WDTOUT10 output 100ms plus, normal level\n");
       _write_smbus(Device_Addr,WDTOUT10_CntlReg1,0);      //WDT10 control

       printf("Set WDTOUT10 Time 10 seconds and enable WDT\n");
```

```
        _write_smbus(Device_Addr,WDTOUT10_CntlReg2,0x8A);   //WDT10 control
        printf("Please hit any key in period of 10 seconds\n");
        getch();
        _write_smbus(Device_Addr,WDTOUT10_CntlReg2,0x8A);   //WDT10 control
        printf("The sytsem will reset when the 10 seconds times out\n");
}

void _write_smbus(UCHAR xAddr,UCHAR xReg, UCHAR xData)
{
        while (1)
        { if (_check_smbus_busy()==0) break;
        }
        outp(SM_REG, xReg);
        xdelay();           //because the CPU too fast,delay for IO
        outp(SM_ADDR, xAddr);
        xdelay();           //because the CPU too fast,delay for IO
        outp(SM_DATA, xData);
        xdelay();           //because the CPU too fast,delay for IO
        outp(SM_CMD, SM_byteAccess);
        xdelay();           //because the CPU too fast,delay for IO
        outp(SM_STATUS, 02); //clear interrupt status
}

UCHAR _read_smbus(UCHAR xAddr,UCHAR xReg)
{
UCHAR   xch,xch2;
        while (1)
        { if (_check_smbus_busy()==0) break;
        }
        outp(SM_REG, xReg);
        xdelay();           //because the CPU too fast,delay for IO
        outp(SM_ADDR, xAddr+1);
        xdelay();           //because the CPU too fast,delay for IO
        outp(SM_CMD, SM_byteAccess);
        xdelay();           //because the CPU too fast,delay for IO
        while (1)
        { if (_check_smbus_busy()==0) break;
        }
        xch=inp(SM_DATA);
        xdelay();           //because the CPU too fast,delay for IO
        outp(SM_STATUS, 2);  //clear interrupt status
        return xch;
}

_check_smbus_busy(void)
{
UCHAR   xch;
        xch=inp(SM_STATUS);
        if (xch & 0x02 ) outp(SM_STATUS, 2);   //clear interrupt status
        if ( xch & 0x02 ) return 1;
        if ( xch & 0x01 ) return 1;
        return 0;
}

xdelay()
{
int     xxi,xxj,xxk=0;
        for (xxi=0 ; xxi< 0x1000 ; xxi++) {
                for (xxj=0 ; xxj < 0x100 ; xxj++) {
                        xxk++;
                }
        }
}
```