IB827

Intel® Atom 945GSE
PICMG 1.0 CPU CARD

USER'S MANUAL

Version 1.0

Acknowledgments

Award is a registered trademark of Award Software International, Inc.

PS/2 is a trademark of International Business Machines Corporation.

Intel® and **Atom**™ are registered trademark and trademark of Intel Corporation.

Microsoft Windows is a registered trademark of Microsoft Corporation.

Winbond is a registered trademark of Winbond Electronics Corporation.

All other product names or trademarks are properties of their respective owners.

Table of Contents

This page is intentionally left blank.

Introduction

Product Description

The IB827 is a PICMG 1.0 full-size single board computer that is configured with the Intel Atom processor N270 at 1.6GHz, FSB533 and the Mobile Intel 945GSE Express Chipset with the ICH7M.

This full-size single board computer provides greater flexibility for developers of embedded computing solutions. It is ideally suited for communication and compact designs as in telephone exchange and applications in automation, industrial control, data acquisition, client and other industrial PC applications.

IB827 features the Intel's Graphics Media Accelerator 950 core, making it compatible with Windows Vista Premium, and Chrontel CH7308 for 24-bit dual LVDS channel. Two DDR2 DIMM sockets on board implement up to 2GB of system memory. Dual Gigabit LAN is also available.

Advanced connectivity and expansion interfaces are provided by one IDE, two high-speed SATA-II, four serial ports, eight USB 2.0 ports and a CF socket. Dimensions of the board are 122mm x 338mm.
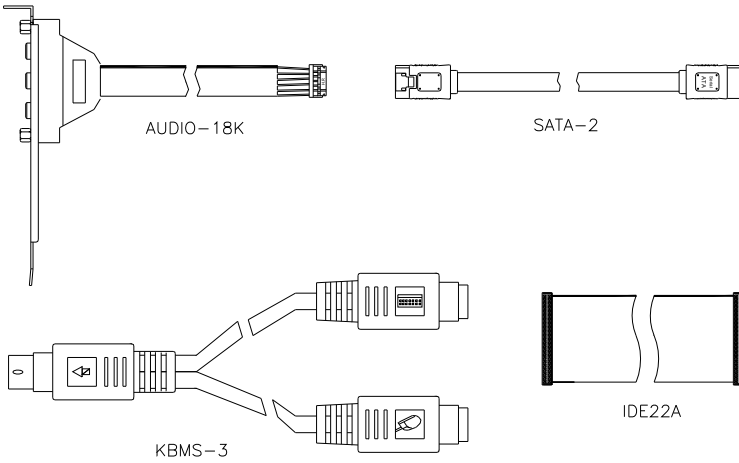
Checklist

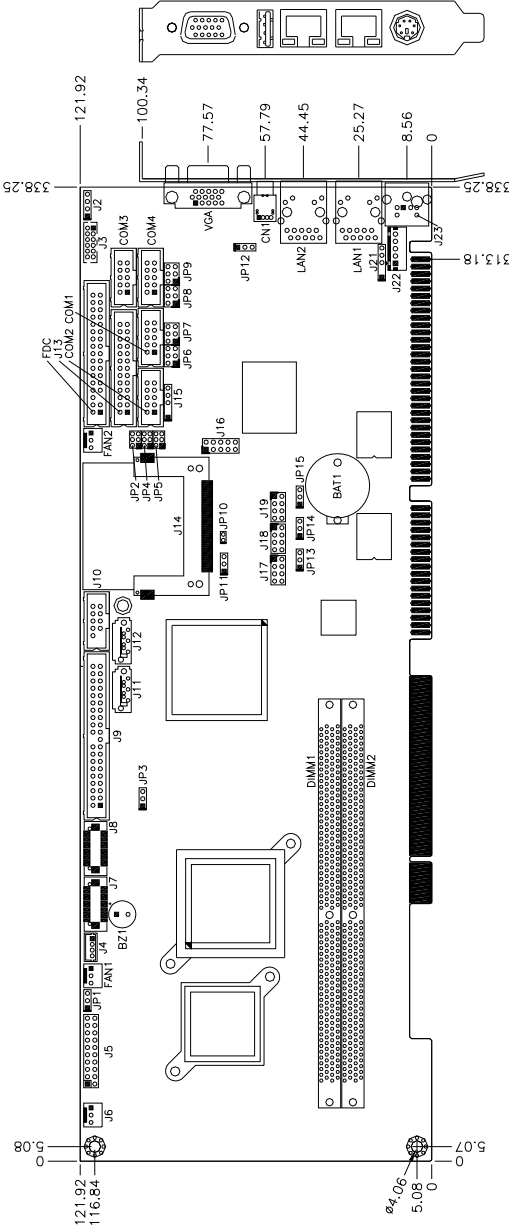Your IB827 package should include the items listed below.

- The IB827 CPU card
- This User's Manual
- 1 CD containing chipset drivers and flash memory utility
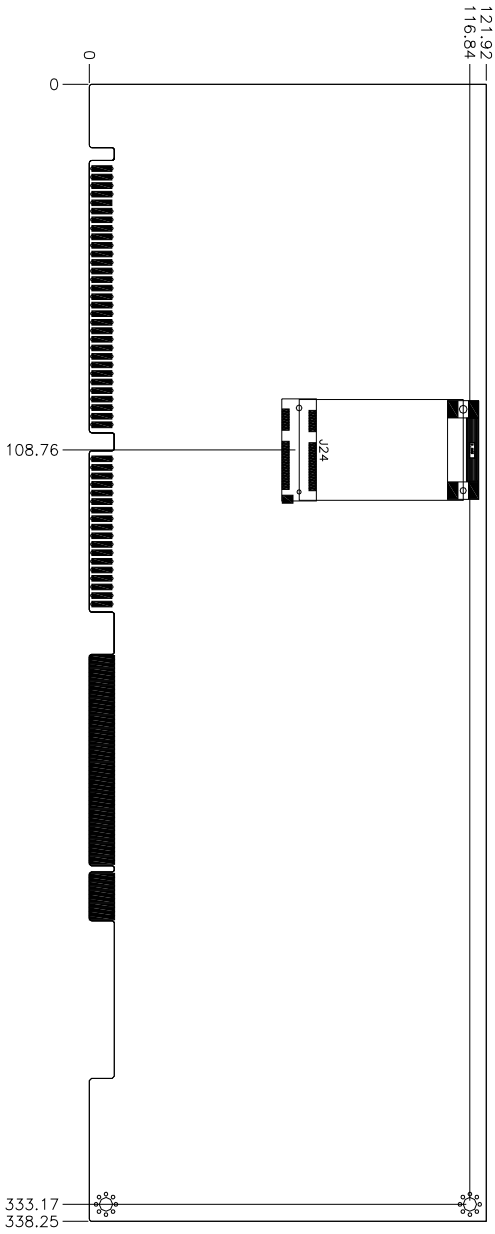- Cables include: AUDIO-18K, KBMS-3, IDE22A, SATA-2

IB827 Specifications

Product Name	**IB827**
Form Factor	Full Size CPU Card (PICMG 1.0)
CPU Type	Intel® Atom™ Processor N270 (45nm Technology)
CPU Speed	1.6 GHz (512K Cache, 2.5W)
CPU FSB	533 MHz FSB
Green /APM	APM1.2
BIOS	Award BIOS, support ACPI Function
CPU Socket	PBGA437
Chipset	Mobile Intel® 945GSE + ICH7M Chipset 945GSE GMCH: QG82945GSE, 27mm x 27mm, 998-pin FC-BGA (3.5W) ICH7M: NH82801GBM, 31mm x 31mm, 652-pin mBGA , (1.5W)
Memory	DDR 2 533MHz DIMM x **2** (w/o ECC), Max. 2GB , Supports single channel. w/o ECC function
VGA	Intel 945GSE GMCH integrated graphics device (GMA950) Supports Direct X 9.1
DVI	N/A
LVDS	Chrontel CH7308B x1 for 24-bit single/dual channel LVDS (N/A for 827E)
LAN	Realtek 8111C PCI Express Gigabit LAN controller x 2 (1 port for 827E)
USB	ICH7M built-in USB 2.0 host controller, support 8 ports
Serial ATA Ports	ICH7M built-in SATA controller, supports 2 ports
Parallel IDE	ICH7M built-in one channel Ultra DMA 33/66/100, CF Type II
PCI-to-ISA bridge	ITE IT8888G x1 for high drive ISA bus
Audio	Intel ICH7M built-in audio controller w/ Realtek ALC662 Codec supports 5.1 CH audio (Line-out, Line-in & MIC)
LPC I/O	**Winbond 83627EHG + Fintek 81216 (N/A for IB827E)** COM1(RS232), COM2(RS232/422/485), COM 3 & 4(support power via pin-9) IrDAx1, Parallel x 1, Floppy Hardware monitor (3 thermal inputs, 4 voltage monitor inputs, VID0-4 & 2 Fan Headers)
Digital IO	4 in & 4 out
Keyboard/Mouse Connector	Supports PS/2 Keyboard/Mouse
Expansion Slots	Mini PCI-express socket x1 for Wireless LAN or other module **(Solder side) N/A for IB827E**
Edge Connector	PS/2 Connector x1 for keyboard/mouse DB15 x1 for VGA, RJ45 x2 for LAN 1, 2 (N/A for 827E) USB x1
On Board Header/Connector	40-pin box header for IDE, 2x DF13-20 header for LVDS, 34-pin box header for FDD, 26-pin box header for LPT, 10-pin box header x4 for COM1/COM2/COM3/COM4 ***Note: (There is no COM3/COM4 for IB827E)*** 3-pin header for CPU fan, 3-pin header for system fan, 3x 8-pin header USB1~USB6 5-pin header for IrDA SATA ports x2, CF connector x1
Watchdog Timer	Yes (256 segments, 0, 1, 2…255 sec/min)
System Voltage	+5V, +12V, -12V & 5VSB
RoHS	Yes
Board Size	338mm x 122mm

Board Dimensions

121.92
116.84

0
0

108.76

J24

333.17
338.25

Installations

This section provides information on how to use the jumpers and connectors on the IB827 in order to set up a workable system. The topics covered are:

Installing the Memory

The IB827 board supports two DDR2 memory sockets to support up to 2GB memory, DDR2 400/533 (w/o ECC function).

Installing and Removing Memory Modules

To install the DDR2 modules, locate the memory slot on the board and perform the following steps:

1. Hold the DDR2 module so that the key of the DDR2 module aligns with that on the memory slot. Insert the module into the socket at a slight angle. Note that the socket and module are both keyed, which means that the module can be installed only in one direction.

2. To seat the memory module into the socket, apply firm and even pressure to each end of the module until you feel it slip down into the socket.

3. With the module properly seated in the socket, push the module downward. Continue pressing downward until the clips at each end lock into position.

4. To remove the DDR2 module, press the clips with both hands.

Setting the Jumpers

Jumpers are used on IB827 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 jumpers on IB827 and their respective functions.

Jumper Locations on IB827
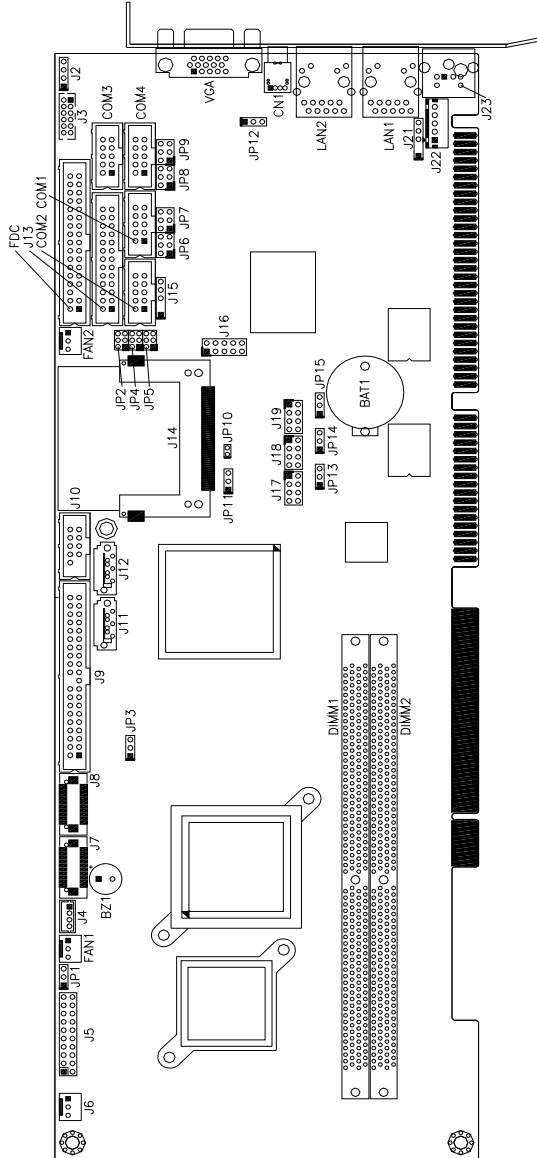
JP1: ATX or AT Power Selection

JP1	ATX Power
□ □ □ 1 2 3	ATX
□ □ □ 1 2 3	AT

JP3: LCD Panel Power Selection

JP3	LCD Panel Power
□ □ □ 1 2 3	3.3V
□ □ □ 1 2 3	5V

JP10: CompactFlash Slave/Master Selection

JP10	CF Setting
□ □ Short	Master
□ □ Open	Slave

JP11: Clear CMOS Setting

JP11	Setting
□ □ □ 1 2 3	Normal
□ □ □ 1 2 3	Clear CMOS

JP2, JP4, JP5: RS232/422/485 (COM2) Selection
COM1 is fixed for RS-232 use only. COM2 is selectable for RS232, RS-422 and RS-485.The following describes the settings for COM2.

```
2  4  6
□  □  □
□  □  □
1  3  5
```
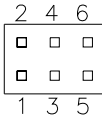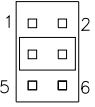
COM2 Function	RS-232	RS-422	RS-485
Jumper Setting (pin closed)	JP5: 1-2 JP2: 3-5 & 4-6 JP4: 3-5 & 4-6	JP5: 3-4 JP2: 1-3 & 2-4 JP4: 1-3 & 2-4	JP5: 5-6 JP2: 1-3 & 2-4 JP4: 1-3 & 2-4
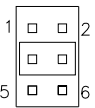
COM2 is jumper selectable for RS-232, RS-422 and RS-485.

Pin #	Signal Name		
	RS-232	**R2-422**	**RS-485**
1	DCD	TX-	DATA-
2	RX	TX+	DATA+
3	TX	RX+	NC
4	DTR	RX-	NC
5	Ground	Ground	Ground
6	DSR	NC	NC
7	RTS	NC	NC
8	CTS	NC	NC
9	RI	NC	NC
10	NC	NC	NC
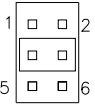
JP6: COM2 RS232 +5V/+12V Power Setting

JP6	Setting	Function
	Pin 1-2 Short/Closed	+12V
	Pin 3-4 Short/Closed	Normal
	Pin 5-6 Short/Closed	+5V
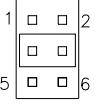
JP7: COM1 RS232 +5V/+12V Power Setting

JP7	Setting	Function
	Pin 1-2 Short/Closed	+12V
	Pin 3-4 Short/Closed	Normal
	Pin 5-6 Short/Closed	+5V

JP8: COM3 RS232 +5V/+12V Power Setting
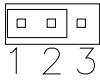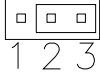
JP8	Setting	Function
	Pin 1-2 Short/Closed	+12V
	Pin 3-4 Short/Closed	Normal
	Pin 5-6 Short/Closed	+5V

JP9: COM4 RS232 +5V/+12V Power Setting
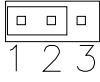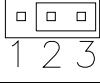
JP9	Setting	Function
	Pin 1-2 Short/Closed	+12V
	Pin 3-4 Short/Closed	Normal
	Pin 5-6 Short/Closed	+5V
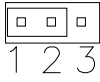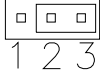
JP12: USB0, USB1 (CN1, J24) Power Setting

JP12	Setting	Function
□ □ □ 1 2 3	Pin 1-2 Short/Closed	+5VSB
□ □ □ 1 2 3	Pin 2-3 Short/Closed	+5V
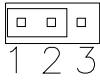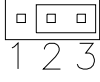
JP13: USB2, USB3 (J17) Power Setting

JP13	Setting	Function
□ □ □ 1 2 3	Pin 1-2 Short/Closed	+5VSB
□ □ □ 1 2 3	Pin 2-3 Short/Closed	+5V

JP14: USB4, USB5 (J18) Power Setting

JP14	Setting	Function
□ □ □ 1 2 3	Pin 1-2 Short/Closed	+5VSB
□ □ □ 1 2 3	Pin 2-3 Short/Closed	+5V

JP15: USB6, USB7 (J19) Power Setting

JP15	Setting	Function
□ □ □ 1 2 3	Pin 1-2 Short/Closed	+5VSB
□ □ □ 1 2 3	Pin 2-3 Short/Closed	+5V

This page is intentionally left blank.

Connectors on IB827

J9: IDE Connector

Signal Name	Pin #	Pin #	Signal Name
Reset IDE	1	2	Ground
Host data 7	3	4	Host data 8
Host data 6	5	6	Host data 9
Host data 5	7	8	Host data 10
Host data 4	9	10	Host data 11
Host data 3	11	12	Host data 12
Host data 2	13	14	Host data 13
Host data 1	15	16	Host data 14
Host data 0	17	18	Host data 15
Ground	19	20	Key
DRQ0	21	22	Ground
Host IOW	23	24	Ground
Host IOR	25	26	Ground
IOCHRDY	27	28	Host ALE
DACK0	29	30	Ground
IRQ14	31	32	No connect
Address 1	33	34	No connect
Address 0	35	36	Address 2
Chip select 0	37	38	Chip select 1
Activity	39	40	Ground
Vcc	41	42	Vcc
Ground	43	44	N.C.

FAN1: CPU Fan Power Connector
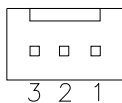
Pin #	Signal Name
1	Ground
2	+12V
3	Rotation detection

FAN2: System Fan Power Connector
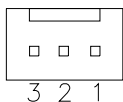
Pin #	Signal Name
1	Ground
2	+12V
3	Rotation detection

FDC: Floppy Drive Connector
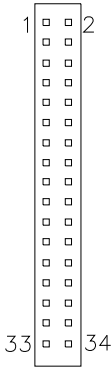
Signal Name	Pin #	Pin #	Signal Name
Ground	1	2	RM/LC
Ground	3	4	No connect
Ground	5	6	No connect
Ground	7	8	Index
Ground	9	10	Motor enable 0
Ground	11	12	Drive select 1
Ground	13	14	Drive select 0
Ground	15	16	Motor enable 1
Ground	17	18	Direction
Ground	19	20	Step
Ground	21	22	Write data
Ground	23	24	Write gate
Ground	25	26	Track 00
Ground	27	28	Write protect
Ground	29	30	Read data
Ground	31	32	Side 1 select
Ground	33	34	Diskette change

COM1~COM4: COM1~ COM4 Serial Ports
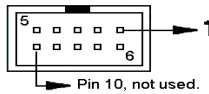
Pin 10, not used.

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

VGA: VGA Connector (D-sub, 15-pin)
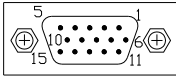
Signal Name	Pin #	Pin #	Signal Name
Red	1	2	Green
Blue	3	4	N.C.
GND	5	6	GND
GND	7	8	GND
VCC	9	10	GND
N.C.	11	12	DDCDATA
HSYNC	13	14	VSYNC
DDCCLK	15		

J2: CD-In Audio Connector

Pin #	Signal Name
1	CD Audio L
2	Ground
3	Ground
4	CD Audio R

J3: Audio Connector (DF11 type)
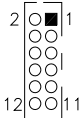
Signal Name	Pin #	Pin #	Signal Name
LINEOUT_R	2	1	LINEOUT_L
Ground	4	3	JD_FRONT
LINEIN_R	6	5	LINEIN
Ground	8	7	JD_LINEIN
MIC-In	10	9	MIC_L
Ground	12	11	JD_MIC1

J4: LCD Backlight Connector (DC type)
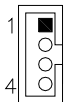
Pin #	Signal Name
1	+12V
2	Backlight Enable
3	*Backlight Adj (DC type)
4	Ground

* LCD backlight can be controlled by the OS.

J5: System Function Connector

J5 provides connectors for system indicators that provide light indication of the computer activities and switches to change the computer status.
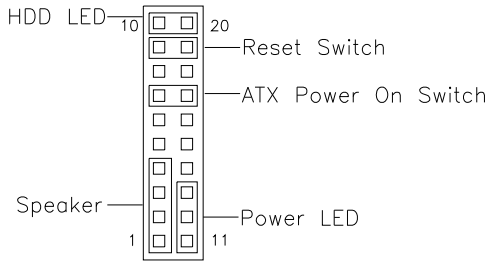
Speaker: Pins 1 - 4

This connector provides an interface to a speaker for audio tone generation. An 8-ohm speaker is recommended.
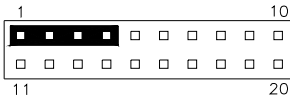
Pin #	Signal Name
1	Speaker out
2	No connect
3	Ground
4	+5V

Power LED: Pins 11 - 13

The power LED indicates the status of the main power switch.
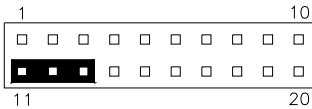
Pin #	Signal Name
11	Power LED
12	No connect
13	Ground

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.
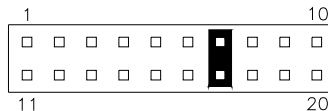
Reset Switch: Pins 9 and 19

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.

```
 1                    10
 □ □ □ □ □ □ □ □ ■ □
 □ □ □ □ □ □ □ □ ■ □
 11                   20
```
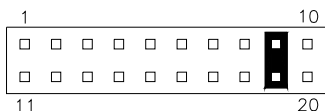
Hard Disk Drive LED Connector: Pins 10 and 20

This connector connects to the hard drive activity LED on control panel. This LED will flash when the HDD is being accessed.

```
 1                    10
 □ □ □ □ □ □ □ □ □ ■
 □ □ □ □ □ □ □ □ □ ■
 11                   20
```
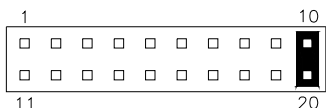
Pin #	Signal Name
10	HDD Active
20	5V

J7, J8: LVDS Connectors (DF13 type)
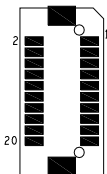
Signal Name	Pin #	Pin #	Signal Name
TX0-	2	1	TX0+
Ground	4	3	Ground
TX1-	6	5	TX1+
*5V/3.3V	8	7	Ground
TX3-	10	9	TX3+
TX2-	12	11	TX2+
Ground	14	13	Ground
TXC-	16	15	TXC+
*5V/3.3V	18	17	ENABKL
+12V	20	19	+12V

Remarks: These connectors support 24-bit.
J7 is 1st channel. J8 is 2nd channel.
JP3 can be used to set 3.3V or 5V.

J6: ATX- Power Signal Connector
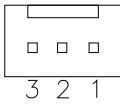
Pin #	Signal Name
1	Ground
2	PS-ON
3	5VSB

J10: SPI Flash Connector (factory use only)

J11, J12: SATA Connectors

J13: Parallel Port Connector

J14: CF Socket

J16: Digital I/O
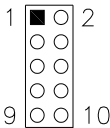
Signal Name	Pin	Pin	Signal Name
GND	1	2	VCC
OUT3	3	4	OUT1
OUT2	5	6	OUT0
IN3	7	8	IN1
IN2	9	10	IN0

CN1, J24: USB0/1 Ports

J17: USB2/3 Ports
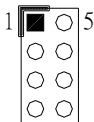
Signal Name	Pin	Pin	Signal Name
Vcc	1	5	Ground
D3-	2	6	D2+
D3+	3	7	D2-
Ground	4	8	Vcc

J18: USB4/5 Ports
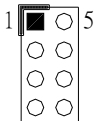
Signal Name	Pin	Pin	Signal Name
Vcc	1	5	Ground
D5-	2	6	D4+
D5+	3	7	D4-
Ground	4	8	Vcc

J19: USB6/7 Ports
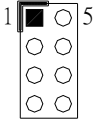
Signal Name	Pin	Pin	Signal Name
Vcc	1	5	Ground
D7-	2	6	D6+
D7+	3	7	D6-
Ground	4	8	Vcc

J23: PS/2 Keyboard/Mouse Connector

Pin #	Signal Name
1	Keyboard data
2	Mouse data
3	GND
4	5V
5	Keyboard clock
6	Mouse clock

LAN1, LAN2: GbE RJ45 Ports

BIOS Setup

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

BIOS Introduction

The Award 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 adds virus and password protection as well as special support for detailed fine-tuning of the chipset controlling the entire system.

BIOS Setup

The Award 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 Award 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.

Phoenix - AwardBIOS CMOS Setup Utility

Standard CMOS Features	Frequency/Voltage Control
Advanced BIOS Features	Load Fail-Safe Defaults
Advanced Chipset Features	Load Optimized Defaults
Integrated Peripherals	Set Supervisor Password
Power Management Setup	Set User Password
PnP/PCI Configurations	Save & Exit Setup
PC Health Status	Exit Without Saving
ESC : Quit	↑ ↓ → ← : Select Item
F10 : Save & Exit Setup	
Time, Date, Hard Disk Type…	

The section below the setup items of the Main Menu displays the control keys for this menu. At the bottom of the Main Menu just below the control keys section, there is another section, which displays information on the currently highlighted item in the list.

Note: *If the system cannot boot after making and saving system changes with Setup, the Award 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 Award 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.*

Standard CMOS Features

"Standard CMOS Features" choice allows you to record some basic hardware configurations in your computer system and set the system clock and error handling. If the board is already installed in a working system, you will not need to select this option. You will need to run the Standard CMOS option, however, if you change your system hardware configurations, the onboard battery fails, or the configuration stored in the CMOS memory was lost or damaged.

```
                       Phoenix - AwardBIOS CMOS Setup Utility
                             Standard CMOS Features
┌─────────────────────────────────────────────────────┬──────────────────────┐
│  Date (mm:dd:yy)          Wed,  Apr 28, 2004          │      Item Help       │
│  Time (hh:mm:ss)          00 : 00 : 00                │  Menu Level  >       │
│                                                       │                      │
│  IDE Channel 0 Master     None                        │  Change the day, month, │
│  IDE Channel 0 Slave      None                        │  Year and century    │
│  IDE Channel 1 Master     None                        │                      │
│  IDE Channel 1 Slave      None                        │                      │
│                                                       │                      │
│  Drive A                  1.44M, 3.5in.               │                      │
│  Drive B                  None                        │                      │
│                                                       │                      │
│  Video                    EGA/VGA                     │                      │
│  Halt On                  All, But Keyboard           │                      │
│                                                       │                      │
│  Base Memory              640K                        │                      │
│  Extended Memory          129024K                     │                      │
│  Total Memory             130048K                     │                      │
└─────────────────────────────────────────────────────┴──────────────────────┘
```

At the bottom of the menu are the control keys for use on this menu. If you need any help in each item field, you can press the <F1> key. It will display the relevant information to help you. The memory display at the lower right-hand side of the menu is read-only. It will adjust automatically according to the memory changed. The following describes each item of this menu.

Date

The date format is:

Day :	**Sun to Sat**
Month :	**1 to 12**
Date :	**1 to 31**
Year :	**1999 to 2099**

To set the date, highlight the "Date" field and use the PageUp/ PageDown or +/- keys to set the current time.

Time
The time format is: **Hour   : 00 to 23**
                        Minute : 00 to 59
                        Second : 00 to 59
To set the time, highlight the "Time" field and use the <PgUp>/ <PgDn> or +/- keys to set the current time.

IDE Channel Master/Slave
The onboard PCI IDE connector provides Primary and Secondary channels for connecting up to two IDE hard disks or other IDE devices.

Press <Enter> to configure the hard disk. The selections include Auto, Manual, and None. Select 'Manual' to define the drive information manually. You will be asked to enter the following items.

CYLS :	Number of cylinders
HEAD :	Number of read/write heads
PRECOMP :	Write precompensation
LANDING ZONE :	Landing zone
SECTOR :	Number of sectors

The Access Mode selections are as follows:
                        CHS   (HD < 528MB)
                        LBA   (HD > 528MB and supports
                               Logical Block Addressing)
                        Large (for MS-DOS only)
                        Auto

Remarks: The main board supports two serial ATA ports and are represented in this setting as IDE Channel 0.

Video

This field selects the type of video display card installed in your system.
You can choose the following video display cards:

EGA/VGA	For EGA, VGA, SEGA, SVGA or PGA monitor adapters. (default)
CGA 40	Power up in 40 column mode.
CGA 80	Power up in 80 column mode.
MONO	For Hercules or MDA adapters.

Halt On

This field determines whether or not the system will halt if an error is
detected during power up.

No errors	The system boot will not be halted for any error that may be detected.
All errors	Whenever the BIOS detects a non-fatal error, the system will stop and you will be prompted.
All, But Keyboard	The system boot will not be halted for a keyboard error; it will stop for all other errors
All, But Diskette	The system boot will not be halted for a disk error; it will stop for all other errors.
All, But Disk/Key	The system boot will not be halted for a keyboard or disk error; it will stop for all others.

Advanced BIOS Features

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

Phoenix - AwardBIOS CMOS Setup Utility
Advanced BIOS Features

CPU Feature	Press Enter	ITEM HELP
Hard Disk Boot Priority	Press Enter	
Virus Warning	Disabled	Menu Level >
CPU L1 and L2 Cache	Enabled	
CPU L3	Enabled	
Hyper-Threading Technology	Enabled	
Quick Power On Self Test	Enabled	
First Boot Device	Hard Disk	
Second Boot Device	CDROM	
Third Boot Device	USB-CDROM	
Boot Other Device	Enabled	
Swap Floppy Drive	Disabled	
Boot Up Floppy Seek	Disabled	
Boot Up NumLock Status	On	
Gate A20 Option	Fast	
Typematic Rate Setting	Disabled	
Typematic Rate (Chars/Sec)	6	
Typematic Delay (Msec)	250	
Security Option	Setup	
APIC Mode	Enabled	
MPS Version Control for OS	1.4	
OS Select For DRAM>64MB	Non-OS2	
Report No FDD For WIN 95	No	
Small Logo (EPA) Show	Disabled	

CPU Feature
Press Enter to configure the settings relevant to CPU Feature.

Hard Disk Boot Priority
With the field, there is the option to choose, aside from the hard disks connected, "Bootable add-in Cards" which refers to other external devices.

Virus Warning
If this option is enabled, an alarm message will be displayed when trying to write on the boot sector or on the partition table on the disk, which is typical of the virus.

CPU L1/L2/L3 Cache
Cache memory is additional memory that is faster than conventional DRAM (system memory). CPUs from 486-type on up contain internal cache memory, and most, but not all, modern PCs have additional (external) cache memory. When the CPU requests data, the system transfers the requested data from the main DRAM into cache memory, for even faster access by the CPU. These allow you to enable (speed up memory access) or disable the cache function.

Hyper-Threading Technology
By default, this field is enabled.

Quick Power On Self Test
When enabled, this field speeds up the Power On Self Test (POST) after the system is turned on. If it is set to *Enabled*, BIOS will skip some items.

First/Second/Third Boot Device
These fields determine the drive that the system searches first for an operating system. The options available include *Floppy*, *LS120*, *Hard Disk*, *CDROM, ZIP100*, *USB-FDD, USB-ZIP*, *USB-CDROM, LAN* and *Disable*.

Boot Other Device
These fields allow the system to search for an OS from other devices other than the ones selected in the First/Second/Third Boot Device.

Swap Floppy Drive
This item allows you to determine whether or not to enable Swap Floppy Drive. When enabled, the BIOS swaps floppy drive assignments so that Drive A becomes Drive B, and Drive B becomes Drive A. By default, this field is set to *Disabled.*

Boot Up Floppy Seek
This feature controls whether the BIOS checks for a floppy drive while booting up. If it cannot detect one (either due to improper configuration or its absence), it will flash an error message.

Boot Up NumLock Status
This allows you to activate the NumLock function after you power up the system.

Gate A20 Option
This field allows you to select how Gate A20 is worked. Gate A20 is a device used to address memory above 1 MB.

Typematic Rate Setting
When disabled, continually holding down a key on your keyboard will generate only one instance. When enabled, you can set the two typematic controls listed next. By default, this field is set to *Disabled.*

Typematic Rate (Chars/Sec)
When the typematic rate is enabled, the system registers repeated keystrokes speeds. Settings are from 6 to 30 characters per second.

Typematic Delay (Msec)
When the typematic rate is enabled, this item allows you to set the time interval for displaying the first and second characters. By default, this item is set to *250msec.*

Security Option
This field allows you to limit access to the System and Setup. The default value is *Setup*. When you select *System,* the system prompts for the User Password every time you boot up. When you select *Setup,* the system always boots up and prompts for the Supervisor Password only when the Setup utility is called up.

APIC Mode
APIC stands for Advanced Programmable Interrupt Controller. The default setting is *Enabled*.

MPS Version Control for OS
This option is specifies the MPS (Multiprocessor Specification) version for your operating system. MPS version 1.4 added extended configuration tables to improve support for multiple PCI bus configurations and improve future expandability. The default setting is *1.4*.

OS Select for DRAM > 64MB
This option allows the system to access greater than 64MB of DRAM memory when used with OS/2 that depends on certain BIOS calls to access memory. The default setting is *Non-OS/2*.

Report No FDD For WIN 95
If you are using Windows 95/98 without a floppy disk drive, select Enabled to release IRQ6. This is required to pass Windows 95/98's SCT test. You should also disable the Onboard FDC Controller in the Integrated Peripherals screen when there's no floppy drive in the system. If you set this feature to Disabled, the BIOS will not report the missing floppy drive to Win95/98.

Small Logo (EPA) Show
The EPA logo appears at the right side of the monitor screen when the system is boot up.

Advanced Chipset Features

This Setup menu controls the configuration of the chipset.

```
                        Phoenix - AwardBIOS CMOS Setup Utility
                             Advanced Chipset Features
┌─────────────────────────────────────────────────┬──────────────────────┐
│ DRAM Timing Selectable         By SPD            │      ITEM HELP       │
│ CAS Latency Time               Auto              │ Menu Level >         │
│ DRAM RAS# to CAS# Delay        Auto              │                      │
│ DRAM RAS# Precharge            Auto              │                      │
│ Precharge delay (tRAS)         Auto              │                      │
│ System Memory Frequency        Auto              │                      │
│ SLP_S4# Assertion Width        4 to 5 Sec        │                      │
│ System BIOS Cacheable          Enabled           │                      │
│ Video BIOS Cacheable           Disabled          │                      │
│ Memory Hole at 15M-16M         Disabled          │                      │
│ PCI Express Root Port Func     Press Enter       │                      │
│                                                  │                      │
│                                                  │                      │
│                                                  │                      │
│ ** VGA Setting **                                │                      │
│ On-Chip Frame Buffer Size      8MB               │                      │
│ DVMT Mode                      DVMT              │                      │
│ DVMT/FIXED memory Size         128MB             │                      │
│ SDVO Device Setting            None              │                      │
│ SDVO LVDS Protocol             1 Ch SPGW 18 bit  │                      │
│ SDVO Panel Number              1024x768          │                      │
│ Boot Display                   CRT               │                      │
│ Panel Scaling                  Auto              │                      │
└─────────────────────────────────────────────────┴──────────────────────┘
```

DRAM Timing Selectable

This option refers to the method by which the DRAM timing is selected.
The default is *By SPD*.

CAS Latency Time

You can configure CAS latency time in HCLKs as 2 or 2.5 or 3. The
system board designer should set the values in this field, depending on
the DRAM installed. Do not change the values in this field unless you
change specifications of the installed DRAM or the installed CPU.

DRAM RAS# to CAS# Delay

This option allows you to insert a delay between the RAS (Row Address
Strobe) and CAS (Column Address Strobe) signals. This delay occurs
when the SDRAM is written to, read from or refreshed. Reducing the
delay improves the performance of the SDRAM.

DRAM RAS# Precharge

This option sets the number of cycles required for the RAS to
accumulate its charge before the SDRAM refreshes. The default setting
for the Active to Precharge Delay is *Auto*.

Precharge Delay (tRAS)
The default setting for the Precharge Delay is *Auto*.

System Memory Frequency
The default setting is *Auto*.

SLP_S4# Assertion Width
The default setting is *4 to 5 Sec*.

System BIOS Cacheable
The setting of *Enabled* allows caching of the system BIOS ROM at F000h-FFFFFh, resulting in better system performance. However, if any program writes to this memory area, a system error may result.

Video BIOS Cacheable
The Setting *Enabled* allows caching of the video BIOS ROM at C0000h-F7FFFh, resulting in better video performance. However, if any program writes to this memory area, a system error may result.

Memory Hole At 15M-16M
In order to improve performance, certain space in memory can be reserved for ISA cards. This memory must be mapped into the memory space below 16 MB. The choices are *Enabled* and *Disabled*.

On-Chip VGA Setting
The fields under the On-Chip VGA Setting and their default settings are:
    On-Chip Frame Buffer Size: 8MB
    DVMT Mode: DVMT
    DVMT/Fixed Memory Size: 128MB
    SDVO Device Setting: None
    SDVO LCDS Protocol: 1 Ch SPGW 18 bit
    SDVO Panel Number: 1024x768
    Boot Display: CRT
    Panel Scaling: Auto

The options under the SDVO LCDS Protocol field include:
    1 Ch SPGW 18 bit
    2 Ch SPGW 18 bit
    1 Ch OpenLDI 18 bit
    2 Ch OpenLDI 18 bit
    1 Ch SPGW 24 bit
    2 Ch SPGW 24 bit
    1 Ch OpenLDI 24 bit
    2 Ch OpenLDI 24 bit

SDVO Panel Number
This field allows you to select the SDVO Panel type. The default values
for these ports are:

   640x480
   852x480
   800x600
   1024x768
   1280x1024
   1280x800
   1366x768
   1400x1050
   1440x900
   1600x1200
   1920x1080
   1920x1200

Integrated Peripherals

This section sets configurations for your hard disk and other integrated peripherals. The first screen shows three main items for user to select. Once an item selected, a submenu appears. Details follow.

Phoenix - AwardBIOS CMOS Setup Utility
Integrated Peripherals

		ITEM HELP
OnChip IDE Device	Press Enter	Menu Level >
Onboard Device	Press Enter	
SuperIO Device	Press Enter	
2nd SuperIO Device	Press Enter	

Phoenix - AwardBIOS CMOS Setup Utility
OnChip IDE Device

		ITEM HELP
IDE HDD Block Mode	Enabled	Menu Level >
IDE DMA Transfer Access	Enabled	
On-chip Primary PCI IDE	Enabled	
IDE Primary Master PIO	Auto	
IDE Primary Slave PIO	Auto	
IDE Primary Master UDMA	Auto	
IDE Primary Slave UDMA	Auto	
On-Chip Secondary PCI IDE	Enabled	
IDE Secondary Master PIO	Auto	
IDE Secondary Slave PIO	Auto	
IDE Secondary Master UDMA	Auto	
IDE Secondary Slave UDMA	Auto	
*** On-Chip Serial ATA Setting ***		
SATA Mode	IDE	
On-Chip Serial ATA	Auto	
SATA PORT Speed Settings	Disabled	
PATA IDE Mode	Secondary	
SATA port	P0, P2 is Primary	

Phoenix - AwardBIOS CMOS Setup Utility
Onboard Device

		ITEM HELP
USB Controller	Enabled	Menu Level >
USB 2.0 Controller	Enabled	
USB Keyboard Support	Disabled	
Azalia/AC97 Audio Select	Auto	

Phoenix - AwardBIOS CMOS Setup Utility
SuperIO Device

POWER ON Function	BUTTON ONLY	ITEM HELP
KB Power ON Password	Enter	Menu Level >
Hot Key power ON	Ctrl-F1	
Onboard FDC Controller	Enabled	
Onboard Serial Port 1	3F8/IRQ4	
Onboard Serial Port 2	2F8/IRQ3	
UART Mode Select	Normal	
RxD , TxD Active	Hi, Lo	
IR Transmission Delay	Enabled	
UR2 Duplex Mode	Half	
Use IR Pins	IR-Rx2Tx2	
Onboard Parallel Port	378/IRQ7	
Parallel Port Mode	SPP	
EPP Mode Select	EPP1.7	
ECP Mode Use DMA	3	
PWRON After PWR-Fail	Off	

Phoenix - AwardBIOS CMOS Setup Utility
2nd Super IO Device

Onboard Serial Port 3	2A0h	ITEM HELP
Serial Port 3 Use IRQ	IRQ11	Menu Level >
Onboard Serial Port 4	2A8h	
Serial Port 4 Use IRQ	IRQ10	

IDE HDD Block Mode

This field allows your hard disk controller to use the fast block mode to transfer data to and from your hard disk drive.

IDE DMA Transfer Access

By default, this field is enabled.

OnChip Primary/Secondary PCI IDE

The integrated peripheral controller contains an IDE interface with support for two IDE channels. Select *Enabled* to activate each channel separately.

IDE Primary/Secondary Master/Slave PIO

These fields allow your system hard disk controller to work faster. Rather than have the BIOS issue a series of commands that transfer to or from the disk drive, PIO (Programmed Input/Output) allows the BIOS to communicate with the controller and CPU directly.

The system supports five modes, numbered from 0 (default) to 4, which primarily differ in timing. When Auto is selected, the BIOS will select the best available mode.

IDE Primary/Secondary Master/Slave UDMA

These fields allow your system to improve disk I/O throughput to 33Mb/sec with the Ultra DMA/33 feature. The options are *Auto* and *Disabled*.

On-Chip Serial ATA Setting

The fields under the SATA setting include:
    SATA Mode: IDE; (other setting is AHCI)
    SATA PORT Speed Settings: Disabled
    On-Chip Serial ATA: Auto
    PATA IDE Mode: Secondary
    SATA port: P0, P2 is Primary

USB Controller

The options for this field are *Enabled* and *Disabled*. By default, this field is set to **Enabled**.

USB 2.0 Controller

The options for this field are *Enabled* and *Disabled*. By default, this field is set to **Enabled**. In order to use USB 2.0, necessary OS drivers must be installed first.

USB Keyboard Support

The options for this field are *Enabled* and *Disabled*.

Azalia/AC97 Audio Select

This field, by default, is set to **Auto**.

Power ON Function

This field is related to how the system is powered on – such as with the use of conventional power button, keyboard or hot keys. The default is **BUTTON ONLY**.

KB Power ON Password

This field allows users to set the password when keyboard power on is the mode of the Power ON function.

Hot Key Power ON

This field sets certain keys, also known as hot keys, on the keyboard that can be used as a 'switch' to power on the system.

Onboard FDC Controller

Select *Enabled* if your system has a floppy disk controller (FDC) installed on the motherboard and you wish to use it. If you install an add-in FDC or the system has no floppy drive, select Disabled in this field. This option allows you to select the onboard FDD port.

Onboard Serial/Parallel Port

These fields allow you to select the onboard serial and parallel ports and their addresses. The default values for these ports are:

Serial Port 1	3F8/IRQ4
Serial Port 2	2F8/IRQ3
Serial Port 3	2A0/IRQ11
Serial Port 4	2A8/IRQ10
Parallel Port	378H/IRQ7

UART Mode Select

This field determines the UART 2 mode in your computer. The default value is *Normal*. Other options include *IrDA* and *ASKIR*.

Parallel Port Mode

This field allows you to determine parallel port mode function.

SPP	Standard Printer Port
EPP	Enhanced Parallel Port
ECP	Extended Capabilities Port
ECP+EPP	Combination of ECP and EPP capabilities
Normal	Normal function

PWRON After PWR-Fail

This field sets the system power status whether *on or off* when power returns to the system from a power failure situation.

Power Management Setup

Power-Supply Type	ATX	ITEM HELP
ACPI Function	Enabled	Menu Level >
ACPI Suspend	S1(POS)	
RUN VGABIOS if S3 Resume	Auto	
Power Management	User Define	
Video Off Method	V/H SYNC+Blank	
Video Off In Suspend	Yes	
Suspend Type	Stop Grant	
Modem Use IRQ	3	
HDD Power Down	Disabled	
Soft-Off by PWR-BTTN	Instant-Off	
CPU THRM-Throttling	50%	
Wake-Up by PCI Card	Disabled	
Power On by Ring	Disabled	
Resume by Alarm	Disabled	
Date (of Month) Alarm	0	
Time (hh:mm:ss) Alarm	0 : 0 : 0	
** Reload Global Timer Events **		
Primary IDE 0	Enabled	
Primary IDE 1	Enabled	
Secondary IDE 0	Enabled	
Secondary IDE 1	Enabled	
FDD, COM, LPT Port	Enabled	
PCI PIRQ[A-D] #	Enabled	

Power Supply Type
The default setting is *ATX*. The other setting is AT mode.

ACPI Function
Enable this function to support ACPI (Advance Configuration and Power Interface).

ACPI Suspend
The default setting of the ACPI Suspend mode is *S1(POS)*.

RUN VGABIOS if S3 Resume
The default setting of this field is *Auto*.

Power Management

This field allows you to select the type of power saving management modes. There are four selections for Power Management.

Min. Power Saving	Minimum power management
Max. Power Saving	Maximum power management.
User Define	Each of the ranges is from 1 min. to 1hr. Except for HDD Power Down which ranges from 1 min. to 15 min.

Video Off Method

This field defines the Video Off features. There are three options.

V/H SYNC + Blank	Default setting, blank the screen and turn off vertical and horizontal scanning.
DPMS	Allows BIOS to control the video display.
Blank Screen	Writes blanks to the video buffer.

Video Off In Suspend

When enabled, the video is off in suspend mode. The default setting is *Yes*.

Suspend Type

The default setting for the Suspend Type field is *Stop Grant*.

Modem Use IRQ

This field sets the IRQ used by the Modem. By default, the setting is *3*.

HDD Power Down

When enabled, and after the set time of system inactivity, the hard disk drive will be powered down while all other devices remain active.

Soft-Off by PWRBTN

This field defines the power-off mode when using an ATX power supply. The *Instant Off* mode allows powering off immediately upon pressing the power button. In the *Delay 4 Sec* mode, the system powers off when the power button is pressed for more than four seconds or enters the suspend mode when pressed for less than 4 seconds.

CPU THRM-Throttling
By default, this field is set to *50%*.

Wake up by PCI Card
By default, this field is disabled.

Power On by Ring
This field enables or disables the power on of the system through the modem connected to the serial port or LAN.

Resume by Alarm
This field enables or disables the resumption of the system operation. When enabled, the user is allowed to set the *Date* and *Time*.

Reload Global Timer Events
The HDD, FDD, COM, LPT Ports, and PCI PIRQ are I/O events that can prevent the system from entering a power saving mode or can awaken the system from such a mode. When an I/O device wants to gain the attention of the operating system, it signals this by causing an IRQ to occur. When the operating system is ready to respond to the request, it interrupts itself and performs the service.

PNP/PCI Configurations

This option configures the PCI bus system. All PCI bus systems on the system use INT#, thus all installed PCI cards must be set to this value.

Phoenix - AwardBIOS CMOS Setup Utility
PnP/PCI Configurations

PNP OS Installed	No	ITEM HELP
Init Display First	PCI Slot	Menu Level
Reset Configuration Data	Disabled	
		Select Yes if you are
Resources Controlled By	Auto (ESCD)	using a Plug and Play
IRQ Resources	Press Enter	capable operating
DMA Resource	Press Ente	system Select No if
		you need the BIOS to
PCI/VGA Palette Snoop	Disabled	configure non-boot
		devices
PCI Express relative items		
Maximum Payload Size	128	

PNP OS Install
Enable the PNP OS Install option if it is supported by the operating system installed. The default value is *No.*

Init Display First
The default setting is *PCI Card*.

Reset Configuration Data
This field allows you to determine whether to reset the configuration data or not. The default value is ***Disabled.***

Resources Controlled by
This PnP BIOS can configure all of the boot and compatible devices with the use of a PnP operating system such as Windows 95.

PCI/VGA Palette Snoop
Some non-standard VGA display cards may not show colors properly. This field allows you to set whether or not MPEG ISA/VESA VGA cards can work with PCI/VGA. When this field is enabled, a PCI/VGA can work with an MPEG ISA/VESA VGA card. When this field is disabled, a PCI/VGA cannot work with an MPEG ISA/VESA card.

Maximum Payload Size
The default setting of the PCI Express Maximum Payload Size is 128.

PC Health Status

This section shows the parameters in determining the PC Health Status. These parameters include temperatures, fan speeds and voltages.

Phoenix - AwardBIOS CMOS Setup Utility
PC Health Status

Shutdown Temperature	Disabled	ITEM HELP
CPU Warning Temperature	Disabled	Menu Level >
Current System Temp	36°C/96°F	
Current CPU Temp	40°C/104°F	
System Fan Speed	0 RPM	
CPU Fan Speed	0 RPM	
Vcore(V)	0.90 V	
+12 V	12.50 V	
+1.8V	1.88V	
+5 V	5.07 V	
-12 V	-12.20V	
+3.3V	3.37V	
VBAT (V)	3.21 V	
3.3VSB(V)	3.34 V	
LCD Back Light Control	0	

Shutdown Temperature

This field allows the user to set the temperature by which the system automatically shuts down once the threshold temperature is reached. This function can help prevent damage to the system that is caused by overheating.

CPU Warning Temperature

This field allows the user to set the temperature so that when the temperature is reached, the system sounds a warning. This function can help prevent damage to the system that is caused by overheating.

Temperatures/Voltages

These fields are the parameters of the hardware monitoring function feature of the board. The values are read-only values as monitored by the system and show the PC health status.

LCD Back Light Control

The default setting of the LCD Back Light Control is 0.

Frequency/Voltage Control

This section shows the user how to configure the processor frequency.

Phoenix - AwardBIOS CMOS Setup Utility
Frequency/Voltage Control

Auto Detect PCI Clk	Disabled	ITEM HELP
Spread Spectrum	Disabled	Menu Level >

Auto Detect PCI Clk

This field enables or disables the auto detection of the PCI clock.

Spread Spectrum Modulated

This field sets the value of the spread spectrum. The default setting is *Disabled*. This field is for CE testing use only.

Load Fail-Safe Defaults

This option allows you to load the troubleshooting default values permanently stored in the BIOS ROM. These default settings are non-optimal and disable all high-performance features.

Load Optimized Defaults

This option allows you to load the default values to your system configuration. These default settings are optimal and enable all high performance features.

Set Supervisor Password

These two options set the system password. Supervisor Password sets a password that will be used to protect the system and Setup utility. User Password sets a password that will be used exclusively on the system. To specify a password, highlight the type you want and press <Enter>. The Enter Password: message prompts on the screen. Type the password, up to eight characters in length, and press <Enter>. The system confirms your password by asking you to type it again. After setting a password, the screen automatically returns to the main screen.

To disable a password, just press the <Enter> key when you are prompted to enter the password. A message will confirm the password to be disabled. Once the password is disabled, the system will boot and you can enter Setup freely.

Save & Exit Setup

This option allows you to determine whether or not to accept the modifications. If you type "Y", you will quit the setup utility and save all changes into the CMOS memory. If you type "N", you will return to Setup utility.

Exit Without Saving

Select this option to exit the Setup utility without saving the changes you have made in this session. Typing "Y" will quit the Setup utility without saving the modifications. Typing "N" will return you to Setup utility.

This page is intentionally left blank.

Drivers Installation

This section describes the installation procedures for software and drivers under the Windows 2000, Windows XP and Windows Vista. The software and drivers are included with the board. If you find the items missing, please contact the vendor where you made the purchase. The contents of this section include the following:

IMPORTANT NOTE:
After installing your Windows operating system (Windows 2000/XP/Vista), you must install first the Intel Chipset Software Installation Utility before proceeding with the drivers installation.

Intel Chipset Software Installation Utility

The Intel Chipset Drivers should be installed first before the software drivers to enable Plug & Play INF support for Intel chipset components. Follow the instructions below to complete the installation under Windows 2000, Windows XP and Windows Vista. (Before installing this utility, please update your system to Windows 2000 SP4 or Windows XP SP2)

1. Insert the CD that comes with the board. Click *Intel* at the left side, then *Intel(R) I945GM/GME/GSE Chipset Drivers*.

2. Click *Intel(R) Chipset Software Installation Utility.*

3. When the welcome screen to the Intel(R) Chipset Software Installation Utility appears, click *Next* to continue.

4. Click *Yes* to accept the software license agreement and proceed with the installation process.

5. On Readme Information screen, click *Next* to continue the installation.

6. The Setup process is now complete. Click *Finish* to restart the computer and for changes to take effect.

VGA Drivers Installation

To install the VGA drivers, follow the steps below to proceed with the installation.

1. Insert the CD that comes with the board. Click *Intel* at the left side, then *Intel(R) I945GM/GME/GSE Chipset Drivers*.

2. Click *Intel(R) I945GM/GME/GSE Chipset Family Graphics Driver.*

3. When the welcome screen of the Intel(R) Graphics Media Accelerator Driver appears, click *Next* to continue.

4. Click *Yes* to to agree with the license agreement and continue the installation.

5. Click *Next* in the Readme File Information window.

6. Click *Next* in the Setup Progress window.

7. Setup is now complete. Click *Finish* to restart the computer and for changes to take effect.

Realtek High Definition Audio Driver Installation

Follow the steps below to install the Realtek High Definition Audio Drivers.

1. Insert the CD that comes with the board. Click *Intel* at the left side, then *Intel(R) I945GM/GME/GSE Chipset Drivers*, and then *Realtek Audio Driver*.

2. Click *Realtek High Definition Audio Driver.*

3. When the welcome screen to the InstallShield Wizard for *Realtek High Definition Audio Driver* appears, click *Next* to start the installation.

4. When the InstallShieldWizard has finished performing maintenance operations on Realtek High Definition Audio Driver, click *Finish* to restart the computer.

LAN Drivers Installation

Follow the steps below to install Realtek 8111C LAN drivers.

1. Insert the CD that comes with the board. Click *LAN Card* at the left side and then *Realtek GbE _FE Ethernet PCI-E NIC Driver*.
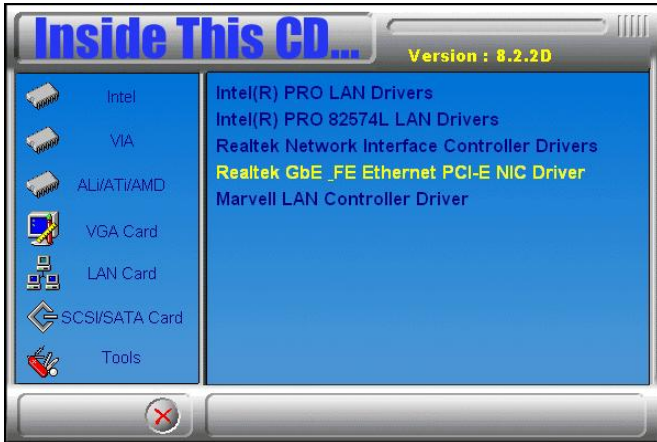
2. In the welcome screen of the InstallShield Wizard for REALTEK GbE & FE Ethernet PCI-E NIC Driver, click *Next*.

3. In the InstallShield Wizard screen, click *Install* to begin the installation.

4. InstallShield Wizard completed. Click *Finish* to exit the Wizard.

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
278 - 27F	Parallel Port #2(LPT2)
2F8h - 2FFh	Serial Port #2(COM2)
2B0 - 2DF	Graphics adapter Controller
378h - 3FFh	Parallel Port #1(LPT1)
360 - 36F	Network Ports
3B0 - 3BF	Monochrome & Printer adapter
3C0 - 3CF	EGA adapter
3D0 - 3DF	CGA adapter
3F0h - 3F7h	Floppy Disk Controller
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	Floppy Disk Controller
IRQ7	Parallel Port #1
IRQ8	Real Time Clock
IRQ9	Reserved
IRQ10	Reserved
IRQ11	Reserved
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 <stdio.h>
#include <stdlib.h>
#include "W627EHF.H"
//========================================================================
int main (int argc, char *argv[]);
void copyright(void);
void EnableWDT(int);
void DisableWDT(void);
//========================================================================
int main (int argc, char *argv[])
{
        unsigned char bBuf;
        unsigned char bTime;
        char **endptr;

        copyright();

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

        if (Init_W627EHF() == 0)
        {
                printf(" Winbond 83627HF is not detected, program abort.\n");
                return 1;
        }
        bTime = strtol (argv[1], endptr, 10);
        printf("System will reset after %d seconds\n", bTime);

        EnableWDT(bTime);

        return 0;
}
//========================================================================
```

```
void copyright(void)
{
        printf("\n======= Winbond 83627EHF Watch Timer Tester (AUTO DETECT) =======\n"\
                "        Usage : W627E_WD reset_time\n"\
                "        Ex : W627E_WD 3  => reset system after 3 second\n"\
                "            W627E_WD 0  => disable watch dog timer\n");
}
//=========================================================================
void EnableWDT(int interval)
{
        unsigned char bBuf;

        bBuf = Get_W627EHF_Reg( 0x2D);
        bBuf &= (!0x01);
        Set_W627EHF_Reg( 0x2D, bBuf);                   //Enable WDTO

        Set_W627EHF_LD( 0x08);                          //switch to logic device 8
        Set_W627EHF_Reg( 0x30, 0x01);                   //enable timer

        bBuf = Get_W627EHF_Reg( 0xF5);
        bBuf &= (!0x08);
        Set_W627EHF_Reg( 0xF5, bBuf);                   //count mode is second

        Set_W627EHF_Reg( 0xF6, interval);               //set timer
}
//=========================================================================
void DisableWDT(void)
{
        Set_W627EHF_LD(0x08);                           //switch to logic device 8
        Set_W627EHF_Reg(0xF6, 0x00);                    //clear watchdog timer
        Set_W627EHF_Reg(0x30, 0x00);                    //watchdog disabled
}
//=========================================================================
```

```
//=========================================================================
//
// 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 "W627EHF.H"
#include <dos.h>
//=========================================================================
unsigned int W627EHF_BASE;
void Unlock_W627EHF (void);
void Lock_W627EHF (void);
//=========================================================================
unsigned int Init_W627EHF(void)
{
        unsigned int result;
        unsigned char ucDid;

        W627EHF_BASE = 0x2E;
        result = W627EHF_BASE;

        ucDid = Get_W627EHF_Reg(0x20);
        if (ucDid == 0x88)
        {       goto Init_Finish;        }

        W627EHF_BASE = 0x4E;
        result = W627EHF_BASE;
        ucDid = Get_W627EHF_Reg(0x20);
        if (ucDid == 0x88)
        {       goto Init_Finish;        }

        W627EHF_BASE = 0x00;
        result = W627EHF_BASE;

Init_Finish:
        return (result);
}
//=========================================================================
void Unlock_W627EHF (void)
{
        outportb(W627EHF_INDEX_PORT, W627EHF_UNLOCK);
        outportb(W627EHF_INDEX_PORT, W627EHF_UNLOCK);
}
//=========================================================================
void Lock_W627EHF (void)
{
        outportb(W627EHF_INDEX_PORT, W627EHF_LOCK);
}
//=========================================================================
void Set_W627EHF_LD( unsigned char LD)
{
        Unlock_W627EHF();
        outportb(W627EHF_INDEX_PORT, W627EHF_REG_LD);
        outportb(W627EHF_DATA_PORT, LD);
        Lock_W627EHF();
}
```

```
//================================================================================
void Set_W627EHF_Reg( unsigned char REG, unsigned char DATA)
{
       Unlock_W627EHF();
       outportb(W627EHF_INDEX_PORT, REG);
       outportb(W627EHF_DATA_PORT, DATA);
       Lock_W627EHF();
}
//================================================================================
unsigned char Get_W627EHF_Reg(unsigned char REG)
{
       unsigned char Result;
       Unlock_W627EHF();
       outportb(W627EHF_INDEX_PORT, REG);
       Result = inportb(W627EHF_DATA_PORT);
       Lock_W627EHF();
       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 __W627EHF_H
#define __W627EHF_H                    1
//================================================================================
#define       W627ehf_INDEX_PORT        (W627EHF_BASE)
#define       W627ehf_DATA_PORT         (W627EHF_BASE+1)
//================================================================================
#define       W627ehf_REG_LD            0x07
//================================================================================
#define W627ehf_UNLOCK                  0x87
#define       W627ehf_LOCK              0xAA
//================================================================================
unsigned int Init_W627EHF(void);
void Set_W627EHF_LD( unsigned char);
void Set_W627EHF_Reg( unsigned char, unsigned char);
unsigned char Get_W627EHF_Reg( unsigned char);
//================================================================================
#endif  //__W627EHF_H
```

D. Digital I/O Sample Code

Filename: W627hf.h

```
//=====================================================================
//
// 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 __W627HF_H
#define __W627HF_H                    1
//=====================================================================
#define      W627HF_INDEX_PORT        (W627HF_BASE)
#define      W627HF_DATA_PORT         (W627HF_BASE+1)
//=====================================================================
#define      W627HF_REG_LD            0x07
//=====================================================================
#define W627HF_UNLOCK                 0x87
#define      W627HF_LOCK              0xAA
//=====================================================================
unsigned int Init_W627HF(void);
void Set_W627HF_LD( unsigned char);
void Set_W627HF_Reg( unsigned char, unsigned char);
unsigned char Get_W627HF_Reg( unsigned char);
//=====================================================================
#endif        //__W627HF_H
```

Filename: W627hf.cpp
```
//========================================================================
//
// 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.
//
//========================================================================
unsigned int W627HF_BASE;
void Unlock_W627HF (void);
void Lock_W627HF (void);
//========================================================================
unsigned int Init_W627HF(void)
{
      unsigned int result;        //0=NA, 1=627HF, 2=627THF, 4=627EHF

//      W627HF_BASE = 0x2E;
      W627HF_BASE = 0x4E;
      result = Get_W627HF_Reg(0x20);
      if (result == 0x52)
      {
            result = 1;
            goto Init_Finish;
      }
      else if (result == 0x82)
      {
            result = 2;
            goto Init_Finish;
      }
      else if (result == 0x88)
      {
            result = 4;
            goto Init_Finish;
      }

//      W627HF_BASE = 0x4E;
      W627HF_BASE = 0x2E;

      result = Get_W627HF_Reg(0x20);
      if (result == 0x52)
      {
            result = 1;
            goto Init_Finish;
      }
      else if (result == 0x82)
      {
            result = 2;
            goto Init_Finish;
      }
      else if (result == 0x88)
      {
            result = 4;
            goto Init_Finish;
      }
```

```
    W627HF_BASE = 0x00;
    result = 0;

Init_Finish:
    return (result);
}
//==================================================================
void Unlock_W627HF (void)
{
    outportb(W627HF_INDEX_PORT, W627HF_UNLOCK);
    outportb(W627HF_INDEX_PORT, W627HF_UNLOCK);
}
//==================================================================
void Lock_W627HF (void)
{
    outportb(W627HF_INDEX_PORT, W627HF_LOCK);
}
//==================================================================
void Set_W627HF_LD( unsigned char LD)
{
    Unlock_W627HF();
    outportb(W627HF_INDEX_PORT, W627HF_REG_LD);
    outportb(W627HF_DATA_PORT, LD);
    Lock_W627HF();
}
//==================================================================
void Set_W627HF_Reg( unsigned char REG, unsigned char DATA)
{
    Unlock_W627HF();
    outportb(W627HF_INDEX_PORT, REG);
    outportb(W627HF_DATA_PORT, DATA);
    Lock_W627HF();
}
//==================================================================
unsigned char Get_W627HF_Reg(unsigned char REG)
{
    unsigned char Result;
    Unlock_W627HF();
    outportb(W627HF_INDEX_PORT, REG);
    Result = inportb(W627HF_DATA_PORT);
    Lock_W627HF();
    return Result;
}
//==================================================================
```

File of the Main.cpp

```
//=====================================================================
// 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 "W627HF.H"
//------------------------------------------------------------------------
void ClrKbBuf(void);
int main (void);
unsigned char GetDI(unsigned char);
void SetDo(unsigned char, unsigned char);
//------------------------------------------------------------------------
int main (void)
{
      unsigned char ucDO = 0;                               //data for digital output
      unsigned char ucDI;                                   //data for digital input
      unsigned char ucBuf;
      char SIO;

      SIO = Init_W627HF();
      if (SIO == 0)
      {
           printf("Can not detect Winbond 83627HF/83627THF/83627EHF, program abort.\n");
           return(1);
      }
      switch (SIO)
      {     //----------------------------------------------------------------
           case 1:
           printf("Winbond 83627HF is detected .\n");
           break;
           //----------------------------------------------------------------
           case 2:
           printf("Winbond 83627THF is detected.\n");
           break;
           //----------------------------------------------------------------
           case 4:
           printf("Winbond 83627EHF is detected.\n");
           break;
           //----------------------------------------------------------------
      }

      //bit 0..3 = input signal
      //bit 4..7 = output signal

      ucDI = GetDI(0x0F);                                   //get current DI status
      SetDo(ucDO, 0xF0);                                    //set current DO status
      return 0;
}
//------------------------------------------------------------------------
unsigned char GetDI(unsigned char Mask)
{
```

```
        unsigned char result;

        Set_W627HF_LD(0x07);                                //switch to logic device 7
        Set_W627HF_Reg(0xF0, Mask);                //set the DIO direction
        result = Get_W627HF_Reg(0xF1) & Mask;
        return (result);
}
//----------------------------------------------------------------------------
void SetDo(unsigned char NewData, unsigned char Mask)
{
        Set_W627HF_LD(0x07);                                //switch to logic device 7
        Set_W627HF_Reg(0xF0, ~Mask);                //set the DIO direction
        Set_W627HF_Reg(0xF1, NewData & Mask);
}
//----------------------------------------------------------------------------
void ClrKbBuf(void)
{
        while(kbhit())
        {       getch();        }
}
//----------------------------------------------------------------------------
```