MI958

AMD G-series APU+A55E FCH

Mini-ITX Motherboard

USER'S MANUAL

Version 1.0

Acknowledgments

Table of Contents

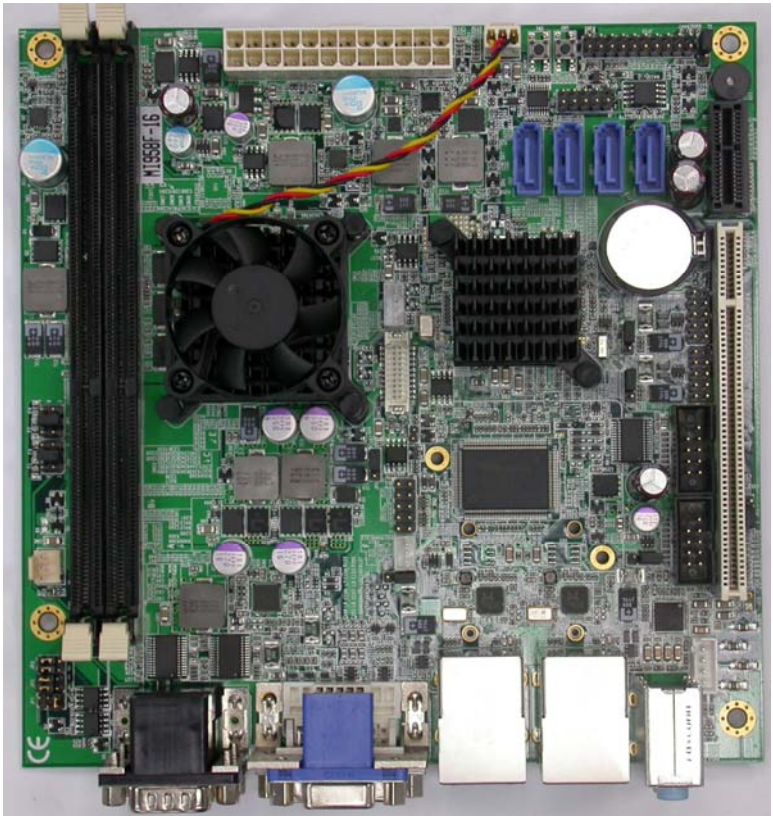
Introduction 1

Product Description 1
Checklist 2
Board Specifications 3
Board Dimensions 4

Installations 5

Installing the Memory 6
Setting the Jumpers 7
Connectors 12

BIOS Setup 21



THE MI958 MINI ITX MOTHERBOARD

Introduction

Product Description

The AMD Embedded G-Series Processors with AMD Radeon™ HD 6000 Series Graphics combine the central processing unit (CPU) with the graphics processing unit (GPU) in a single-chip Accelerated Processing Unit (APU) package. The APU connects to the AMD A55E Fusion Controller Hub through the Unified Media Interface (UMI) to provide connections to the different system devices. Dimensions of the board are 170mm x 170mm.

AMD Embedded G-Series processor-based systems include the DDR3 memory interface, PCIe, UMI, DDI (Digital Display Interface), power delivery and miscellaneous test signals that connect to the processor.

The AMD A55E Fusion Controller Hub is designed to deliver the quality and performance needed for everyday computing, ultitasking, and multimedia functionality. It also provides expanded I/O connectivity for advanced usage models to take advantage of modern peripheral devices.

The main features of the board are:

- Supports Dual Core AMD Embedded G-Series Processors
- Supports up to DDR3-1066 MT/s
- Two DDR3 SDRAM DIMMs, Max.8GB memory
- Onboard Realtek Gigabit LAN
- 4x SATA-III, 8x USB 2.0, 4x COM, Watchdog timer,
- 1x PCI,1x PCI-E(x1) slots
- 1x VGA,1x DVI,1x LVDS

Checklist

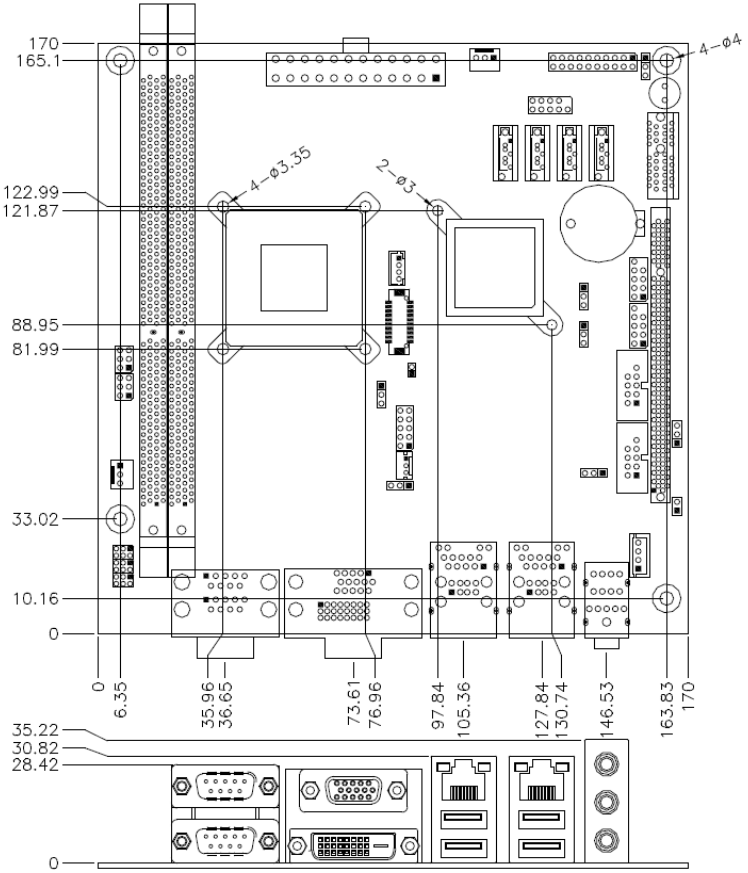
Your AMD MI958 Development Board package should include the items listed below.

- The MI958 Mini-ITX motherboard
- This User's Manual
- 1 CD containing chipset drivers and flash memory utility
- 1 Serial ATA cable

MI958 Specifications

Product Name	MI958F-10 (G-T40N APU onboard) MI958F-14 (G-T48N APU onboard) MI958F-16 (G-T56N APU onboard) MI958F-165 (G-T56N C0 version APU onboard)
Form Factor	Mini ITX
CPU Type	AMD G-series APU Processor, 40 nm FT1 413-BGA package (19mm x 19mm)
CPU Operating Frequency	G-T56N C0 version DC @1.65GHz (TDP=18W) G-T56N DC @1.6GHz (TDP=18W) G-T48N DC @1.4GHz (TDP=18W) G-T40N DC @1.0GHz (TDP=9W)
Cache	1MB L2 cache
BIOS	AMI BIOS, supports ACPI Function
Chipset	AMD A55E FCH (TDP=4.7W) BGA package 605balls (23 mm x 23 mm)
Memory	2 x DDR3 UDIMM, Single Channel, Max. 8GB Up to DDR3-1333(G-T56N C0 version only) / DDR3-1066
Display	AMD G-series APU built-in GPU (G-T56N/G-T48N w/Radeon™ HD6310; G-T40N w/Radeon™ HD6250) Supports DirectX®11, UVD3. Supports two simultaneous displays 1 x CRT [From APU DAC] 1 x DVI-**D** [From APU DP #1)
Expansion Slots	PCI slot x 1 PCIe(1x) slot x1 MiniPCIe(1x) x 1
LAN	Realtek 8111**E** PCI-Express GbE x **2** [From APU] **Only LAN #1 will support EuP**
USB	A55E FCH built-in USB 2.0 host controller, support 8 ports (4 x rear I/O ports + 4 x internal pin headers)
Audio	A55E FCH built-in HD interface + Realtek ALC269QHD Codec w/class-D speaker amplifier(2.3W per channel @ 5V power supply) support 2-channel audio out + amp
Serial ATA	A55E FCH built-in SATA III Controller for **4**
LPC I/O	Fintek F81865-I COM1 (RS232/422/485), COM2/COM3/COM4 (RS232 only), Hardware Monitor (2 thermal inputs,4 voltage monitor inputs & 2 Fan headers) [CPU FAN controllable only; SYS FAN cannot] COM1/2 with pin-9 with power for 2 ports (500 mA for each port)
Clock Generator	Leverage A55E FCH internal Clock generator
Digital IO	4 in & 4 out
Edge Connector	Dual DB9 stack connector for COM #1 , #2 DVI-**D**+ DB15 stack connector x 1 for DVI & CRT RJ-45 GbE LAN + dual USB stack connector x2 RCA Jack 3 x 1 for HD Audio
On Board Headers	DF13 LVDS x 1 for 18-bit single channel 2 x 5 pins header x 2 for 4 ports USB 2 x 5 pins header x 2 for COM3/COM4 2 x 5 pins pin-header x1 for Digital IO 1 x 4 pins box header x 1 for speaker 1 x 4 pins box-header x 1 for LCD backlight control SATA connector x 4 (blue color)
Watchdog Timer	Yes (256 segments, 0, 1, 2…255. sec/min)
Power Connector	ATX Power connector
RoHS	Yes
Others	CPU cooler onboard for T56N/T48N APU ; Heatsink for T40N APU EuP feature onboard (Fintek F75160)
Board Size	170mm x 170mm

Board Dimensions



Installations

This section provides information on how to use the jumpers and connectors on the motherboard in order to set up a workable system. The topics covered are:

Installing the Memory 6
Setting the Jumpers 7
Connectors 12

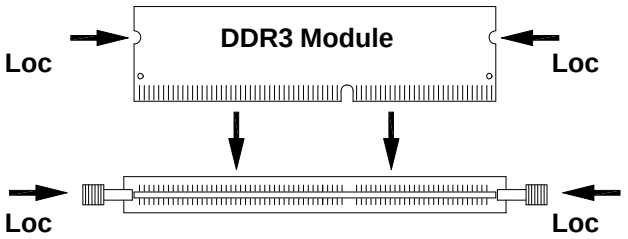
Installing the Memory

The motherboard supports two DDR3 memory socket for a maximum total memory of 8GB in DDR3 memory type.

Installing and Removing Memory Modules

To install the DDR3 modules, locate the memory slot on the board and perform the following steps:

1. Hold the DDR3 module so that the key of the DDR3 module align with those on the memory slot.
2. Gently push the DDR3 module in an upright position until the clips of the slot close to hold the DDR3 module in place when the DDR3 module touches the bottom of the slot.
3. To remove the DDR3 module, press the clips with both hands.

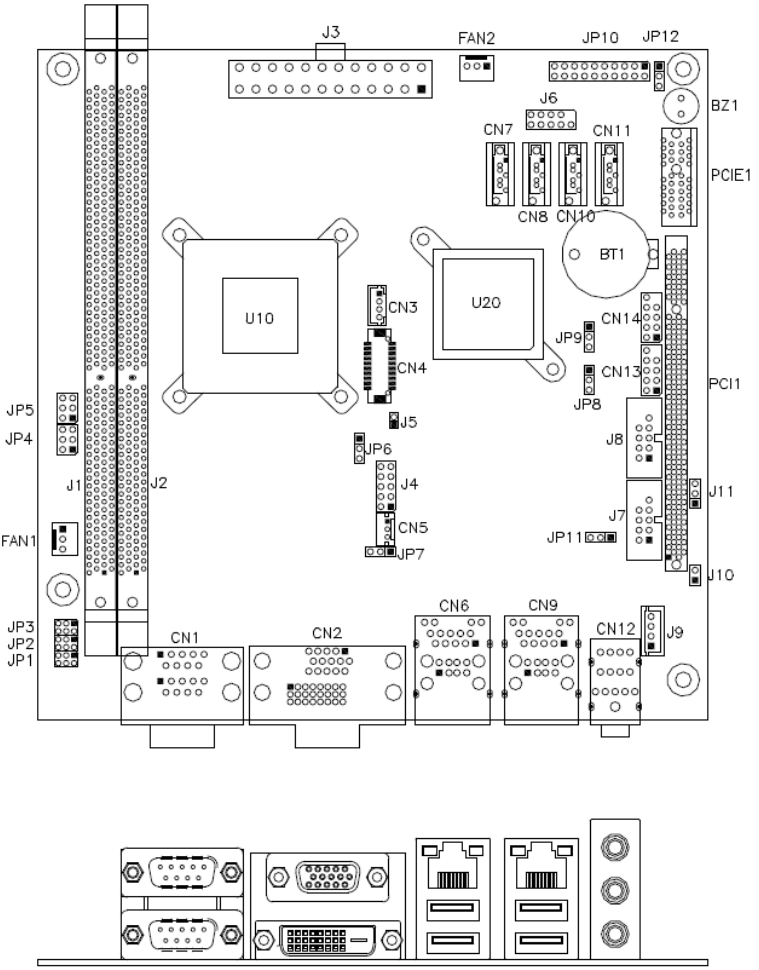


Setting the Jumpers

Jumpers are used on the motherboard to select various settings and features according to your needs and applications. Contact your supplier if you have doubts about the best configuration for your needs. The following lists the connectors and their respective functions.

Jumper Locations 8
JP6: LCD Panel Power Selection 9
JP12: Clear CMOS Setting..... 9
JP1, JP2, JP3: RS232/422/485 (COM1) Selection……..…9
JP4 :COM1 RS232 +5V/+12V Power Setting 10
JP5 :COM2 RS232 +5V/+12V Power Setting 10
JP7 : USB1, USB2 (CN6) Power Setting..... 10
JP11 : USB3, USB4 (CN9) Power Setting..... 10
JP8 : USB5, USB6 (CN13) Power Setting..... 11
JP9 : USB7, USB8 (CN14) Power Setting..... 11
J11: PCI/PCIE Riser Card Selection 11

Jumper Locations



JP6: LCD Panel Power Selection

JP6	LCD Panel Power
	3.3V
	5V

JP12: Clear CMOS Setting

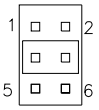
JP13	Setting
	Normal
	Clear CMOS

JP1,JP2,JP3: RS232/422/485 (COM1) Selection
COM2~COM4 are fixed for RS-232 use only.

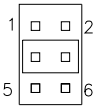
COM1 is selectable for RS232, RS-422 and RS-485.
The following table describes the jumper settings for COM1 selection.

COM1 Function	RS-232	RS-422	RS-485
Jumper Setting (pin closed)	JP3: 3-5 & 4-6 JP2: 3-5 & 4-6 JP1: 1-2	JP3: 1-3 & 2-4 JP2: 1-3 & 2-4 JP1: 3-4	JP3: 1-3 & 2-4 JP2: 1-3 & 2-4 JP1: 5-6

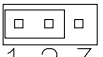
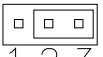
JP4: COM1 RS232 +5V/+12V Power Setting

JP4	Setting	Function
	Pin 1-2 Short/Closed	+12V
	Pin 3-4 Short/Closed	Normal
	Pin 5-6 Short/Closed	+5V

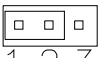
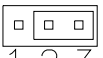
JP5: COM2 RS232 +5V/+12V Power Setting

JP5	Setting	Function
	Pin 1-2 Short/Closed	+12V
	Pin 3-4 Short/Closed	Normal
	Pin 5-6 Short/Closed	+5V

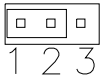
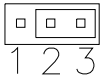
JP7: USB1, USB2 (CN6) Power Setting

JP7	Setting	Function
	Pin 1-2 Short/Closed	+5VSB
	Pin 2-3 Short/Closed	+5V

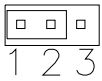
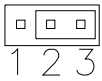
JP11: USB3, USB4 (CN9) Power Setting

JP11	Setting	Function
	Pin 1-2 Short/Closed	+5VSB
	Pin 2-3 Short/Closed	+5V

JP8: USB5, USB6 (CN13) Power Setting

JP8	Setting	Function
	Pin 1-2 Short/Closed	+5VSB
	Pin 2-3 Short/Closed	+5V

JP9: USB7, USB8 (CN14) Power Setting

JP9	Setting	Function
	Pin 1-2 Short/Closed	+5VSB
	Pin 2-3 Short/Closed	+5V

J11: PCI/PCIE Riser Card Selection

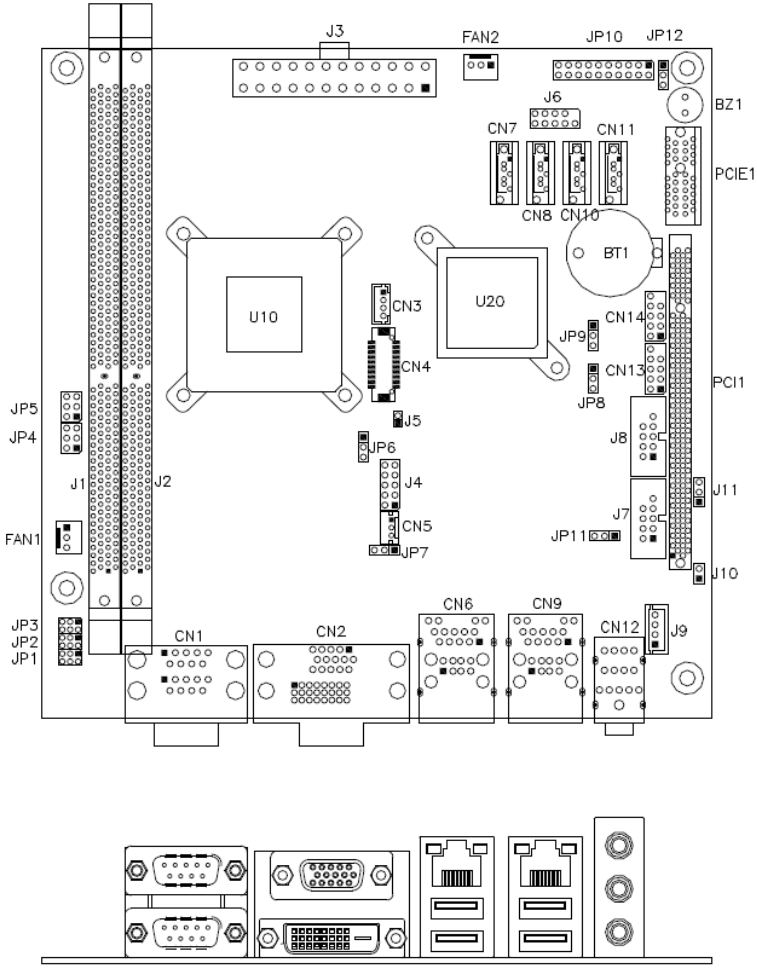
J11	Riser Card
	IP390 Riser Card Install
	IP151, IP240 Riser Card Install

Motherboard Connectors

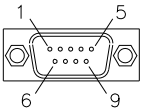
The motherboard connectors allow you to connect external devices such as keyboard, floppy disk drives, hard disk drives, printers, etc. The following table lists the connectors and their respective functions.

CN1A, CN1B: COM1(UP) and COM2(DOWN) Connector 14
CN2A, CN2B: VGA(UP) and DVI(DOWN) Connector..... 14
CN6: GbE_1 RJ-45 and USB1/2 Ports..... 14
CN9: GbE_2 RJ-45 and USB3/4 Ports..... 14
CN12: Audio Connector..... 14
CN7,CN8,CN10, CN11: Serial ATA Connectors 14
FAN1: System Fan Power Connector..... 14
FAN2: CPU Fan Power Connector..... 14
J3: ATX Power Supply Connector 15
J4: Digital I/O 15
J5: SMBUS Connector 15
J6: SPI Flash Connector (factory use only) 15
JP10: System Function Connector..... 16
CN13,CN14: USB5/6 USB7/8Port Pin Header..... 17
J7,J8: COM3/4 Serial Port..... 17
CN4: LVDS Connector..... 18
CN3: LCD Backlight Connector..... 18
CN5: Output Voltage Connector 18
J9: Speaker Connector..... 18
PCIE1: PCI-E(x1) Slot 19
PCI1: PCI Slot (supports 2 Master)..... 19

Connector Locations



CN1A, CN1B: COM1(up) and COM2(down)Connector



Signal Name	Pin #	Pin #	Signal Name
DCD	1	6	DSR
RXD	2	7	RTS
TXD	3	8	CTS
DTR	4	9	RI
GND	5	10	Not Used

CN2A, CN2B: VGA(UP) and DVI(DOWN) Connector

CN6: GbE_1 RJ-45 and USB1/2 Ports

CN9: GbE_2 RJ-45 and USB3/4 Ports

CN12: Audio Connector

The audio connector, from top to bottom, is composed of Line in, Line out and Microphone jacks.

CN7,CN8,CN10,CN11: Serial ATA Connectors

FAN1: System Fan Power Connector

FAN1 is a 3-pin header for system fan. The fan must be a 12V (500mA).

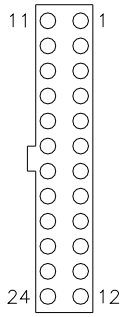
Pin #	Signal Name
1	Ground
2	+12V
3	Rotation detection

FAN2: CPU Fan Power Connector

FAN2 is a 3-pin header for the CPU fan. The fan must be a 12V(500mA).

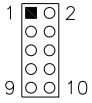
Pin #	Signal Name
1	Ground
2	+12V
3	Rotation detection

J3: ATX Power Supply Connector



Signal Name	Pin #	Pin #	Signal Name
3.3V	13	1	3.3V
-12V	14	2	3.3V
Ground	15	3	Ground
PS-ON	16	4	+5V
Ground	17	5	Ground
Ground	18	6	+5V
Ground	19	7	Ground
-5V	20	8	Power good
+5V	21	9	5VSB
+5V	22	10	+12V
+5V	23	11	+12V
Ground	24	12	+3.3V

J4: Digital I/O



Signal Name	Pin	Pin	Signal Name
GND	1	2	VCC
OUT3	3	4	OUT1
OUT2	5	6	OUT0
IN3	7	8	IN1
IN2	9	10	IN0

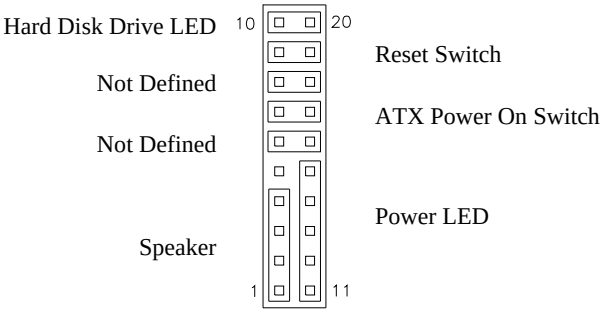
J5: SMBUS Connector

Signal Name	Pin	Pin	Signal Name
CLK	1	2	DATA

J6: SPI Flash Connector(factory use only)

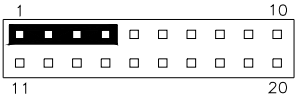
JP10: System Function Connector

JP10 provides connectors for system indicators that provide light indication of the computer activities and switches to change the computer status. JP10 is a 20-pin header that provides interfaces for the following functions.



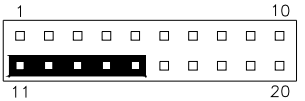
Speaker: Pins 1 - 4

This connector provides an interface to a speaker for audio tone generation. An 8-ohm speaker is recommended.



Pin #	Signal Name
1	Speaker out
2	No connect
3	Ground
4	+5V

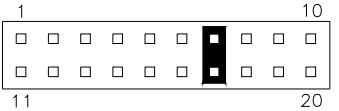
Power LED: Pins 11 - 15



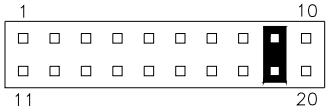
Pin #	Signal Name
11	Power LED
12	No connect
13	Ground
14	No connect
15	Ground

ATX Power ON Switch: Pins 7 and 17

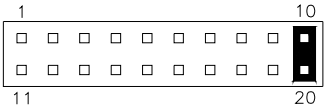
This 2-pin connector is an “ATX Power Supply On/Off Switch” on the system that connects to the power switch on the case. When pressed, the power switch will force the system to power on. When pressed again, it will force the system to power off.



Reset Switch: Pins 9 and 19
The reset switch allows the user to reset the system without turning the main power switch off and then on again. Orientation is not required when making a connection to this header.

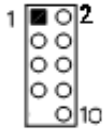


Hard Disk Drive LED Connector: Pins 10 and 20
This connector connects to the hard drive activity LED on control panel. This LED will flash when the HDD is being accessed.



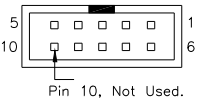
Pin #	Signal Name
10	HDD Active
20	5V

CN13,CN14: USB5/6,USB7/8 Port Pin Header



Signal Name	Pin	Pin	Signal Name
Vcc	1	2	Vcc
D0-	3	4	D1-
D0+	5	6	D1+
Ground	7	8	Ground
Key Pin	9	10	NC

J7,J8: COM3,COM4 Serial Port

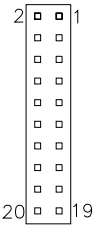


COM

Signal Name	Pin #	Pin #	Signal Name
DCD, Data carrier detect	1	6	DSR, Data set ready
RXD, Receive data	2	7	RTS, Request to send
TXD, Transmit data	3	8	CTS, Clear to send
DTR, Data terminal ready	4	9	RI, Ring indicator
GND, ground	5	10	Not Used

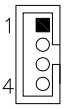
CN4 : LVDS Connectors

The LVDS connectors on board supports 18-bit.



Signal Name	Pin #	Pin #	Signal Name
TX0-	2	1	TX0+
Ground	4	3	Ground
TX1-	6	5	TX1+
5V/3.3V	8	7	Ground
DP0-HPD	10	9	Reset
TX2-	12	11	TX2+
Ground	14	13	Ground
TXC-	16	15	TXC+
5V/3.3V	18	17	ENABKL
DDC_DATA	20	19	DDC_CLK

CN3: LCD Backlight Connector



Pin #	Signal Name
1	+12V
2	Backlight Enable
3	Brightness Control
4	Ground

CN5: Output Voltage Connector

Pin #	Signal Name
1	3.3V
2	Ground
3	5V

J9 Speaker Connector

The J9 connector supports 2Watt(RMS)/4 ohm stereo audio power amplifier.

Pin #	Signal Name
1	Audio L+
2	Audio L-
3	Audio R-
4	Audio R+

PCIE1: PCI-E(x1) Slot

PCI1: PCI Slot (supports 2 Master)

This page is intentionally left blank

BIOS Setup

This chapter describes the different settings available in the AMI BIOS that comes with the board. The topics covered in this chapter are as follows:

BIOS Introduction 22
BIOS Setup 22
Main BIOS Setup 23
Advanced Settings 24
Chipset Settings 32
Boot Settings 38
Security Settings 39
Save & Exit Settings 39

BIOS Introduction

The BIOS (Basic Input/Output System) installed in your computer system's ROM supports Intel processors. The BIOS provides critical low-level support for a standard device such as disk drives, serial ports and parallel ports. It also password protection as well as special support for detailed fine-tuning of the chipset controlling the entire system.

BIOS Setup

The BIOS provides a Setup utility program for specifying the system configurations and settings. The BIOS ROM of the system stores the Setup utility. When you turn on the computer, the BIOS is immediately activated. Pressing the <Del> key immediately allows you to enter the Setup utility. If you are a little bit late pressing the <Del> key, POST (Power On Self Test) will continue with its test routines, thus preventing you from invoking the Setup. If you still wish to enter Setup, restart the system by pressing the "Reset" button or simultaneously pressing the <Ctrl>, <Alt> and <Delete> keys. You can also restart by turning the system Off and back On again. The following message will appear on the screen:

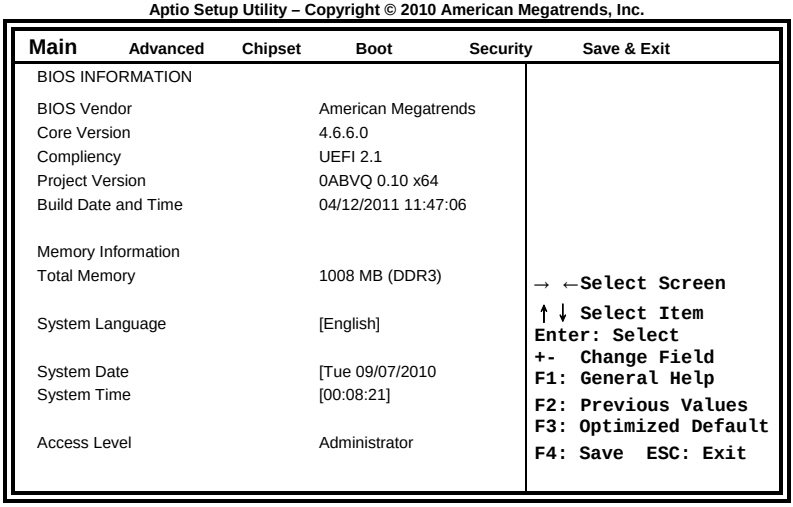
```
Press  <DEL>  to  Enter  Setup
```

In general, you press the arrow keys to highlight items, <Enter> to select, the <PgUp> and <PgDn> keys to change entries, <F1> for help and <Esc> to quit.

When you enter the Setup utility, the Main Menu screen will appear on the screen. The Main Menu allows you to select from various setup functions and exit choices.

Main BIOS Setup

This setup allows you to record some basic hardware configurations in your computer system and set the system clock.



***Note**: If the system cannot boot after making and saving system changes with Setup, the AMI BIOS supports an override to the CMOS settings that resets your system to its default.*

***Warning:** It is strongly recommended that you avoid making any changes to the chipset defaults. These defaults have been carefully chosen by both AMI and your system manufacturer to provide the absolute maximum performance and reliability. Changing the defaults could cause the system to become unstable and crash in some cases.*

System Language

Choose the system default language.

System Date

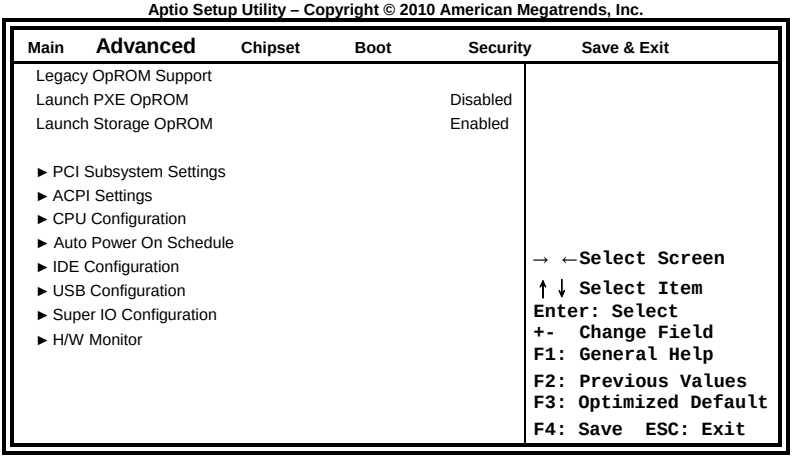
Set the Date. Use Tab to switch between Data elements.

System Time

Set the Time. Use Tab to switch between Time elements.

Advanced Settings

This section allows you to configure and improve your system and allows you to set up some system features according to your preference.



Launch PXE OpROM

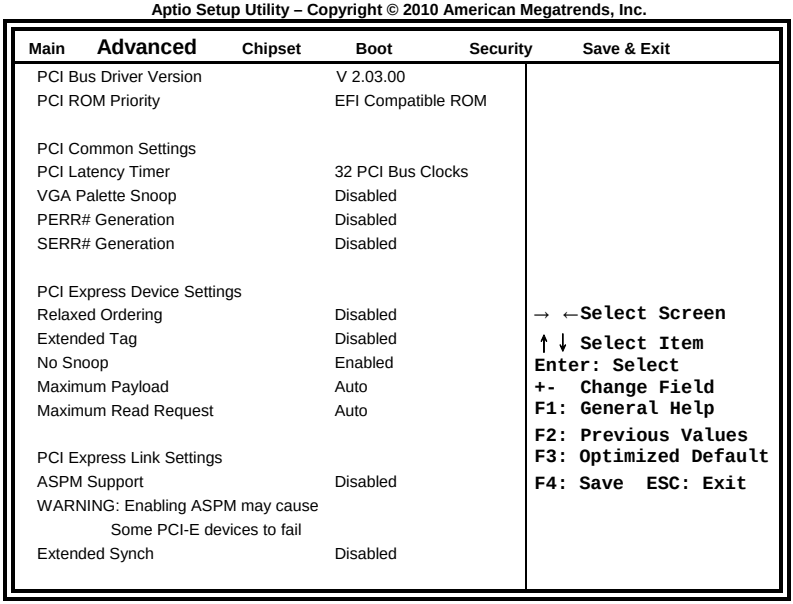
Enable or Disable Boot Option for Legacy Network Devices.

Launch Storage OpROM

Enable or Disable Boot Option for Legacy Mass Storage Devices with Option ROM.

PCI Subsystem Settings

This section allows you to configure the PCI, PCI-X and PCI Express settings.



PCI ROM Priority

In case of multiple Option ROMs (Legacy and EFI Compatible), specifies what PCI Option ROM to launch.

PCI Latency Timer

Value to be programmed into PCI Latency Timer Register.

VGA Palette Snoop

Enables or Disables VGA Palette Registers Snooping.

PERR# Generation

Enables or Disables PCI Device to Generate PERR#.

SERR# Generation

Enables or Disables PCI Device to Generate SERR#.

Relaxed Ordering

Enables or Disables PCI Express Device Relaxed Ordering.

Extended Tag
If ENABLED allows Device to use 8-bit Tag field as a requester.

No Snoop
Enables or Disables PCI Express Device No Snoop option.

Maximum Payload
Set Maximum Payload of PCI Express Device or allow System BIOS to select the value.

Maximum Read Request
Launches (Enabled/Disabled) the boot option for legacy network devices.

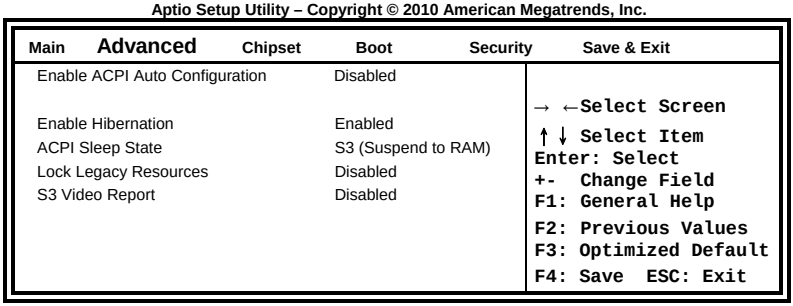
PCI Express Link Settings
Set Maximum Read Request Size of PCI Express Device or allow System BIOS to select the value.

ASPM Support
Set the ASPM Level:
Force L0 – Force all links to L0 State
AUTO – BIOS auto configure
DISABLE – Disables ASPM

Extended Synch
If ENABLED allows generation of Extended Synchronization patterns.

ACPI Settings

This section configures the system ACPI parameters.



Enabled ACPI Auto Configuration

Enables or Disables BIOS ACPI Auto Configuration.

Enable Hibernation

Enables or Disables System ability to Hibernate (OS/S4 Sleep State). This option may be not effective with some OS.

ACPI Sleep State

Select the highest ACPI sleep state the system will enter, when the SUSPEND button is pressed.

Lock legacy Resources

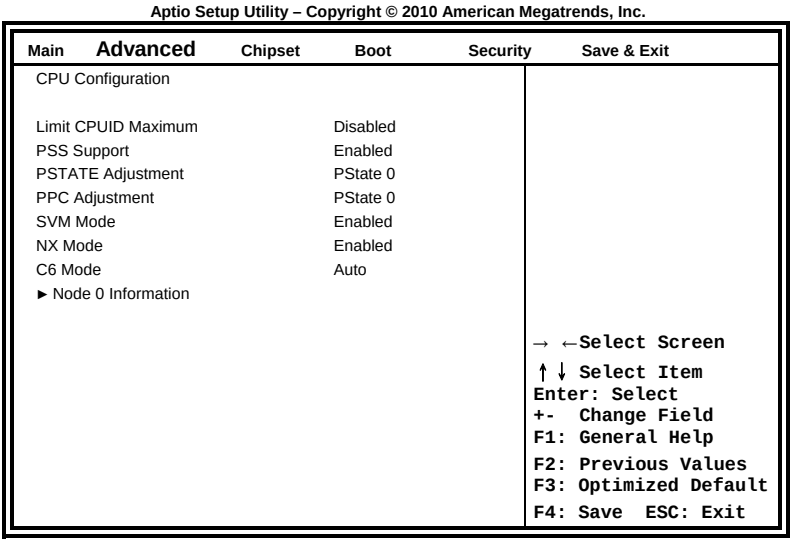
Enabled or Disabled Lock of Legacy Resources

S3 Video Repost

Enabled or Disabled S3 Video Repost.

CPU Configuration

This section shows the CPU configuration parameters.



Limit CPUID Maximum

Disabled for Windows XP.

PSS Support

Enabled /disabled the generation of ACPI_PPC, and _PCT objects.

PSTATE Adjustment

Provide to adjust startup P-state level.

PPC adjustment

Provide to adjust_PPC object.

NX Mode

Enabled/disabled NO-execute page protection Function.

SVM Mode

Enabled/disabled CPU Virtualization.

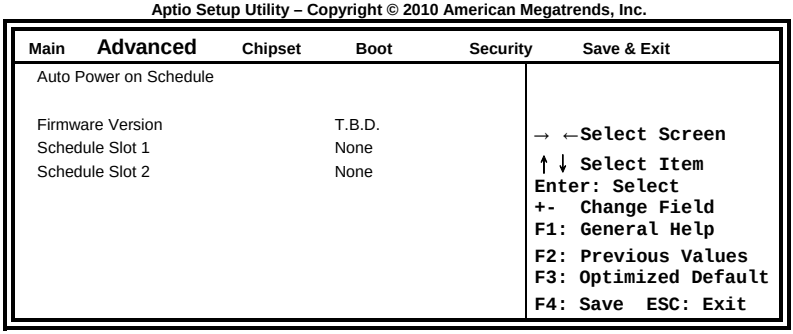
C6 Mode

Enabled/disabled C6.

Node 0 Information

View Memory Information related to Node 0.

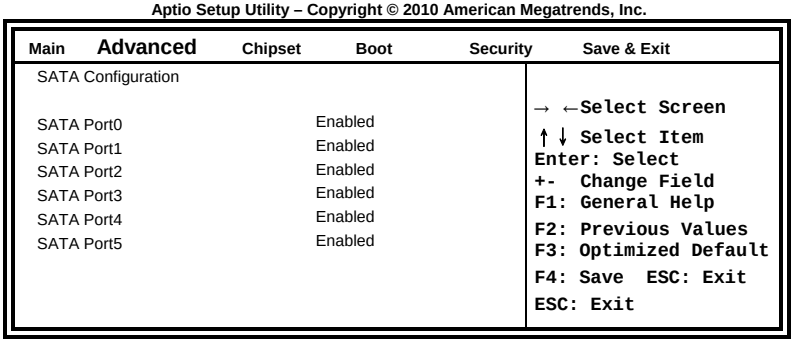
Auto Power On Schedule
This section setups the power on time for the system.



Schedule Slot 1
Setup the hou/minute for sytem power on.

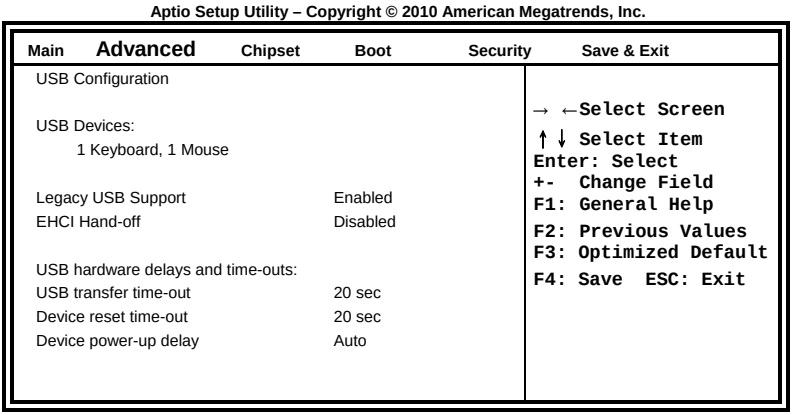
Schedule Slot 2
Setup the hou/minute for sytem power on.

IDE Configuration
This section shows the IDE devices configuration.



Serial-ATA Controller
Enable / Disable Serial ATA Controller.

USB Configuration



Legacy USB Support

Enables Legacy USB support.
AUTO option disables legacy support if no USB devices are connected. DISABLE option will keep USB devices available only for EFI applications.

EHCI Hand-off

Enabled/Disabled. This is a workaround for Oses without EHCI hand-off support. The EHCI ownership change should be claimed by EHCI driver.

USB transfer time-out

The time-out value for Control, Bulk, and Interrupt transfers.

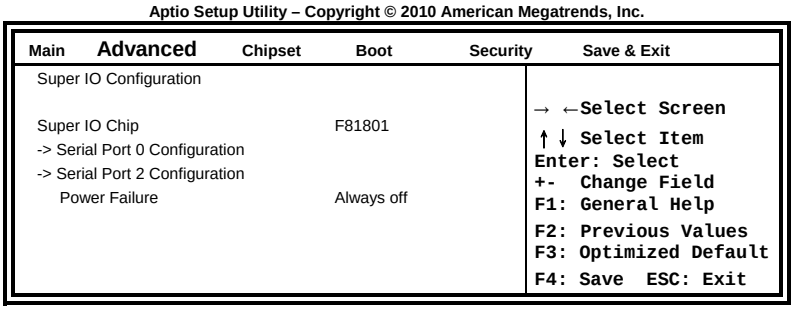
Device reset time-out

USB mass storage device Start Unit command time-out.

Device power-up delay

Maximum time the device will take before it properly reports itself to the Host Controller. 'Auto' users default value: for a Root port it is 100ms, for a Hub port the delay is taken from Hub descriptor.

Super IO Configuration



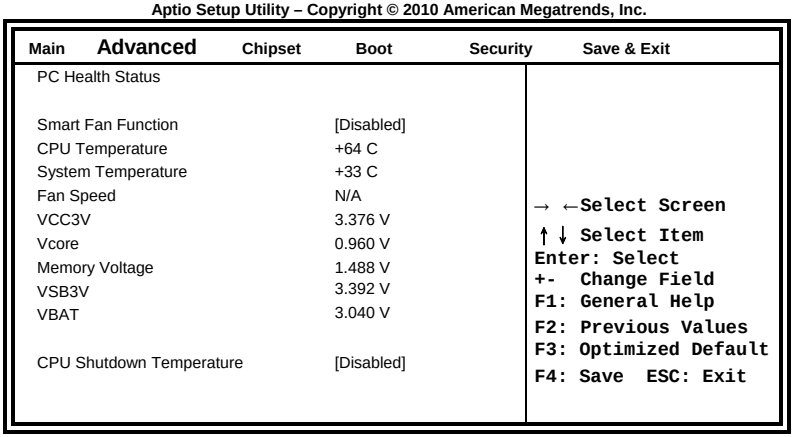
Serial Port 0/1 Configuration

Set Parameters of Serial Port 0/1 (COMA/COMB)

Power Failure

The options: Keep last state, By pass mode, Always on, and Always off.

H/W Monitor



Temperatures/Voltages

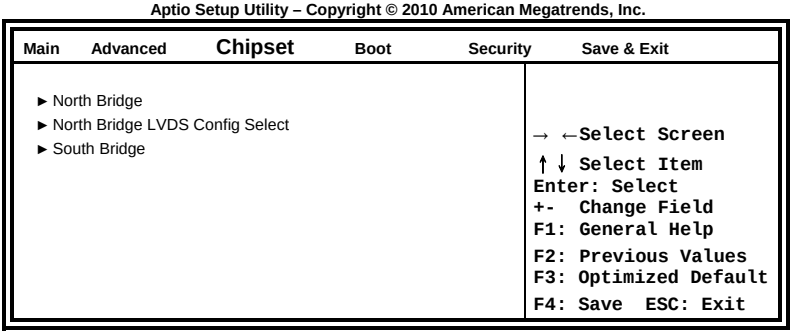
The values are read-only values as monitored by the system and show the PC health status.

CPU Shutdown Temperature

Aside from the Disabled options, this field allows the setting of shutdown temperature from 70C to 95C.

Chipset Settings

This section allows you to configure and improve your system and allows you to set up some system features according to your preference.



North Bridge

This item shows the North Bridge Parameters.

North Bridge LVDS Config Select

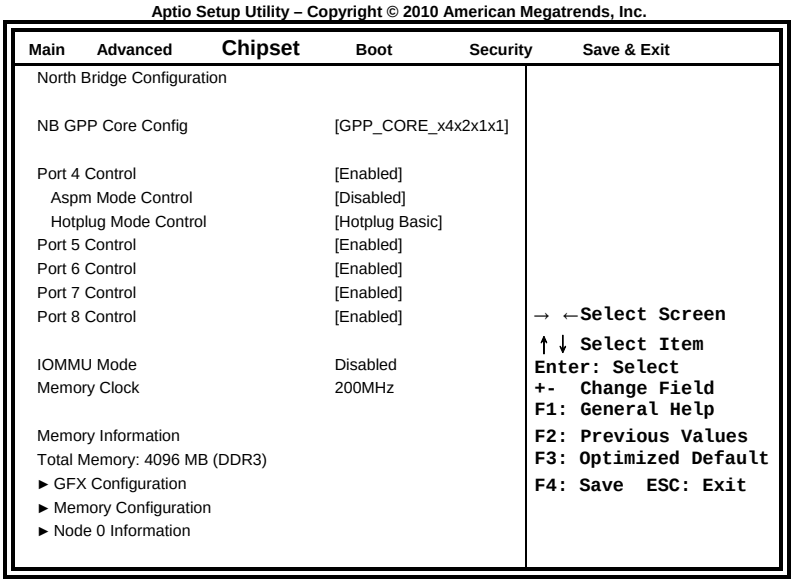
This item shows the Specify INT15 options for LVDS

South Bridge

This item shows the South Bridge Parameters.

North Bridge

This section allows you to configure the North Bridge Chipset.



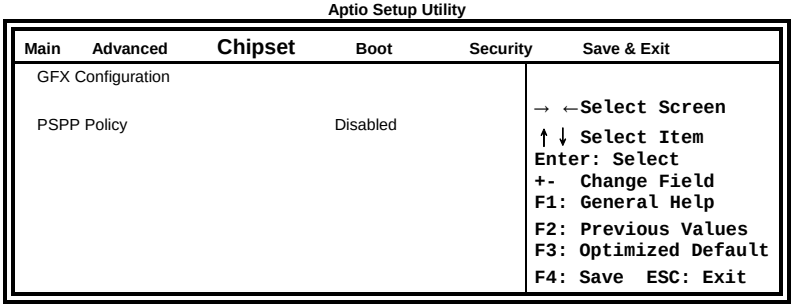
IOMMU Mode

IOMMU is supported on LINUX based systems to convert 32bit I/O to 64bit MMIO.

Memory Clock

This option allows user to select different memory clock.

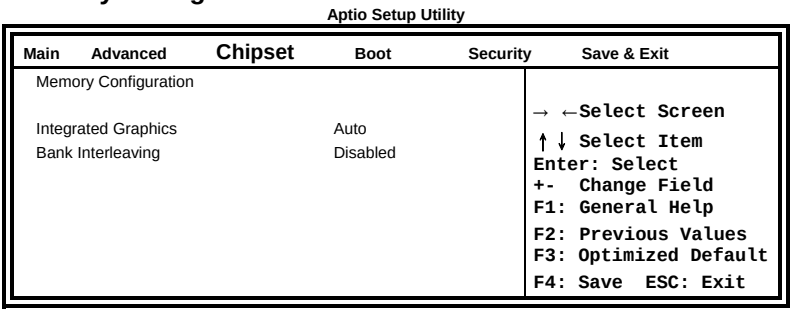
GFX Configuration



PSPP Policy

PCIe speed power policy.

Memory Configuration



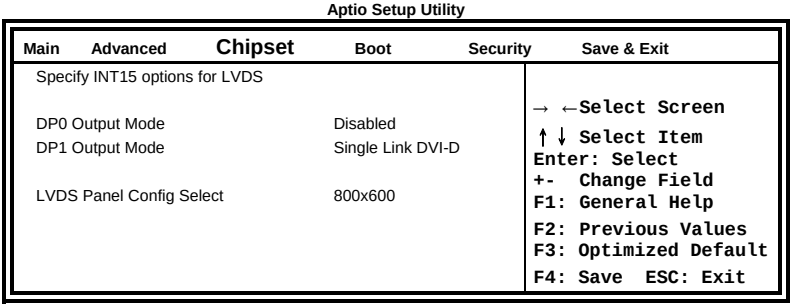
Integrated Graphics

Enable Integrate Graphics controller.

Node 0 Information

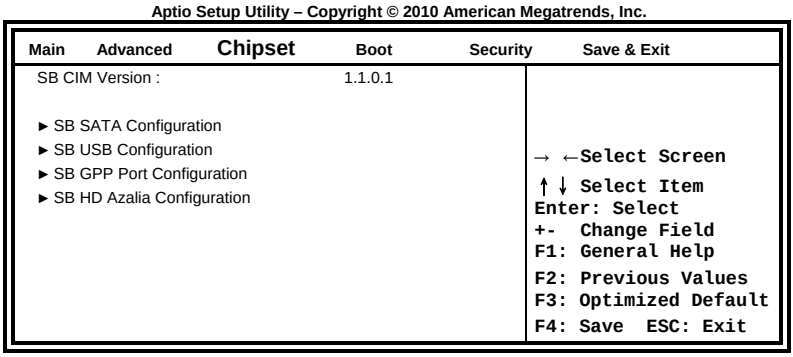
View memory information related to Node 0.

North Bridge LVDS Config Select

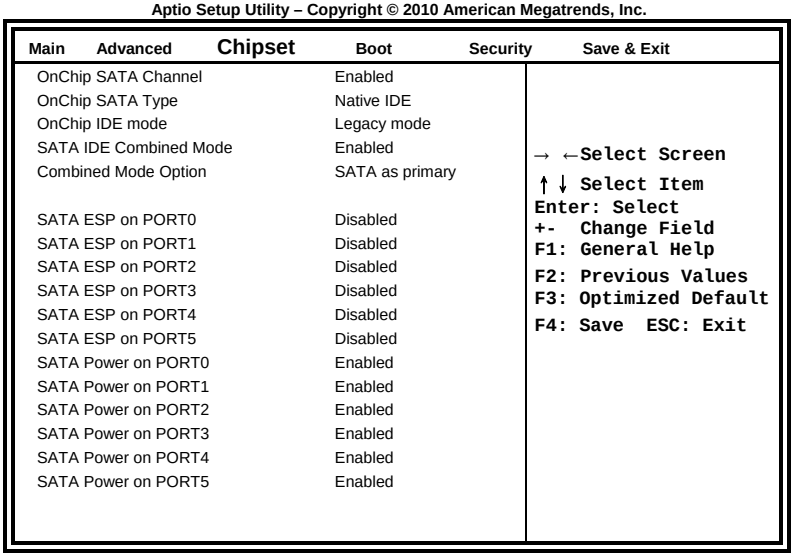


South Bridge

This section allows you to configure the South Bridge Chipset.



SB SATA Configuration



OnChip SATA Type

Native IDE / n RAID / n AHCI / n AHCI / n Legacy IDE / n IDE->AHCI / n HyperFlash

SB USB Configuration

Aptio Setup Utility – Copyright © 2010 American Megatrends, Inc.

Main	Advanced	Chipset	Boot	Security	Save & Exit
OHCI HC (Bus 0 Dev 18 Fn 0)			Enabled		
OHCI HC (Bus 0 Dev 19 Fn 0)			Enabled		
OHCI HC (Bus 0 Dev 22 Fn 0)			Enabled		
OHCI HC (Bus 0 Dev 20 Fn 5)			Enabled		
USB PORT 0			Enabled		
USB PORT 1			Enabled		
USB PORT 2			Enabled		
USB PORT 3			Enabled		
USB PORT 4			Enabled		
USB PORT 5			Enabled		
USB PORT 6			Enabled		
USB PORT 7			Enabled		
USB PORT 8			Enabled		
USB PORT 9			Enabled		
USB PORT 10			Enabled		
USB PORT 11			Enabled		
USB PORT 12			Enabled		
USB PORT 13			Enabled		
USB PORT FL0			Enabled		
USB PORT FL1			Enabled		
USB Device Wakeup From S3 or S4					

→ ←Select Screen
↑↓ Select Item
Enter: Select
+- Change Field
F1: General Help
F2: Previous Values
F3: Optimized Default
F4: Save ESC: Exit

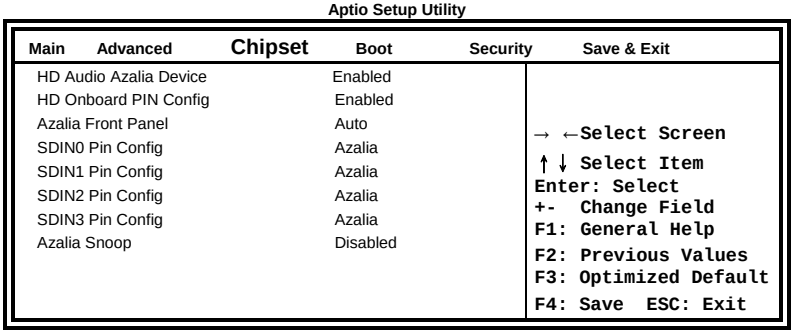
SB GPP Port Configuration

Aptio Setup Utility – Copyright © 2010 American Megatrends, Inc.

Main	Advanced	Chipset	Boot	Security	Save & Exit
SB GPP Function			Enabled		
GPP Port Link Configuration			1:1:1:1 mode		
hide unused GPP port			Enabled		
GPP Link ASPM			Disabled		
NB-SB PHY PLL Power Down			Enabled		
SB GPP PHY PLL Power Down			Enabled		
SB GPP LANE REVERSAL			Disabled		

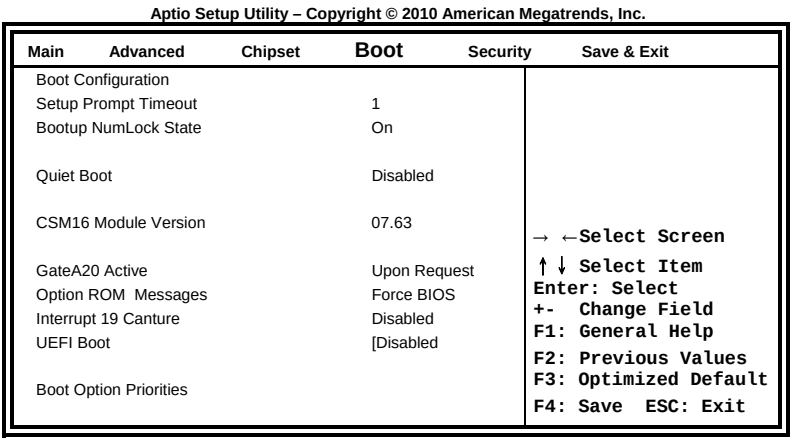
→ ←Select Screen
↑↓ Select Item
Enter: Select
+- Change Field
F1: General Help
F2: Previous Values
F3: Optimized Default
F4: Save ESC: Exit

SB HD Azalia Configuration



Boot Settings

This section allows you to configure the boot settings according to your preference.



Setup Prompt Timeout

Number of seconds to wait for setup activation key.
65535(0xFFFF) means indefinite waiting.

Bootup NumLock State

Select the keyboard NumLock state.

Quiet Boot

Enables/Disables Quiet Boot option.

GateA20 Active

UPON REQUEST – GA20 can be disabled using BIOS services.
ALWAYS – do not allow disabling GA20; this option is useful when any RT code is executed above 1MB.

Option ROM Messages

Set display mode for Option ROM. Options are Force BIOS and Keep Current.

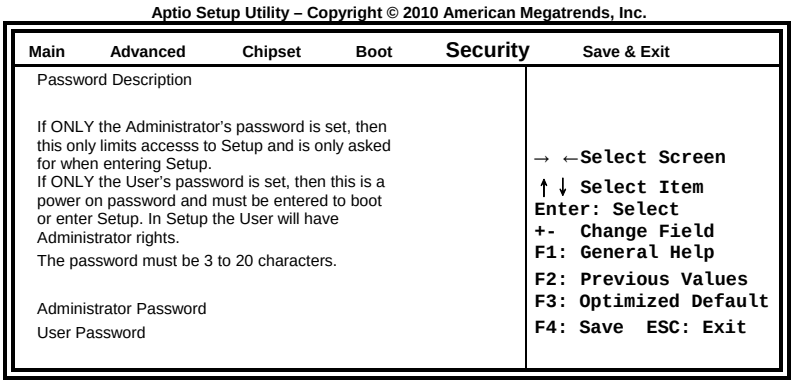
Interrupt 19 Canture

Enable: Allows Option ROMs to trap Int 19.

UEFI Option Priorities

Enables/Disables UEFI boot from disks.

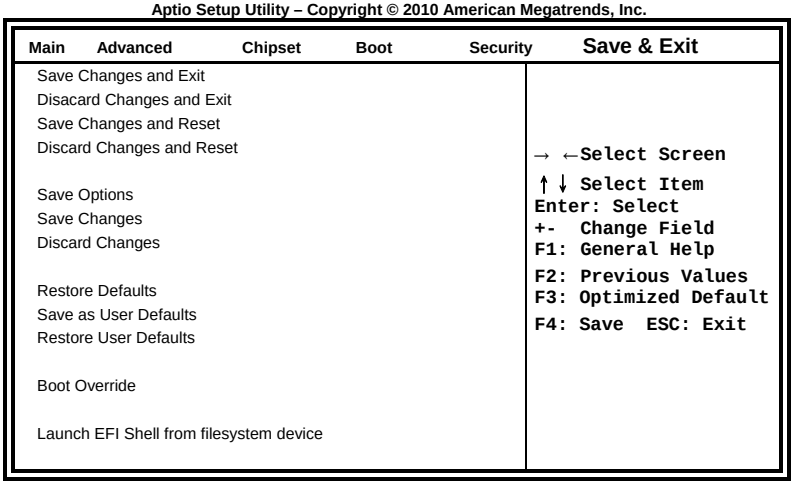
Security Settings



Administrator Password
Set Setup Administrator Password.

User Password
Set User Password.

Save & Exit Settings



Save Changes and Exit
Exit system setup after saving the changes.

Disacard Changes and Exit
Exit system setup without saving any changes.

Save Changes and Reset
Reset the system after saving the changes.

Discard Changes and Reset
Reset system setup without saving any changes.

Save Changes
Save Changes done so far to any of the setup options.

Discard Changes
Discard Changes done so far to any of the setup options.

Restore Defaults
Restore/Load Defaults values for all the setup options.

Save as User Defaults
Save the changes done so far as User Defaults.

Restore User Defaults
Restore the User Defaults to all the setup options.

Boot Override
Pressing ENTER causes the system to enter the OS.

Launch EFI Shell from filesystem device
Attempts to launch EFI Shell application (Shellx64.efi) from one of the available filesystem devices.

Drivers Installation

This section describes the installation procedures for software and drivers. The software and drivers are included with the motherboard. If you find the items missing, please contact the vendor where you made the purchase. The contents of this section include the following:

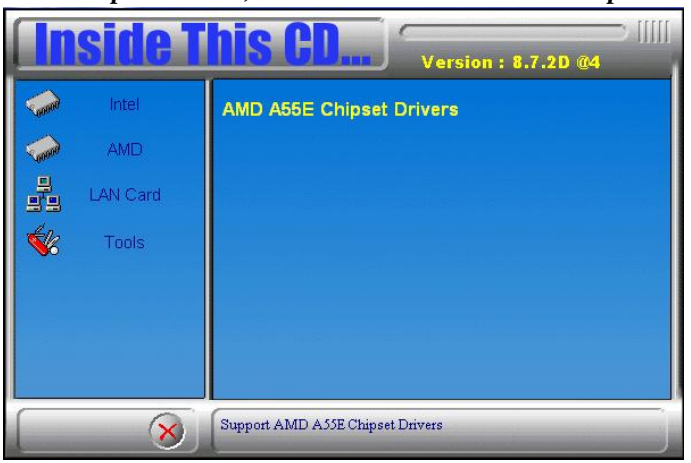
AMD A55E Chipset Family Graphics Driver Installation 42
Realtek High Definition Audio Driver Installation 45
Realtek LAN Controller Drivers Installation 47

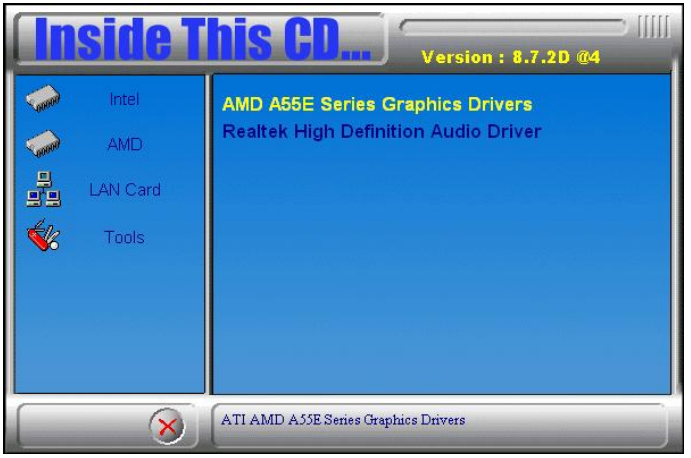
IMPORTANT NOTE:
After installing your Windows operating system, you must install first the Intel Chipset Software Installation Utility before proceeding with the drivers installation.

AMD A55E Chipset Family Graphics Driver Installation

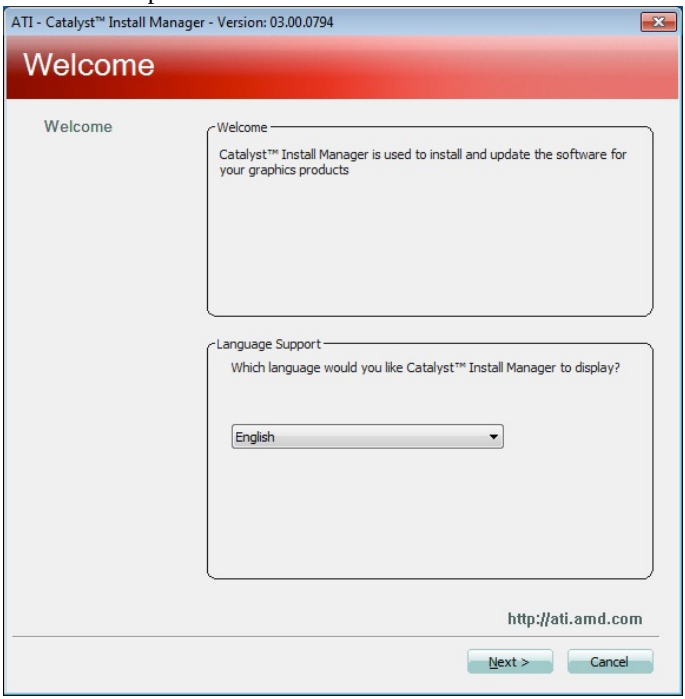
Follow the steps below to install the AMD A55E chipset family graphics drivers.

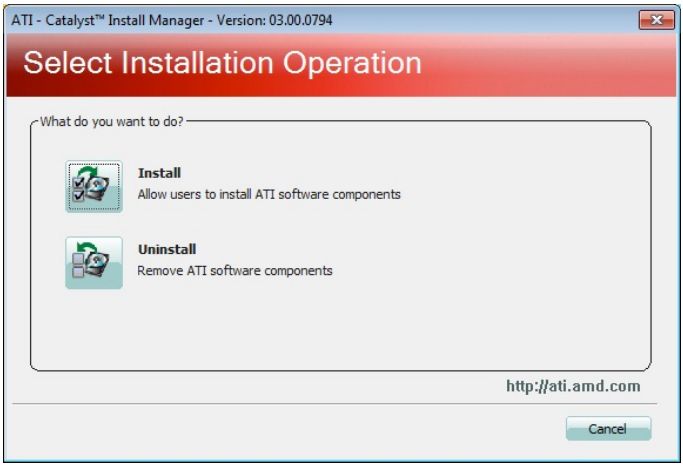
1. Insert the CD that comes with the board. Click ***AMD***, then ***AMD A55E Chipset Drivers***, and then ***AMD A55E Series Graphics Drivers***.



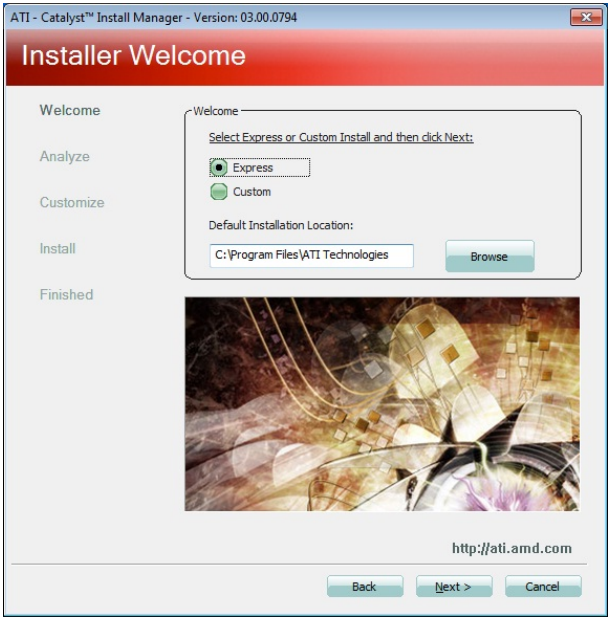


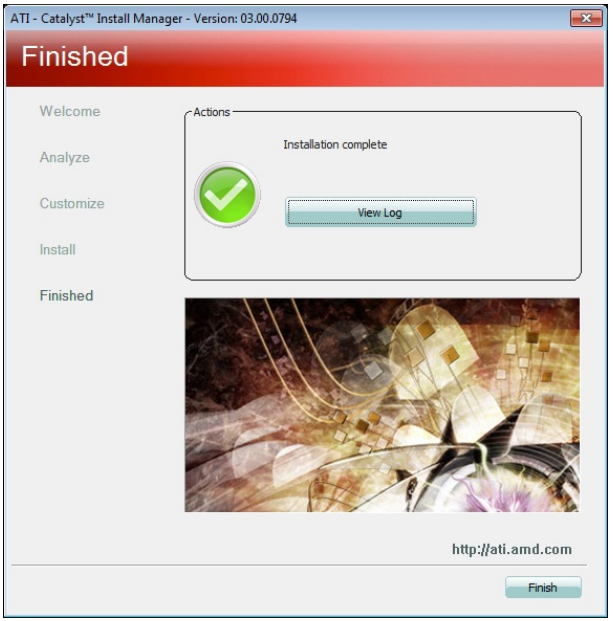
2. When the welcome screen to the ATI – Catalyst™ Install Manager appears, click ***Next.*** Now, click ***Install*** to allow the installation of the software components.





3. Select ***Express*** and click ***Next*** to proceed with the installation. On the following screen, click ***Finish*** to complete the installation process.

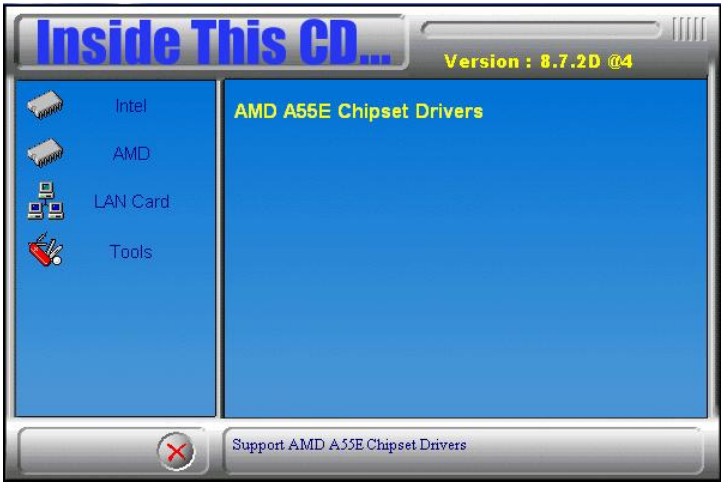


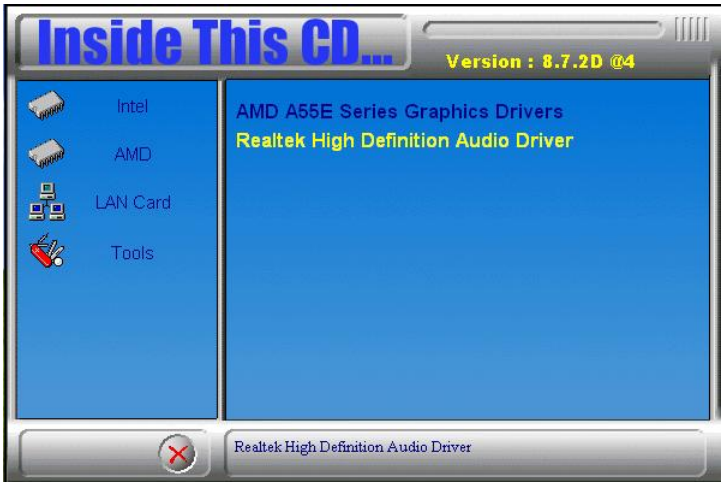


Realtek High Definition Audio Driver Installation

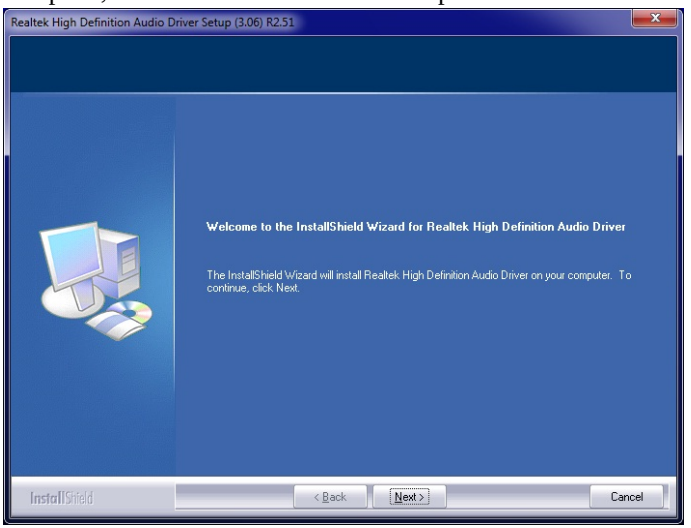
Follow the steps below to install the Realtek HD audio drivers.

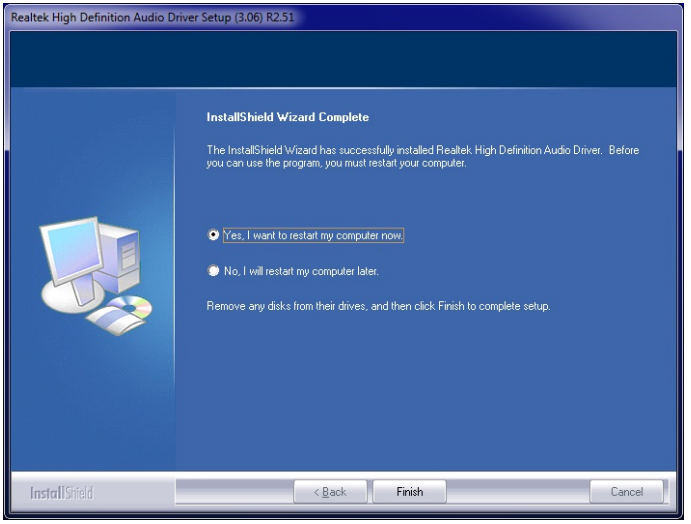
1. Insert the CD that comes with the board. Click ***AMD***, and then ***Realtek High Definition Audio Driver***.





2. When the welcome screen to the Audio Driver Setup appears, click ***Next*** to start the software installation. Once the InstallShield Wizard is complete, click ***Finish*** to restart the computer.





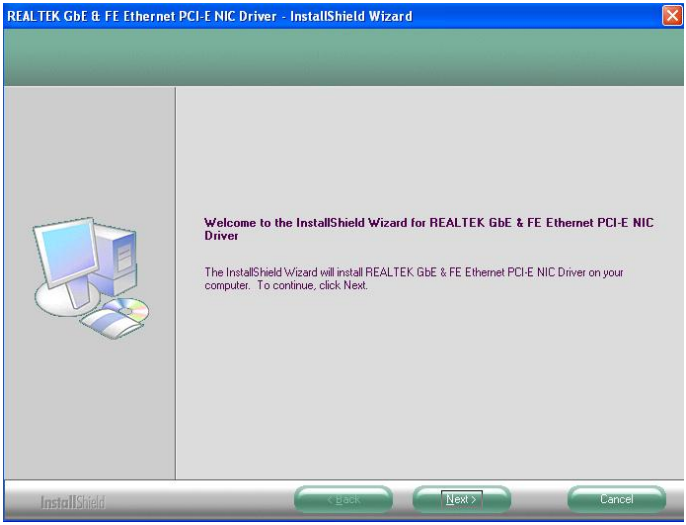
Realtek LAN Controller Drivers Installation

Follow the steps below to install the Realtek LAN Drivers.

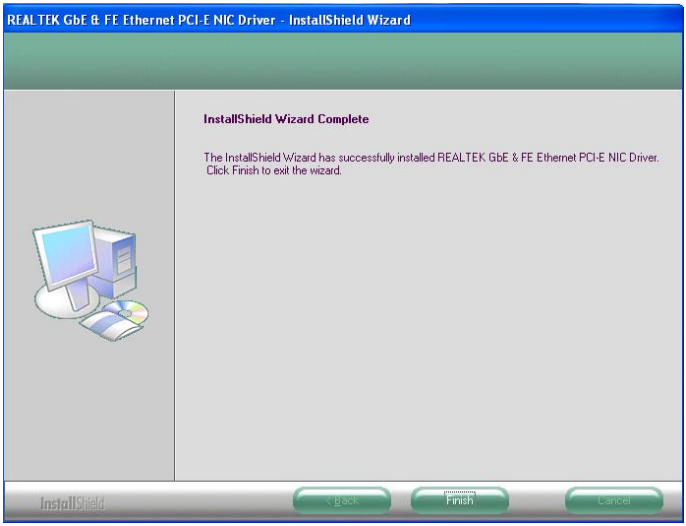
1. Insert the CD that comes with the board. Click ***Intel***, then ***LAN Card,*** and then ***Realtek Lan Controller Drivers***.

2. Click ***Realtek RTL8111E LAN Drivers.***

3.When the welcome screen to InstallShield Wizard appears, click ***Next*** to start the installation



4.When the InstallShieldWizard has finished installing the Realtek LAN drivers, click ***Finish***.



Appendix

A. I/O Port Address Map

Each peripheral device in the system is assigned a set of I/O port addresses which also becomes the identity of the device. The following table lists the I/O port addresses used.

Address	Device Description
000h - 01Fh	DMA Controller #1
020h - 03Fh	Interrupt Controller #1
040h - 05Fh	Timer
060h - 06Fh	Keyboard Controller
070h - 07Fh	Real Time Clock, NMI
080h - 09Fh	DMA Page Register
0A0h - 0BFh	Interrupt Controller #2
0C0h - 0DFh	DMA Controller #2
0F0h	Clear Math Coprocessor Busy Signal
0F1h	Reset Math Coprocessor
1F0h - 1F7h	IDE Interface
278h - 27Fh	Parallel Port #2(LPT2)
2E8h – 2EFh	Serial Port #4(COM4)
2F8h - 2FFh	Serial Port #2(COM2)
2B0h- 2DFh	Graphics adapter Controller
360h - 36Fh	Network Ports
3B0h - 3BFh	Monochrome & Printer adapter
3C0h - 3CFh	EGA adapter
3D0h - 3DFh	CGA adapter
3E8h – 3EFh	Serial Port #3(COM3)
3F8h - 3FFh	Serial Port #1(COM1)

B. Interrupt Request Lines (IRQ)

Peripheral devices use interrupt request lines to notify CPU for the service required. The following table shows the IRQ used by the devices on board.

Level	Function
IRQ0	System Timer Output
IRQ1	Keyboard
IRQ2	Interrupt Cascade
IRQ3	Serial Port #2
IRQ4	Serial Port #1
IRQ5	Reserved
IRQ6	Reserved
IRQ7	Reserved
IRQ8	Real Time Clock
IRQ9	Reserved
IRQ10	Serial Port #3
IRQ11	Serial Port #4
IRQ12	PS/2 Mouse
IRQ13	80287
IRQ14	Primary IDE
IRQ15	Secondary IDE

C. Watchdog Timer Configuration

The WDT is used to generate a variety of output signals after a user programmable count. The WDT is suitable for use in the prevention of system lock-up, such as when software becomes trapped in a deadlock. Under these sorts of circumstances, the timer will count to zero and the selected outputs will be driven. Under normal circumstance, the user will restart the WDT at regular intervals before the timer counts to zero.

SAMPLE CODE:

```
//---------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//---------------------------------------------------------------------------
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include "F81865.H"
//---------------------------------------------------------------------------
int main (int argc, char *argv[]);
void EnableWDT(int);
void DisableWDT(void);
//---------------------------------------------------------------------------
int main (int argc, char *argv[])
{
        unsigned char bBuf;
        unsigned char bTime;
        char **endptr;

        char SIO;

        printf("Fintek 81865 watch dog program\n");

        SIO = Init_F81865();
        if (SIO == 0)
        {
                printf("Can not detect Fintek 81865, program abort.\n");
                return(1);
        }//if (SIO == 0)

        if (argc != 2)
        {
                printf(" Parameter incorrect!!\n");
                return (1);
        }

        bTime = strtol (argv[1], endptr, 10);
        printf("System will reset after %d seconds\n", bTime);

        if (bTime)
        {       EnableWDT(bTime); }
        else
        {       DisableWDT();       }

        return 0;
}
```

```
//-----------------------------------------------------------------------------
void EnableWDT(int interval)
{
	unsigned char bBuf;

	bBuf = Get_F81865_Reg(0x2B);
	bBuf &= (~0x20);
	Set_F81865_Reg(0x2B, bBuf);			//Enable WDTO

	Set_F81865_LD(0x07);				//switch to logic device 7
	Set_F81865_Reg(0x30, 0x01);			//enable timer

	bBuf = Get_F81865_Reg(0xF5);
	bBuf &= (~0x0F);
	bBuf |= 0x52;
	Set_F81865_Reg(0xF5, bBuf);			//count mode is second

	Set_F81865_Reg(0xF6, interval);		//set timer

	bBuf = Get_F81865_Reg(0xFA);
	bBuf |= 0x01;
	Set_F81865_Reg(0xFA, bBuf);			//enable WDTO output

	bBuf = Get_F81865_Reg(0xF5);
	bBuf |= 0x20;
	Set_F81865_Reg(0xF5, bBuf);			//start counting
}
//-----------------------------------------------------------------------------
void DisableWDT(void)
{
	unsigned char bBuf;

	Set_F81865_LD(0x07);				//switch to logic device 7

	bBuf = Get_F81865_Reg(0xFA);
	bBuf &= ~0x01;
	Set_F81865_Reg(0xFA, bBuf);			//disable WDTO output

	bBuf = Get_F81865_Reg(0xF5);
	bBuf &= ~0x20;
	bBuf |= 0x40;
	Set_F81865_Reg(0xF5, bBuf);			//disable WDT
}
//-----------------------------------------------------------------------------
```

```
//---------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//---------------------------------------------------------------------------
#include "F81865.H"
#include <dos.h>
//---------------------------------------------------------------------------
unsigned int F81865_BASE;
void Unlock_F81865 (void);
void Lock_F81865 (void);
//---------------------------------------------------------------------------
unsigned int Init_F81865(void)
{
	unsigned int result;
	unsigned char ucDid;

	F81865_BASE = 0x4E;
	result = F81865_BASE;

	ucDid = Get_F81865_Reg(0x20);
	if (ucDid == 0x07)                                  //Fintek 81865
	{	goto Init_Finish;	}

	F81865_BASE = 0x2E;
	result = F81865_BASE;

	ucDid = Get_F81865_Reg(0x20);
	if (ucDid == 0x07)                                  //Fintek 81865
	{	goto Init_Finish;	}

	F81865_BASE = 0x00;
	result = F81865_BASE;

Init_Finish:
	return (result);
}
//---------------------------------------------------------------------------
void Unlock_F81865 (void)
{
	outportb(F81865_INDEX_PORT, F81865_UNLOCK);
	outportb(F81865_INDEX_PORT, F81865_UNLOCK);
}
//---------------------------------------------------------------------------
void Lock_F81865 (void)
{
	outportb(F81865_INDEX_PORT, F81865_LOCK);
}
//---------------------------------------------------------------------------
void Set_F81865_LD( unsigned char LD)
{
	Unlock_F81865();
	outportb(F81865_INDEX_PORT, F81865_REG_LD);
	outportb(F81865_DATA_PORT, LD);
	Lock_F81865();
}
//---------------------------------------------------------------------------
void Set_F81865_Reg( unsigned char REG, unsigned char DATA)
{
	Unlock_F81865();
	outportb(F81865_INDEX_PORT, REG);
	outportb(F81865_DATA_PORT, DATA);
	Lock_F81865();
}
//---------------------------------------------------------------------------
unsigned char Get_F81865_Reg(unsigned char REG)
{
	unsigned char Result;
	Unlock_F81865();
```

```
        outportb(F81865_INDEX_PORT, REG);
        Result = inportb(F81865_DATA_PORT);
        Lock_F81865();
        return Result;
}
//----------------------------------------------------------------------------
```

```
//----------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//----------------------------------------------------------------------------
#ifndef __F81865_H
#define __F81865_H                     1
//----------------------------------------------------------------------------
#define         F81865_INDEX_PORT               (F81865_BASE)
#define         F81865_DATA_PORT                (F81865_BASE+1)
//----------------------------------------------------------------------------
#define         F81865_REG_LD                   0x07
//----------------------------------------------------------------------------
#define F81865_UNLOCK                   0x87
#define         F81865_LOCK                             0xAA
//----------------------------------------------------------------------------
unsigned int Init_F81865(void);
void Set_F81865_LD( unsigned char);
void Set_F81865_Reg( unsigned char, unsigned char);
unsigned char Get_F81865_Reg( unsigned char);
//----------------------------------------------------------------------------
#endif  //__F81865_H
```