MB899X

Intel ® Core™ 2 Duo
Core™ Duo/ Solo 945GME
Mini-ITX Motherboard

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 Intel® Core 2 Duo and Intel® Core Duo/Solo processors are registered trademarks 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 MB899X Mini ITX board incorporates the Mobile Intel® 945GME Express Chipset for Embedded Computing, consisting of the Intel® 945GME Graphic Memory Controller Hub (GMCH) and Intel® I/O Controller Hub 7-M (ICH7-M), an optimized integrated graphics solution with a 533MHz and 667MHz front-side bus. Dimensions of the board are 170mm x 170mm.

The integrated powerful 3D graphics engine, based on Intel® Graphics Media Accelerator 950 (Intel® GMA 950) architecture, operates at core speeds of up to 400 MHz. It features a low-power design, is validated with the Intel® Core Duo and Intel® Core Solo processors on 65nm process. With dual channel DDR2 667MHz two DIMM socket on board, the board supports up to 4GB of DDR2 system memory.

Intel® Graphics supports a unique intelligent memory management scheme called Dynamic Video Memory Technology (DVMT). DVMT handles diverse applications by providing the maximum (224MB) availability of system memory for general computer usage, while supplying additional graphics memory when a 3D-intensive application requests it. The Intel GMA 950 graphics architecture also takes advantage of the high-performance Intel processor. Intel GMA 950 graphics supports Dual Independent Display technology.

The main features of the board are:

- Supports Intel® Core™ 2 Duo (Merom-Napa Refresh),
- Intel® Core™ Duo/Solo mobile processors
- Supports up to 2.33GHz, 533MHz/667MHz FSB
- Two DDR2 SDRAM DIMM, Max. 4GB memory
- Onboard 10/100 BaseT and Marvell PCI-Express Gigabit LAN
- Intel® 945GME Express VGA for CRT / LVDS
- 2x SATA, 6x USB 2.0, 4x COM, Watchdog timer,
- 1x MiniPCI, 1x MiniPCI-E, 1xPCI-E(x16) slots

Checklist

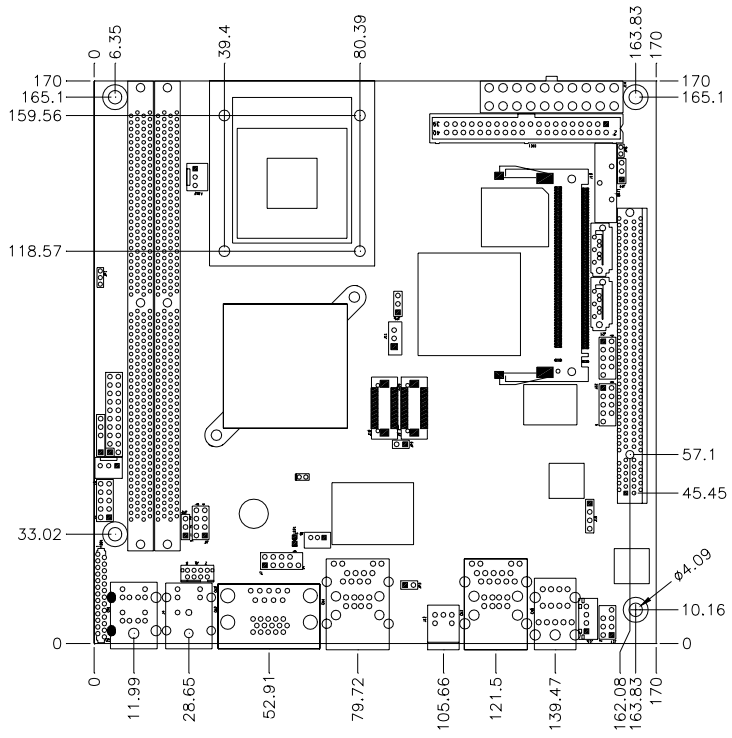
Your MB899X package should include the items listed below.

- The MB899X Mini-ITX motherboard
- This User's Manual
- 1 CD containing chipset drivers and flash memory utility
- Cable kit (IDE, Serial port, Serial ATA)

MB899X Specifications

CPU Supported	Intel® Core™ 2 Duo , Intel® Core™ Duo /Solo processors
CPU Voltage	0.700V ~ 1.5V (IMVP-6)
System Speed	Up to 2.33GHz or above
CPU FSB	533MHz/667MHz FSB
Cache	2MB
Green /APM	APM1.2
CPU Socket	mPGA 478MT Socket
Chipset	Intel 945GME Chipset GMCH: 82945GME 1466-pin Micro-FCBGA ICH7M: 82801GBM 652-pin mBGA
BIOS	Award BIOS, support ACPI Function
Memory	DDR2 667/533 SDRAM DIMM x2 (w/o ECC function), Max. 4GB
VGA	945GME built-in, supports CRT, RCA, S-VIDEO, HDTV
PCIEx16 SLOT	For External PCI Express Based graphics card
LVDS LCD Panel	945GME built-in, supports 18+18 bits, single or dual channel LVDS
LAN	1. ICH7M built-in 10/100BT MAC + Intel 82562ET PHY 2. Marvell 88E8053 PCI Express Gigabit LAN controller x1
USB	ICH7M built-in USB 2.0 host controller, support 6 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
Audio	ICH7M Built-in Audio controller + AC97 Codec ALC655 w/ 6 channels (Line-out, Line-in, Mic.), SPDIF-OUT + NS LM4950 (8-ohm 2W stereo audio power amplifier)
LPC I/O	W83627EHF: COM1, COM2 (RS232), Slim FDC 1.44MB, IrDA x1 & hardware monitor (3 thermal, 4 voltage monitor inputs, 2 fan headers) Fintek F81216: COM3, COM4 (RS232)
Digital IO	4 in & 4 out
Keyboard/Mouse	Supports PS/2 Keyboard/Mouse Connector
Expansion Slots	PIC-E (x16) slot x1 and Mini PCI socket x1, Mini PCIe X1
Edge Connector	PS/2 Connector x1 for keyboard/mouse Gbit LAN RJ-45 + dual USB stack connector 10/100 LAN RJ45 + dual USB stack connector DB9 x1 for COM 1; DB15 x1 for VGA RCA Jack x1 +S-Video for TV-Out SPDIF-OUT connector x1 RCA Jack 3x1 for Audio (Line-Out, Line-In & Mic)
Onboard Header/ Connector	40 pins box-header x1 for IDE1; 26 pins header x1 for Slim Floppy CF Connector x1 @ solder side 10 pins pin-header x1 for Digital I/O;9 pins pin-header x3 for COM2~COM4 8 pins pin-header x 1 for USB 5,6; 5 pins pin-header x 1 for IrDA DF13 Connector x2 for LVDS; 8 pins pin-header x1 for HDTV 7 pins pin-header x1 for audio Line-Out & Mic 4 pins pin-header x1 for built-in speaker SATA connector x2 for 2 SATA ports
Watchdog Timer	Yes (256 segments, 0, 1, 2…255 sec/min)
System Voltage	+5V, +3.3V, +12V, -12V, 5VSB (2A)
Others	Modem Wakeup, LAN Wakeup
Board Size	170mm x 170mm (Mini ITX)

Board Dimensions

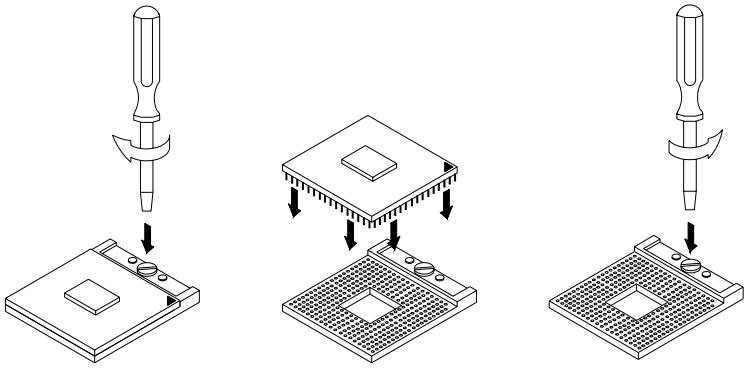


Installations

This section provides information on how to use the jumpers and connectors on the MB899X in order to set up a workable system. The topics covered are:

Installing the CPU

The MB899X board supports a Socket 478MT (Napa) processor socket for Intel® Core™ 2 Duo , Intel® Core™ Duo and Intel® Core™ Solo mobile processors.
The processor socket comes with a screw to secure the processor. As shown in the left picture below, loosen the screw first before inserting the processor. Place the processor into the socket by making sure the notch on the corner of the CPU corresponds with the notch on the inside of the socket. Once the processor has slide into the socket, fasten the screw. Refer to the figures below.



***NOTE**: Ensure that the CPU heat sink and the CPU top surface are in total contact to avoid CPU overheating problem that would cause your system to hang or be unstable.*

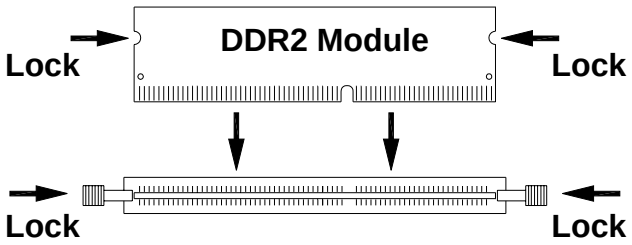
Installing the Memory

The MB899X board supports two DDR2 memory socket for a maximum total memory of 4GB in DDR2 memory type.

Installing and Removing Memory Modules

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

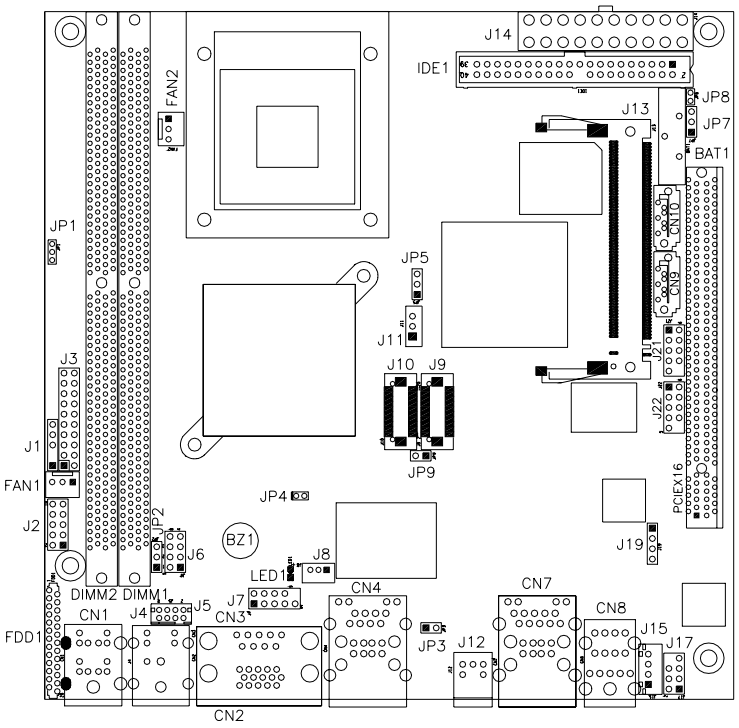
1. Hold the DDR2 module so that the key of the DDR2 module align with those on the memory slot.
2. Gently push the DDR2 module in an upright position until the clips of the slot close to hold the DDR2 module in place when the DDR2 module touches the bottom of the slot.
3. To remove the DDR2 module, press the clips with both hands.



Setting the Jumpers

Jumpers are used on MB899X 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 on MB899X and their respective functions.

Jumper Locations on MB899X



JP2: CPU FSB Selection (reserved)

JP2	CPU FSB
	533MHz
	667MHz

JP5: LCD Panel Power Selection

JP5	LCD Panel Power
	3.3V
	5V

JP7: Clear CMOS Setting

JP7	Setting
	Normal
	Clear CMOS

JP8: CompactFlash Slave/Master Selection

JP8	CF Setting
	Master
	Slave

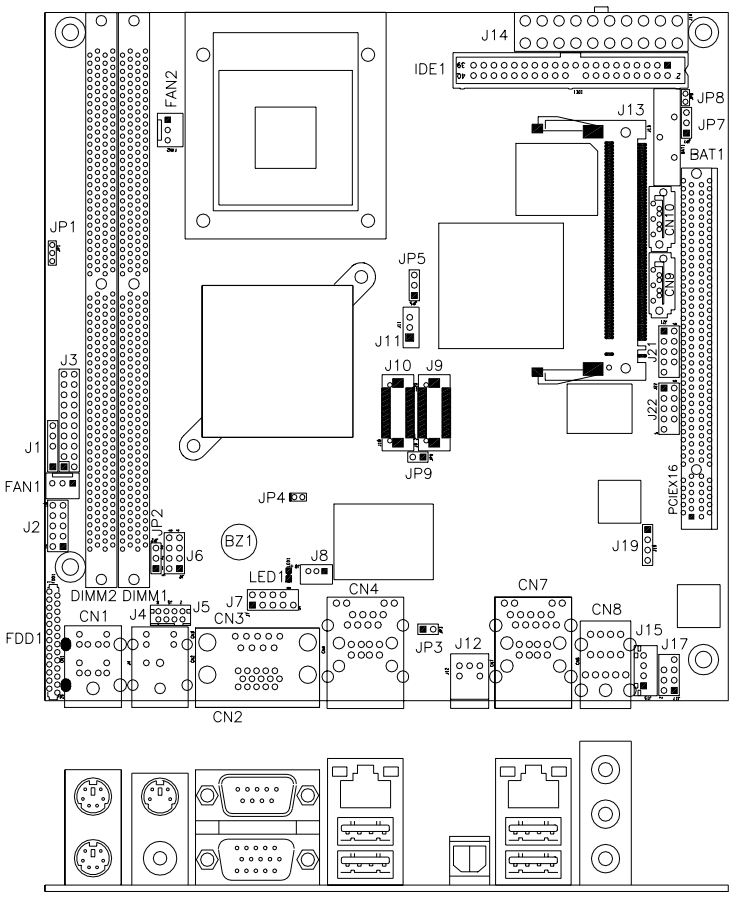
JP9: Internal/External VGA Selection

JP9	VGA Source
	Internal
	External

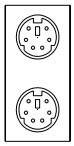
Connectors on MB899X

The connectors on MB899X allows you to connect external devices such as keyboard, floppy disk drives, hard disk drives, printers, etc. The following table lists the connectors on MB899X and their respective functions.

Connector Locations on MB899X



CN1: PS/2 Keyboard and PS/2 Mouse Connectors

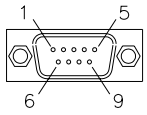


PS/2 Mouse

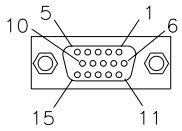
PS/2 Keyboard

Signal Name	Keyboard	Mouse	Signal Name
Keyboard data	1	1	Mouse data
N.C.	2	2	N.C.
GND	3	3	GND
5V	4	4	5V
Keyboard clock	5	5	Mouse clock
N.C.	6	6	N.C.

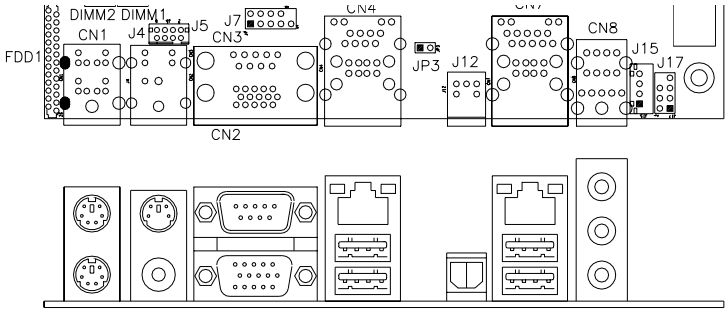
CN2, CN3: COM1 and VGA Connector



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



Signal Name	Pin #	Pin #	Signal Name
Red	1	2	Green
Blue	3	4	N.C.
GND	5	6	GND
GND	7	8	GND
N.C.	9	10	GND
N.C.	11	12	N.C.
HSYNC	13	14	VSYNC
NC	15		



CN4: 10/100 RJ-45 and USB1/2 Ports

J12: SPDIF Out Connector

CN6, CN7: GbE RJ-45 and USB3/4 Ports

CN8: Audio Connector

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

CN9, CN10: Serial ATA Connectors

FAN1: System Fan Power Connector

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

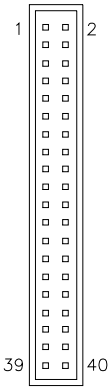
Pin #	Signal Name
1	Ground
2	+12V
3	Rotation detection

FAN2: CPU Fan Power Connector

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

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

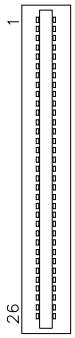
IDE1: IDE Connector



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	Protect pin
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

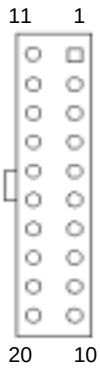
FDD1: Floppy Drive Connector

FDD1is a slim 26-pin connector and will support up to 2.88MB FDD.



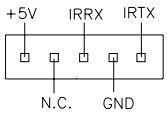
Signal Name	Pin #	Pin #	Signal Name
VCC	1	2	INDEX
VCC	3	4	DRV_SEL
VCC	5	6	DSK_CH
NC	7	8	NC
NC	9	10	MOTOR
DINST	11	12	DIR
NC	13	14	STEP
GND	15	16	WDATA
GND	17	18	WGATE
GND	19	20	TRACK
NC	21	22	WPROT
GND	23	24	RDATA
GND	25	26	SIDE

J14: ATX Power Supply Connector



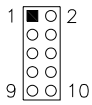
Signal Name	Pin #	Pin #	Signal Name
3.3V	11	1	3.3V
-12V	12	2	3.3V
Ground	13	3	Ground
PS-ON	14	4	+5V
Ground	15	5	Ground
Ground	16	6	+5V
Ground	17	7	Ground
-5V	18	8	Power good
+5V	19	9	5VSB
+5V	20	10	+12V

J1: IrDA Connector



Pin #	Signal Name
1	+5V
2	No connect
3	Ir RX
4	Ground
5	Ir TX

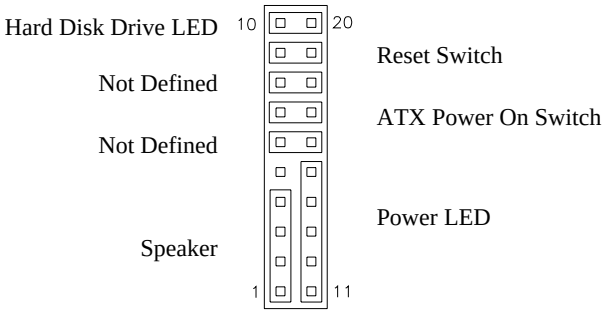
J2: Digital I/O



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

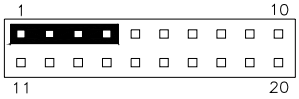
J3: System Function Connector

J3 provides connectors for system indicators that provide light indication of the computer activities and switches to change the computer status. J3 is a 20-pin header that provides interfaces for the following functions.



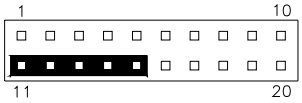
Speaker: Pins 1 - 4

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

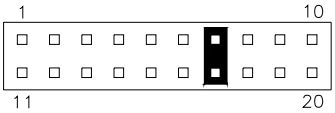
Power LED: Pins 11 - 15



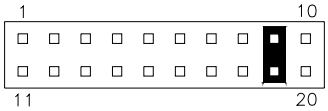
Pin #	Signal Name
11	Power LED
12	No connect
13	Ground
14	No connect
15	Ground

ATX Power ON Switch: Pins 7 and 17

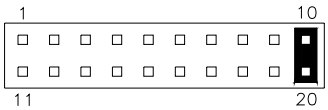
This 2-pin connector is an “ATX Power Supply On/Off Switch” on the system that connects to the power switch on the case. When pressed, the power switch will force the system to power on. When pressed again, it will force the system to power off.



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

J4: TV-OUT (S-Video & Composite) Connector

J5: TV-OUT (Y, Pr, Pb) Connector

The pin assignments of the TV out connector are as follows:

Signal Name	Pin	Pin	Signal Name
NC	1	2	NC
SL/Y	3	4	Ground
SC/Pr	5	6	Ground
CVBS/Pb	7	8	Ground

CVBS : Composite signal
Pb : Component Chrominance (Pb) analog signal

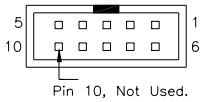
SL : S-Video Luminance analog signal
Y : Component Luminance (Y) analog signal

SC : S-Video Chrominance analog signal
Pr : Component Chrominance (Pr) analog signal

J6: USB5/6 Port Pin Header

Signal Name	Pin	Pin	Signal Name
Vcc	1	5	Ground
D-	2	6	D+
D+	3	7	D-
Ground	4	8	Vcc

J7,J21,J22: COM2,3,4 Serial Port



COM2

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

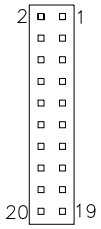
J8: Wake On LAN Connector

J8 is a 3-pin header for the Wake On LAN function. Wake On LAN will function properly only with an ATX power supply with 5VSB that has 200mA.

Pin #	Signal Name
1	+5VSB
2	Ground
3	-PME

J9, J10: LVDS Connectors (1st channel, 2nd channel)

The LVDS connectors on board consist of the first channel (J9) and second channel (J10) and supports 18-bit or 36-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
NA	10	9	NA
TX2-	12	11	TX2+
Ground	14	13	Ground
TXC-	16	15	TXC+
5V/3.3V	18	17	ENABKL
+12V	20	19	+12V

J11: LCD Backlight Connector

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

J13: Mini PCI Connector

U36: Mini PCI- E(x1) Connector (bottom side)

J15: Speaker Connector

The J15 connector supports 2W/8ohm stereo audio power amplifier.

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

J17: Front Audio Connector

Signal Name	Pin	Pin	Signal Name
Rear Audio R	1	2	Rear Audio L
Front Audio R	3	4	Front Audio L
Mic In	5	6	VREF Out
Ground	7	8	

REMARKS: To use the front audio connector, the jumpers on pin 1-3 and pin 2-4 must be removed.

PCIEX16: PCI-E(x16) Slot

J19: CD-In Pin Header

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

J20: Compact Flash Connector

This page is intentionally left blank.

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 Setup

"Standard CMOS Setup" choice allows you to record some basic hardware configurations in your computer system and set the system clock and error handling. If the motherboard 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

		Item Help
Date (mm:dd:yy)	Wed, Apr 28, 2004	Menu Level >
Time (hh:mm:ss)	00 : 00 : 00	
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, 3.5 in.	
Drive B	None	
Video	EGA/VGA	
Halt On	All Errors	
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.

Drive A / Drive B

These fields identify the types of floppy disk drive A or drive B that has been installed in the computer. The available specifications are:

360KB	1.2MB	720KB	1.44MB	2.88MB
5.25 in.	5.25 in.	3.5 in.	3.5 in.	3.5 in.

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	
Quick Power On Self Test	Enabled	
First Boot Device	Floppy	
Second Boot Device	Hard Disk	
Third Boot Device	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	Yes	
Small Logo (EPA) Show	Enabled	

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 and L2 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.

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-Floppy*, *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. The default setting is ***Enabled***.

Advanced Chipset Features

This Setup menu controls the configuration of the chipset.

Phoenix - AwardBIOS CMOS Setup Utility
Advanced Chipset Features

		ITEM HELP
DRAM Timing Selectable	By SPD	Menu Level >
CAS Latency Time	4	
DRAM RAS# to CAS# Delay	4	
DRAM RAS# Precharge	4	
Precharge delay (tRAS)	12	
System Memory Frequency	533MHZ	
SLP_S4# Assertion Width	1 to 2 Sec	
System BIOS Cacheable	Enabled	
Video BIOS Cacheable	Disabled	
Memory Hole at 15M-16M	Disabled	
PCI Express Root Port Func	Press Enter	
** On-Chip VGA Setting **		
PEG/On Chip VGA Control	Auto	
On-Chip Frame Buffer Size	8MB	
DVMT Mode	DVMT	
DVMT/FIXED memory Size	128MB	
Onboard PCI-E LAN	Enable	
LAN PXE Option ROM	All Disable	

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 ***4***.

Precharge Delay (tRAS)
The default setting for the Precharge Delay is ***12***.

System Memory Frequency
The default setting is ***533MHz***.

SLP_S4# Assertion Width
The default setting is ***1 to 2 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:

PEG/On Chip VGA Control: Auto
On-Chip Frame Buffer Size: 8MB
DVMT Mode: DVTM
DVMT/Fixed Memory Size: 128MB

Onboard PCI-E LAN
By default, this setting is enabled.

LAN PXE Option ROM
By default, this setting is disabled. Other selections include ICH6 Integrated LAN and Marvell PCI-E LAN.

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

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

Phoenix - AwardBIOS CMOS Setup Utility
OnChip IDE Device

IDE HDD Block Mode	Enabled	ITEM HELP
On-chip Primary PCI IDE	Enabled	
IDE Primary Master PIO	Auto	Menu Level >
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 ***		
On-Chip Serial ATA	Auto	
PATA IDE Mode	Secondary	
SATA port	P0, P2 is Primary	

Phoenix - AwardBIOS CMOS Setup Utility
Onboard Device

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

Phoenix - AwardBIOS CMOS Setup Utility
SuperIO Device

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

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.

On-chip Primary PCI IDE Enabled
This field, by default, 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 includes On-Chip Serial ATA (Auto), PATA IDE Mode (Secondary) and SATA Port (PO, 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. ***Please update your system to Windows 2000 SP4 or Windows XP SP2.***

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

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 Port

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

Serial Port 1	3F8/IRQ4
Serial Port 2	2F8/IRQ3

UART Mode Select

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

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

Phoenix - AwardBIOS CMOS Setup Utility
Power Management Setup

		ITEM HELP
ACPI Function	Enabled	
ACPI Suspend	S1(POS)	
RUN VGABIOS if S3 Resume	Auto	Menu Level >
Power Management	User Define	
Video Off Method	DPMS	
Video Off In Suspend	Yes	
Suspend Type	Stop Grant	
Modem Use IRQ	3	
Suspend Mode	Disabled	
HDD Power Down	Disabled	
Soft-Off by PWR-BTTN	Instant-Off	
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	Disabled	
Primary IDE 1	Disabled	
Secondary IDE 0	Disabled	
Secondary IDE 1	Disabled	
FDD, COM, LPT Port	Disabled	
PCI PIRQ[A-D] #	Disabled	

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***.

Suspend Mode
When enabled, and after the set time of system inactivity, all devices except the CPU will be shut off.

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.

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

Init Display First	PCI Slot	ITEM HELP
Reset Configuration Data	Disabled	
		Menu Level
Resources Controlled By	Auto (ESCD)	
IRQ Resources	Press Enter	Select Yes if you are using a Plug and Play capable operating system Select No if you need the BIOS to configure non-boot devices
PCI/VGA Palette Snoop	Disabled	
INT Pin 1 Assignment	Auto	
INT Pin 2 Assignment	Auto	
INT Pin 3 Assignment	Auto	
INT Pin 4 Assignment	Auto	
INT Pin 5 Assignment	Auto	
INT Pin 6 Assignment	Auto	
INT Pin 7 Assignment	Auto	
INT Pin 8 Assignment	Auto	
PCI Express relative items		
Maximum Payload Size	4096	

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 4096.

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

		ITEM HELP
Shutdown Temperature	Disabled	
CPU Warning Temperature	Disabled	Menu Level >
Current System Temp	45°C/113°F	
Current CPU Temp	45°C/113°F	
System FAN Speed	5400 RPM	
CPU FAN Speed	5400 RPM	
Vcore(V)	1.02 V	
12 V	1.32 V	
1.8V	1.8V	
-5V	-5.02V	
+5V	5.25 V	
-12V	-12.59	
3.3V	3.37V	
VBAT (V)	3.21 V	
5VSB(V)	5.67 V	
Smart Fan2 Temperature	Disabled	
Smart Fan2 Tolerance Value	5	

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

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.

Smart Fan2 Temperature

This field enables or disables the smart fan feature. At a certain temperature, the fan starts turning. Once the temperature drops to a certain level, it stops turning again.

Smart Fan Tolerance Value

The default value is 5.

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 Modulated	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.

Drivers Installation

This section describes the installation procedures for software and drivers under the Windows 2000 and Windows XP. 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:

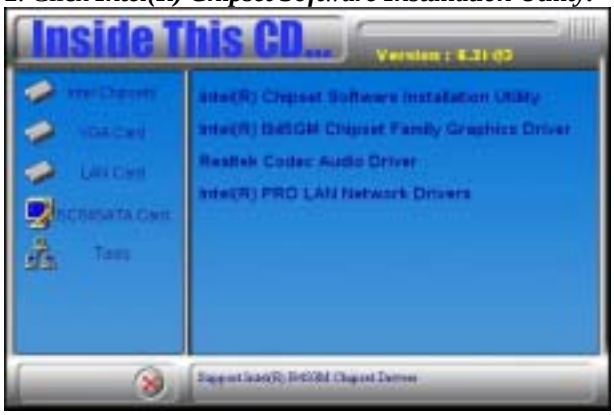
IMPORTANT NOTE:
After installing your Windows operating system (Windows 2000/ XP), you must install first the Intel Chipset Software Installation Utility before proceeding with the drivers installation.

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/XP.

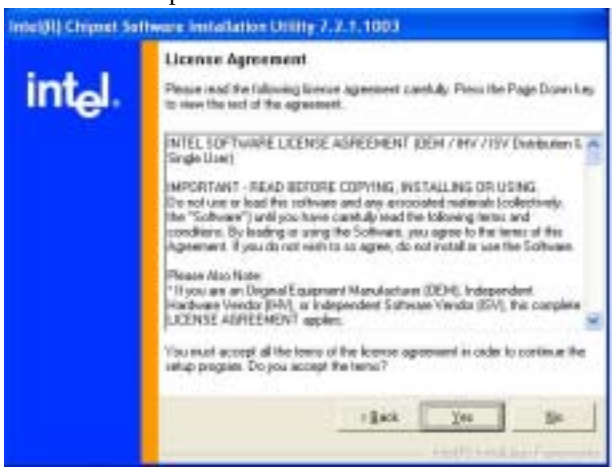
1. Insert the CD that comes with the board. Click ***Intel Chipsets*** and then ***Intel(R) I945GMEChipset Drivers***.

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

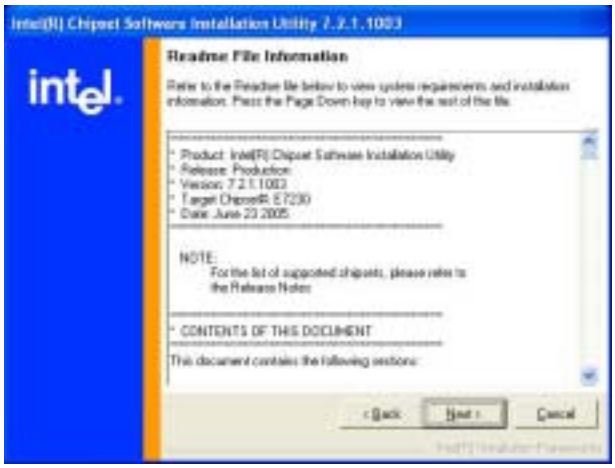


3. When the Welcome screen appears, click ***Next*** to continue.

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.



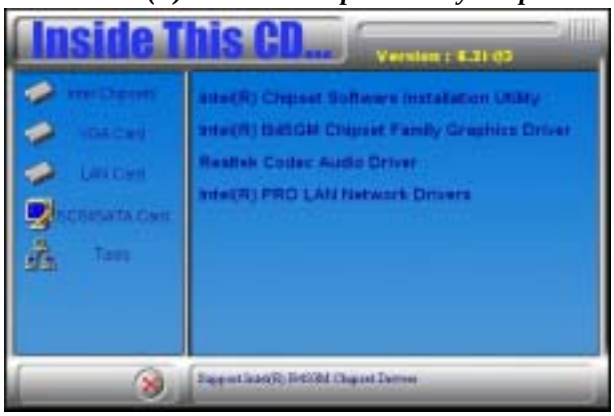
6. The Setup process is now complete. Click ***Finish*** to restart the computer and for changes to take effect. When the computer has restarted, the system will be able to find some devices. Restart your computer when prompted.

VGA Drivers Installation

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

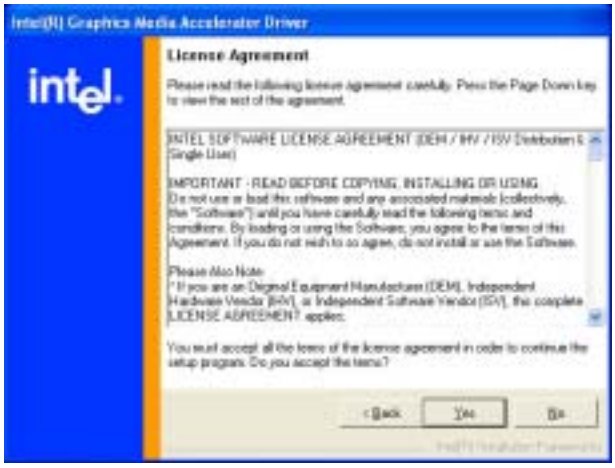
1. Insert the CD that comes with the motherboard. Click ***Intel Chipsets*** and then ***Intel(R) I945GMEChipset Drivers***.

2. Click ***Intel(R) I945GMEChipset Family Graphics Driver.***

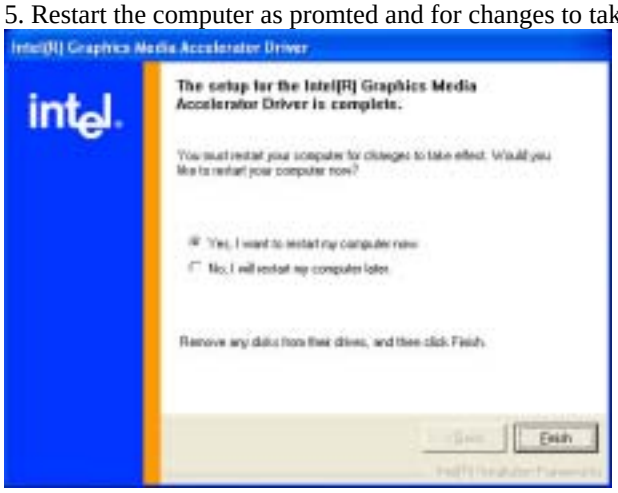


3. When the Welcome screen appears, click ***Next*** to continue.

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



5. Restart the computer as promted and for changes to take effect.



IMPORTANT NOTE:
When you have restarted the computer, your computer screen will be blank. At this point, press CTRL-ALT-F1 simultaneously, if you are using CRT monitor. If you are using LVDS LCD panel, press CTRL-ALT-F3. If you are using DVI monitor, press CTRL-ALT-F4.

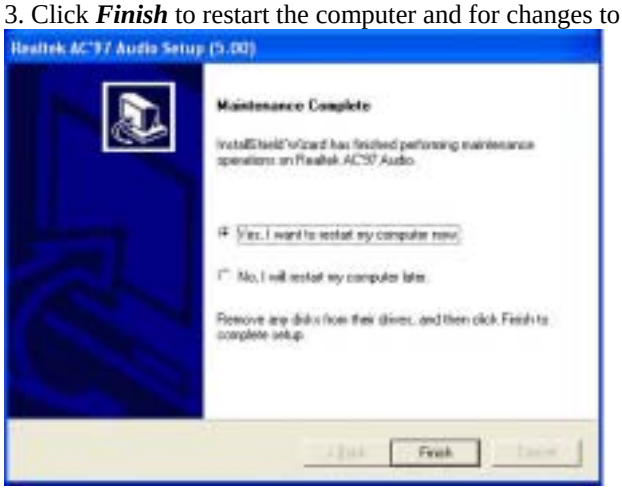
AC97 Codec Audio Driver Installation

Follow the steps below to install the Realtek AC97 Codec Audio Drivers.

1. Insert the CD that comes with the motherboard. Click ***Intel Chipsets*** and then ***Intel(R) I945GMEChipset Drivers***.

2. Click ***Realtek AC'97 Codec Audio Driver.***

3. Click ***Finish*** to restart the computer and for changes to take effect. .



LAN Drivers Installation

Follow the steps below to complete the installation of the Intel PRO LAN drivers.

1. Insert the CD that comes with the motherboard. Click ***Intel Chipsets*** and then ***Intel(R) I945GMEChipset Drivers***, then ***Intel(R) PRO LAN Network Drivers***.

2. Click ***Install Base Software*** to continue.

3. When prompted, please to restart the computer for new settings to take effect.

Follow the steps below to install the **Marvell Gigabit LAN** drivers.

1. Insert the CD that comes with the motherboard. Click ***LAN Card*** and then ***Marvell LAN Controller Driver***.

2. Click Next when the InstallShield Wizard welcome screen appears.

3. Click Next to agree with the license agreement.

4. Click Next when the Readme Information screen appears to proceed with the drives installation process.

5. When the Installation is complete, click Finish for the changes to take effect.

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();        }
}
//---------------------------------------------------------------------------
```

This page is intentionally left blank.