

# **IB895**

**Intel® Cedarview-D or  
Cedarview-M / NM10  
3.5" Disk Size SBC**

# **USER'S MANUAL**

**Version 1.1A**

---

## **Acknowledgments**

AMI is a registered trademark of American Megatrends Inc.

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

Intel and Intel® Sandy Bridge DC/QC Processor are registered trademarks of Intel Corporation.

Microsoft Windows is a registered trademark of Microsoft Corporation.

Winbond a registered trademark of Winbond Electronics Corporation.

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

---

# Table of Contents

<b>Introduction</b> .....	<b>1</b>
Product Description.....	1
Checklist.....	2
IB895 Specifications .....	3
Board Dimensions .....	4
<b>Installations</b> .....	<b>5</b>
Installing the Memory.....	6
Setting the Jumpers.....	7
Connectors on IB895.....	11
<b>BIOS Setup</b> .....	<b>19</b>
<b>Drivers Installation</b> .....	<b>39</b>
Intel Chipset Software Installation Utility.....	40
VGA Drivers Installation.....	41
Realtek HD Audio Driver Installation .....	42
LAN Drivers Installation .....	43
<b>Appendix</b> .....	<b>44</b>
A. I/O Port Address Map.....	44
B. Interrupt Request Lines (IRQ).....	46
C. Digital I/O Sample Code.....	47
D. Watchdog Timer Configuration.....	52

---

*This page is intentionally left blank.*

# Introduction

## Product Description

---

---

The IB895 is a 3.5-inch single board computer based on the Intel® Atom Cedarview chipset. The Cedar Trail is a platform that uses the Intel Cedarview-D or Cedarview -M Processor and Intel NM10 Express Chipset family in the desktop platforms.

The role of the processor in the system is to manage the flow of information between the following interfaces: DDR3 System Memory interface, DVI display interface, VGA graphics interface, Low Voltage Differential Signaling (LVDS) and the Direct Media Interface (DMI). The processor supports single channel, one DDR3 SODIMMs.

The Intel® processors provide advanced performance in both computing and graphics quality. This meets the requirement of customers in the gaming, POS, digital signage and server market segment.

The processor is built on 32-nanometer Hi-K process technology with Micro-FCBGA11 packaging.

### IB895 FEATURES:

- Supports Intel® Cedarview -D and Cedarview-M processors
- One DDR3 SO-DIMM, 800/1066 MHz, Max. 2GB/4GB memory\*
- Integrated Graphics for DVI-I, LVDS displays
- 2 x SATA Connector
- 4x COM Port Connector
- 2 x Mini-PCIE(x1) slot (w/ USB support)
- 2x GbE (RJ-45) Connector
- 1x 24V to 12V DC-IN Power Connector

**\* Note: IB895 with Intel Atom D2550 supports up to 4GB memory and IB895N with Intel Atom N2600 supports up to 2GB memory.**

### **Checklist**

---

---

Your IB895 package should include the items listed below.

- The IB895 Embedded Flex Motherboard
- This User's Manual
- 1 CD containing chipset drivers and flash memory utility

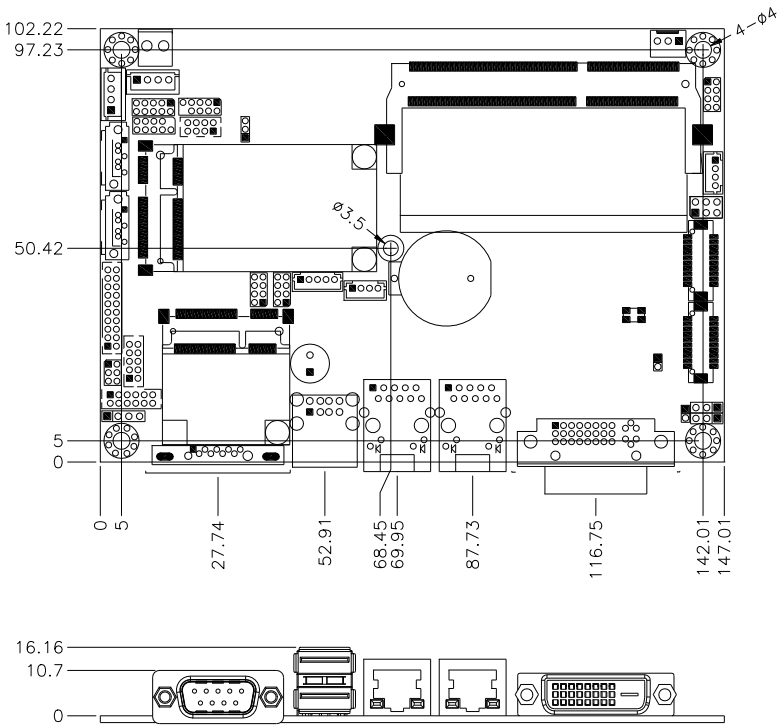
## IB895 Specifications

<b>Product Name</b>	IB895 [default silk screen model # on PCB] IB895N(N2600 onboard)
<b>Form Factor</b>	3.5"
<b>CPU Type/Speed</b>	Intel® Atom™ DC <b>D2550</b> / 1MB cache / 1.86 GHz [TDP=10W] (IB895) Intel® Atom™ DC <b>N2600</b> / 1MB cache / 1.60 GHz [TDP=3.5W] (IB895N) Package = FCBGA437 [ 22 mm x 22 mm] ; Cores / Threads = 2 / 4
<b>Chipset</b>	Intel® CG82NM10 PCH [TDP = 2.1W] Package = BGA360, 17mm x 17 mm
<b>BIOS</b>	AMI BIOS, support ACPI Function
<b>Memory</b>	Intel® Atom™ on-die memory controller 1x DDR3-1066 SO-DIMM socket [Horizontal type] <b>Maximum memory: 4GB for D2550; 2GB for N2600</b> Non-ECC, Unbuffered, 1.5V
<b>VGA</b>	Intel® GMA 3650 (Gfx freq @ 640MHz) for D2550 Intel® Graphics Media Acc GMA elerator 3600 (Gfx freq @ 400MHz) for N2600 Supports DirectX 9 Graphics, OpenGL 3.0 DVI-I x 1 (Thru VGA + ASM1422 for DVI-D via D2550 or N2600)
<b>LVDS</b>	24-bit dual channel via CH7511thru eDP
<b>LAN</b>	Intel 82583V PCIe Gigabit LAN x 2
<b>USB</b>	NM10 built-in USB 2.0 host controller, support 6 ports
<b>Serial ATA Ports</b>	NM10 built-in SATA II controller, supports 2 ports
<b>Audio</b>	Intel® NM10 PCH built-in HD Audio controller + Realtek ALC269QHD Codec w/class-D speaker amplifier(2.3W per channel @ 5V power supply) [7mm x 7mm @ 48-QFN] ; support 2-channel audio out + amp
<b>LPC I/O</b>	Nuvoton NCT6627UD [128-pin QFP, 14x14x1.4mm] - COM #1 (RS232/422/485) support ring-in with power @ 500 mA (selectable for 5V or 12V) [EXAR SP339EER1 232/422/485 transceiver x 1 for jumper-less] - COM #2 (RS232 only) - COM #3 / COM #4 (RS-232 only) [Hardware Monitor] 2 x Thermal inputs; 2 x Voltage monitoring 1 x Fan Header(DC Fan type)
<b>Digital IO</b>	4 in & 4 out
<b>Keyboard/ Mouse Connector</b>	Pin headers
<b>Expansion Slots</b>	Mini PCI-e socket x 2 (1x Full-sized+1xHalf-sized) **Full-sized MiniPCIe(1x) support mSATA**
<b>Edge Connector</b>	DVI-I, 2x RJ45, 2x USB, COM1
<b>Onboard Header/Connector</b>	Keyboard/mouse, 4x USB 2.0, 2x DF13 for 24-bit dual channel LVDS, Audio, speaker, COM2, COM3/4, LPC (80-port card debugging purpose), 2x Mini PCI-e(1x), smart battery, backlight/brightness control, 2x power connector SATA HDD [JST type]

# INTRODUCTION

<b>Watchdog Timer</b>	Yes (256 segments, 0, 1, 2...255 sec/min)
<b>Power Connector</b>	+12V ~ +24V DC-in
<b>Others</b>	- i-Smart function (TI MSP430G2433 MCU) - AT24C02C EEPROM [SO8 type] via SMBus (Optional)
<b>OS Support</b>	- Windows 7 (32-bit only) - Linux
<b>RoHS</b>	Yes
<b>Board Size</b>	102mm x 147mm

## Board Dimensions





## Installations

This section provides information on how to use the jumpers and connectors on the IB895 in order to set up a workable system. The topics covered are:

Installing the Memory .....	6
Setting the Jumpers.....	7
Connectors on IB895 .....	11

### Installing the Memory

---

---

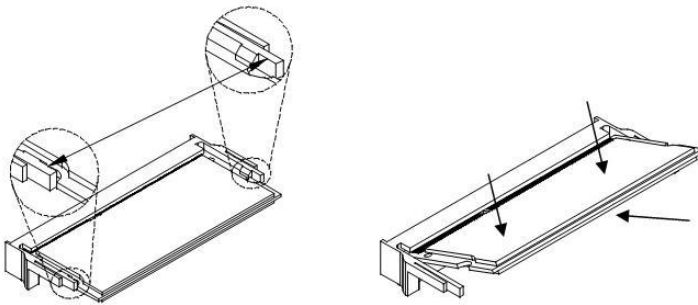
The IB895 board supports one DDR3 memory socket for a maximum total memory of 2GB or 4GB DDR3 memory type.

**Note:** *IB895 with Intel Atom DC D2550 supports up to 4GB and IB895N with Intel Atom DC N2600 supports up to 2GB.*

#### Installing and Removing Memory Modules

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

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



---

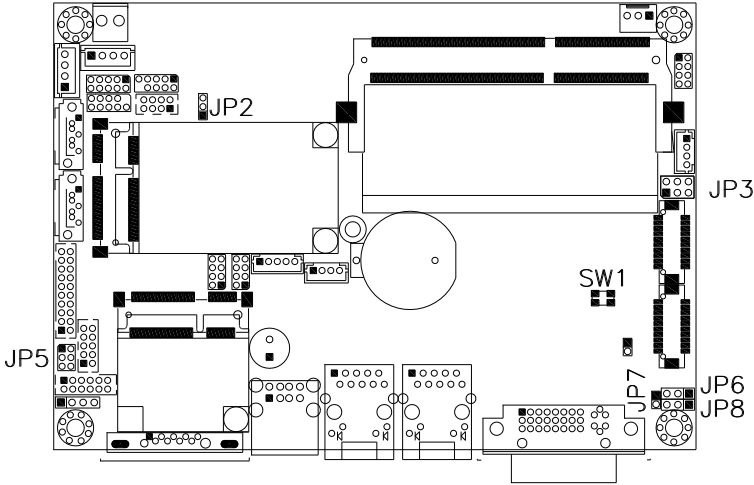
## **Setting the Jumpers**

---

Jumpers are used on IB895 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 IB895 and their respective functions.



Jumper Locations on IB895 .....	8
JP2: Clear CMOS Contents.....	9
JP3: BL Voltage Setting .....	9
JP5: COM1 RS232 RI/+5V/+12V Power Setting.....	9
JP6: LVDS Panel Power Selection .....	9
JP7: BL_ADJ_LEVEL Setting.....	10
JP8: Factory use only .....	10
SW1 : LVDS Panel Type setting .....	10

**Jumper Locations on IB895**

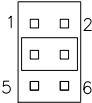


Jumpers on IB895.....	Page
JP2: Clear CMOS Contents .....	9
JP3: BL Voltage Setting .....	9
JP5: COM1 RS232 RI/+5V/+12V Power Setting.....	9
JP6: LVDS Panel Power Selection.....	9
JP7: BL_ADJ_LEVEL Setting .....	10
JP8: Factory use only.....	10
SW1 : LVDS Panel Type setting.....	10

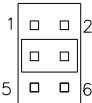
**JP2: Clear CMOS Contents**

JP2	Setting	Function
	Pin 1-2 Short/Closed	Normal
	Pin 2-3 Short/Closed	Clear CMOS



**JP3: BL Voltage Setting**

JP3	Setting	Function
	Pin 1-2 Short/Closed	+3.3V
	Pin 3-4 Short/Closed	+5V
	Pin 5-6 Short/Closed	+12V(Default)



**JP5: COM1 RS232 RI/+5V/+12V Power Setting**

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

**JP6: LVDS Panel Power Selection**

JP6	Setting	Panel Voltage
	Pin 1-2 Short/Closed	3.3V (default)
	Pin 2-3 Short/Closed	5V

**JP7: BL\_ADJ\_LEVEL Setting**

JP7	Setting	Voltage
	Open	3.3V
	Close	5V(default)

**JP8: Factory use only****SW1: LVDS Panel Type Setting**

SW1-4	SW1-3	SW1-2	SW1-1	Panel Type
ON	ON	ON	ON	800*600 18bit 1ch
ON	ON	ON	OFF	1024*768 18bit 1ch
ON	ON	OFF	ON	1024*768 24bit 1ch
ON	ON	OFF	OFF	1280*768 18bit 1ch
ON	OFF	ON	ON	1280*800 18bit 1ch
ON	OFF	ON	OFF	1280*960 18bit 1ch
ON	OFF	OFF	ON	1280*1024 24bit 2ch
ON	OFF	OFF	OFF	1366*768 18bit 1ch
OFF	ON	ON	ON	1366*768 24bit 1ch
OFF	ON	ON	OFF	1440*900 24bit 2ch
OFF	ON	OFF	ON	1440*1050 24bit 2ch
OFF	ON	OFF	OFF	1600*900 24bit 2ch
OFF	OFF	ON	ON	1680*1050 24bit 2ch
OFF	OFF	ON	OFF	1600*1200 24bit 2ch
OFF	OFF	OFF	ON	1920*1080 24bit 2ch
OFF	OFF	OFF	OFF	1920*1200 24bit 2ch

---

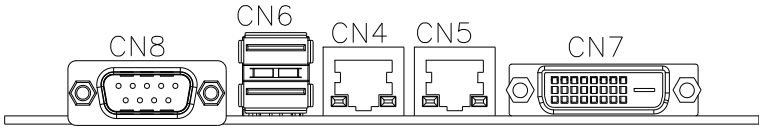
## Connectors on IB895

---

Connector Locations on IB895 .....	12
CN4, CN5: Gigabit LAN (Intel 82583V) .....	13
CN6: USB 1/2 Connector .....	13
CN7: VGA DVI-I Connector .....	13
CN8: DB9 Connector .....	13
CN1, CN3: SATA Connector .....	14
CN2: LCD Backlight Connector .....	14
J1: Board Input Power Connector .....	14
J2, J3: SATA HDD Power Connector .....	14
J5: Digital I/O Connector .....	14
J6: SPI Flash Connector (factory use only) .....	14
J7: Keyboard & Mouse Connector (DF11 Connector) .....	15
J11: Smart Battery Interface Connector .....	15
J12, J13: USB3/4/5/6/7/8 Connector .....	15
J15: COM3, COM4 Serial Port (DF11 Connector) .....	16
J16: COM2/RS232 Serial Port .....	16
J18: Audio Connector (DF11 Connector) .....	16
J22: Amplify Connector .....	17
J23: LVDS EEPROM Flash Connector (factory use only) .....	17
CH1, CH2: LVDS Connectors .....	17
JP1: Front Panel Connector .....	18







**CN4, CN5: Gigabit LAN (Intel 82583V)**

This RJ45 LAN connector (CN4 only) features for EUP LAN wakeup.

**CN6: USB 1/2 Connector**

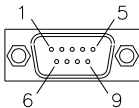
**CN7: VGA DVI-I Connector**

**CN8: DB9 Connector**

(COM1) is a DB-9 connector.

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

COM1 is jumper-less for RS-232, RS-422 and RS-485 and configured with BIOS Selection.



Pin #	Signal Name		
	RS-232	R2-422	RS-485
1	DCD	RX+	DATA-
2	RX	RX-	DATA+
3	TX	TX+	NC
4	DTR	TX-	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

**CN1, CN3: SATA Connectors****CN2: LCD Backlight Connector**

Pin #	Signal Name
1	Backlight Power
2	Backlight Enable
3	Brightness Control
4	Ground

**J1: Board Input Power Connector**

Pin #	Signal Name
1	+12V to +24V
2	GND

**J2, J3: SATA HDD Power Connector**

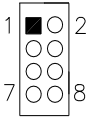
Pin #	Signal Name
1	+5V
2	Ground
3	Ground
4	+12V

**J4: Debug 80 Port Connector (factory use only)****J5: Digital I/O Connector**

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

**J6: SPI Flash Connector (factory use only)**

**J7: Keyboard & Mouse Connector (DF11 Connector)**




Signal Name	Pin #	Pin #	Signal Name
VCC	1	2	VCC
MDA	3	4	KBDA
MCL	5	6	KBCL
Ground	7	8	Ground

**J8: DDR3 SO-DIMM Socket**

**J9: Mini PCIE Connector(Support mSATA)**

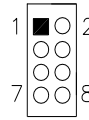
*J9 also supports mSATA. However, when J9 is used for mSATA, then CN3 SATA port cannot be used. Only one of them can be used at one time to support SATA.*

**J11: Smart Battery Interface Connector**



Pin #	Signal Name
1	RST
2	EXTSMI
3	Ground
4	DATA
5	CLK

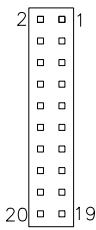
**J12, J13: USB3/4/5/6/7/8 Connector**



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

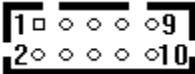
**J14: MCU Flash Connector (factory use only)**

**J15: COM3, COM4 Serial Port (DF11 Connector)**



Signal Name	Pin #	Pin #	Signal Name
DSR3	2	1	DCD3
RTS3	4	3	RXD3
CTS3	6	5	TXD3
RI3	8	7	DTR3
NC	10	9	Ground
DSR4	12	11	DCD4
RTS4	14	13	RXD4
CTS4	16	15	TXD4
RI4	18	17	DTR4
NC	20	19	Ground

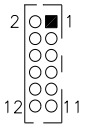
**J16: COM2/RS232 Serial Port**



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

**J17: Mini PCIE Connector**

**J18: Audio Connector (DF11 Connector)**



Signal Name	Pin #	Pin #	Signal Name
LINEOUT_R	2	1	LINEOUT_L
Ground	4	3	JD_FRONT
LINEIN_R	6	5	LINEIN_L
Ground	8	7	JD_LINEIN
MIC-R	10	9	MIC_L
Ground	12	11	JD_MIC1

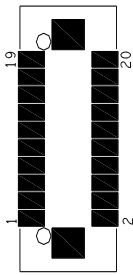
**J22: Amplify Connector**



Pin #	Signal Name
1	OUTL+
2	OUTL-
3	OUTR-
4	OUTR+

**J23: LVDS EEPROM Flash Connector (factory use only)**

**CH1, CH2: LVDS Connectors**



Signal Name	Pin #	Pin #	Signal Name
N.C	19	20	N.C
ENABLE	17	18	LCD_PWR
CLK+	15	16	CLK-
GND	13	14	GND
LD2+	11	12	LD2-
LD3+	9	10	LD3-
GND	7	8	LCD_PWR
LD1+	5	6	LD1-
GND	3	4	GND
LD0+	1	2	LD0-

**JP1: Front Panel Connector**

The following table shows the pin outs of the 2x4 pin header

1	□	○	2				
	○	○					
	○	○					
7	○	○	8				

Signal Name	Pin #	Pin #	Signal Name
Ground	1	2	PWR_SW
PWR_LED+	3	4	PWR_LED-(GND)
HDD_LED+	5	6	HDD_LED-
Ground	7	8	RESET

JP1 provides connectors for system indicators that provide light indication of the computer activities and switches to change the computer status. JP1 is an 8-pin header that provides interfaces for the following functions.

**ATX Power ON Switch: Pins 1 and 2**

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.

**Power LED: Pins 3 and 4**

Pin #	Signal Name
3	LED(+)
4	LED(-)

**Hard Disk Drive LED Connector: Pins 5 and 6**

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
5	LED(+)
6	LED(-)

**Reset Switch: Pins 7 and 8**

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.

# BIOS Setup

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

BIOS Introduction.....	20
BIOS Setup.....	20
Advanced Settings.....	22
Chipset Settings.....	31
Boot Settings.....	36
Security Settings.....	37
Save & Exit Settings.....	38

### **BIOS Introduction**

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

### **BIOS Setup**

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

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

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

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



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

**System Date**

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

**System Time**

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

## Advanced Settings

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

Aptio Setup Utility					
Main	Advanced	Chipset	Boot	Security	Save & Exit
	Legacy OpROM Support				
	Launch PXE OpROM			Disabled	
	▶ PCI Subsystem Settings				
	▶ ACPI Settings				
	▶ Wake up event setting				
	▶ CPU Configuration				
	▶ iSmart Configuration				
	▶ IDE Configuration				
	▶ USB Configuration				
	▶ W83627UHG Super IO Configuration				
	▶ H/W Monitor				
	▶ PPM Configuration				
					→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit

### Launch PXE OpROM

Enable or Disable Boot Option for Legacy Network Devices.

**PCI Subsystem Settings**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
	PCI Bus Driver Version		V 2.05.01		
	PCI ROM Priority		Legacy ROM		
	PCI Common Settings				
	PCI Latency Timer		32 PCI Bus Clocks		
	VGA Palette Snoop		Disabled		
	PERR# Generation		Disabled		
	SERR# Generation		Disabled		
					→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit

**PCI ROM Priority**

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

**PCI Latency Timer**

Value to be programmed into PCI Latency Timer Register.

**VGA Palette Snoop**

Enables or Disables VGA Palette Registers Snooping.

**PERR# Generation**

Enables or Disables PCI Device to Generate PERR#.

**SERR# Generation**

Enables or Disables PCI Device to Generate SERR#.

## ACPI Settings

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
ACPI Settings					
Enable ACPI Auto Configuration			Disabled		
					→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit

### Enabled ACPI Auto Configuration

Enables or Disables BIOS ACPI Auto Configuration.

**Wake up event setting**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
			Wake on Ring	Disabled	→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit
			Wake on PCIE PME	Disabled	

**Wake on Ring**

The options are Disabled and Enabled.

**Wake on PCIE PME**

The options are Disabled and Enabled.

**CPU Configuration**

This section shows the CPU configuration parameters.

Aptio Setup Utility					
Main	Advanced	Chipset	Boot	Security	Save & Exit
CPU Configuration					
Processor Type		Intel(R) Atom(TM) CPU			
EMT64		Not Supported			
Processor Speed		1865 MHz			
System Bus Speed		533 MHz			
Ratio Status		14			
Actual Ratio		14			
System Bus Speed		533 MHz			
Processor Stepping		30661			
Microcode Revision		265			
L1 Cache RAM		2x56 k		→ ← Select Screen	
L2 Cache RAM		2x512 k		↑ ↓ Select Item	
Processor Core		Dual		Enter: Select	
Hyper-Threading		Supported		+- Change Opt	
Hyper-Threading		Enabled		F1: General Help	
Execute Disable Bit		Enabled		F2: Previous Values	
Limit CPUID Maximum		Disabled		F3: Optimized Default	
				F4: Save & EXIT	
				ESC: Exit	

**Hyper-threading**

Enabled for Windows XP and Linux (OS optimized for Hyper-Threading Technology) and Disabled for other OS (OS not optimized for Hyper-Threading Technology). When Disabled, only one thread per enabled core is enabled.

**Execute Disable Bit**

XD can prevent certain classes of malicious buffer overflow attacks when combined with a supporting OS (Windows Server 2003 SP1, Windows XP SP2, SuSE Linux 9.2, Re33dHat Enterprise 3 Update 3.)

**Limit CPUID Maximum**

Disabled for Windows XP.

**iSmart Controller**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
iSmart Controller					→ ← Select Screen
EuP/ErP standby power Control					↑ ↓ Select Item
Power-On after Power failure					Enter: Select
Schedule Slot 1					+ - Change Opt
Schedule Slot 2					F1: General Help
					F2: Previous Values
					F3: Optimized Default
					F4: Save & EXIT
					ESC: Exit

**EuP/ErP standby power Control**

Saving the power consumption on power off.

**Power-On after Power failure**

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

**Schedule Slot**

None / Power On / Power On/Off – Setup the hour/minute for system power on

**IDE Configuration**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
SATA Port0					→ ← Select Screen
SATA Port1					↑ ↓ Select Item
SATA Controller(s)					Enter: Select
Configure SATA as					+ - Change Opt
					F1: General Help
					F2: Previous Values
					F3: Optimized Default
					F4: Save & EXIT
					ESC: Exit

**SATA Controller(s)**

Enable / Disable Serial ATA Controller.

**Configure SATA as**

- (1) IDE Mode.
- (2) AHCI Mode.

**USB Configuration**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
USB Configuration					
USB Devices:					
None					→ ← Select Screen
Legacy USB Support					↑ ↓ Select Item
Enabled					Enter: Select
EHCI Hand-off					+ - Change Opt
Enabled					F1: General Help
USB hardware delays and time-outs:					F2: Previous Values
USB Transfer time-out					F3: Optimized Default
20 sec					F4: Save & EXIT
Device reset time-out					ESC: Exit
20 sec					
Device power-up delay					
Auto					

**Legacy USB Support**

Enables Legacy USB support.

AUTO option disables legacy support if no USB devices are connected.

DISABLE option will keep USB devices available only for EFI applications.

**EHCI Hand-off**

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

**USB Transfer time-out**

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

**Device reset time-out**

USB mass Storage device start Unit command time-out.

**Device power-up delay**

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



## Super IO Configuration

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
W83627UHG Super IO Configuration					→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit
Super IO Chip		Winbond W83627UHG			
▶ W83627UHG Serial Port 0 Configuration					
▶ W83627UHG Serial Port 1 Configuration					
▶ W83627UHG Serial Port 2 Configuration					
▶ W83627UHG Serial Port 3 Configuration					

## Serial Port Configuration

Set Parameters of Serial Ports. User can Enable/Disable the serial port and Select an optimal settings for the Super IO Device.

## H/W Monitor

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
Pc Health Status					→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit
System temperature		+50 C			
CPU temperature		+55 C			
CPU Fan Speed		N/A			
V CORE		+1.184 V			
+12V		+12.032 V			
+3.3V		+3.248 V			
+5V		+4.960 V			
CPU Shutdown Temperature		Disabled			
SmartFan Control		Disabled			

## 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.

## CPU Shutdown Temperature

The default setting is Disabled.

## Smart Fan Control

Disabled (default)

60 °C

70 °C

80 °C

90 °C

## PPM Configuration

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
PPM Configuration					
EIST			Enabled		→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit

## Chipset Settings

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

Aptio Setup Utility					
Main	Advanced	Chipset	Boot	Security	Save & Exit
<ul style="list-style-type: none"> <li>▶ Host Bridge</li> <li>▶ South Bridge</li> </ul>				→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Field F1: General Help F2: Previous Values F3: Optimized Default F4: Save ESC: Exit	

### Host Bridge

This item shows the Host Bridge Parameters.

### South Bridge

This item shows the South Bridge Parameters.

### Host Bridge

This section allows you to configure the Host Bridge Chipset.

Aptio Setup Utility					
Main	Advanced	Chipset	Boot	Security	Save & Exit
<ul style="list-style-type: none"> <li>▶ Memory Frequency and Timing</li> <li>▶ Intel IGD Configuration</li> </ul>				→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit	
*****Memory Information*****					
Memory Frequency			1067 MHz(DDR3)		
Total Memory			2048 MB		
DIMM#1			2048 MB		

**Memory Frequency and Timing**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
Memory Frequency and Timing					
MRC Fast Boot			Enabled		
Max TOLUD			Dynamic		
					→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit

**MRC Fast Boot**

The options are Disabled and Enabled.

**Max TOLUD**

The default setting is Dynamic.

**Intel IGD Configuration**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
Intel IGD Configuration					
IGFX-Boot Type			VBIOS Default		
					→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit

**IGFX-Boot Type**

Select the video Device which will be activated during POST .

## South Bridge

This section allows you to configure the South Bridge Chipset.

Aptio Setup Utility					
Main	Advanced	Chipset	Boot	Security	Save & Exit
<ul style="list-style-type: none"> <li>▶ TPT Device</li> <li>▶ PCI Express Root Port0</li> <li>▶ PCI Express Root Port1</li> <li>▶ PCI Express Root Port2</li> <li>▶ PCI Express Root Port3</li> </ul>					
DMI Link ASPM Control			Enabled	→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit	
PCI-Exp. High Priority Port			Disabled		
High Precision Event Timer Configuration					
High Precision Timer			Enabled		
SLP_SP4 Assertion Width			1-2 Seconds		

### DMI Clink ASPM Control

The control of Active State Power Management on both NB side and SB side of the DMI Link.

### PCI-Exp. High Priority Port

The options are Disabled, Port1, Port2, Port3, and Port4.

### High Precision Event Timer Configuration

Enable/or Disable the High Precision Event Timer.

### SLP\_S4 Assertion Stretch Enable

Select a minimum assertion width of the SLP\_S4# signal.

**TPT Device**

## Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
		Azalia Controller	HD Audio		→ ← Select Screen
		Select USB Mode	By Controllers		↑ ↓ Select Item
		UHCI #1 (port 0 and 1)	Enabled		Enter: Select
		UHCI #2 (port 2 and 3)	Enabled		+ - Change Opt
		UHCI #3 (port 4 and 5)	Enabled		F1: General Help
		UHCI #4 (port 6 and 7)	Enabled		F2: Previous Values
		USB 2.0(EHCI) Support	Enabled		F3: Optimized Default
					F4: Save & EXIT
					ESC: Exit

**PCI Express Root Port0**

## Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
		PCI Express Port 0	Enabled		→ ← Select Screen
		Port 0 IOxAPIC	Disabled		↑ ↓ Select Item
		Automatic ASPM	Manual		Enter: Select
		ASPM L0s	Root Port Only		+ - Change Opt
		ASPM L1	Enabled		F1: General Help
					F2: Previous Values
					F3: Optimized Default
					F4: Save & EXIT
					ESC: Exit

**PCI Express Root Port1**

## Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
		PCI Express Port 0	Auto		→ ← Select Screen
		Port 0 IOxAPIC	Disabled		↑ ↓ Select Item
		Automatic ASPM	Manual		Enter: Select
		ASPM L0s	Root Port Only		+ - Change Opt
		ASPM L1	Enabled		F1: General Help
					F2: Previous Values
					F3: Optimized Default
					F4: Save & EXIT
					ESC: Exit

**PCI Express Root Port2**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
			Auto		→ ← Select Screen
			Disabled		↑ ↓ Select Item
			Auto		Enter: Select
					+ - Change Opt
					F1: General Help
					F2: Previous Values
					F3: Optimized Default
					F4: Save & EXIT
					ESC: Exit

**PCI Express Root Port3**

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
			Auto		→ ← Select Screen
			Disabled		↑ ↓ Select Item
			Auto		Enter: Select
					+ - Change Opt
					F1: General Help
					F2: Previous Values
					F3: Optimized Default
					F4: Save & EXIT
					ESC: Exit

**Boot Settings****Aptio Setup Utility**

Main	Advanced	Chipset	Boot	Security	Save & Exit
Boot Configuration					
Setup Prompt Timeout			1		
Bootup NumLock State			On		
Quiet Boot			Disabled		→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit
Fast Boot			Disabled		
CSM16 Module Version			07.68		
GateA20 Active			Upon Request		
Option ROM Messages			Force BIOS		
Interrupt 19 Capture			Enabled		
CSM Support			Enabled		
Boot Option Priorities					

**Setup Prompt Timeout**

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

**Bootup NumLock State**

Select the keyboard NumLock state.

**Quiet Boot**

Enables/Disables Quiet Boot option.

**Fast Boot**

Enables/Disables boot with initialization of a minimal set of devices required to launch active boot option. Has no effect for BBS boot options.

**GateA20 Active**

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

**Option ROM Messages**

Set display mode for Option ROM. Options: Force BIOS; Keep Current.

**Interrupt 19 Capture**

Enable: Allows Option ROMs to trap Int 19.

**CSM Support**

Enables/Disables/Auto CSM Support.



## Security Settings

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

Aptio Setup Utility					
Main	Advanced	Chipset	Boot	Security	Save & Exit
Password Description  If ONLY the Administrator's password is set, then this only limits access to Setup and is only asked for when entering Setup. If ONLY the User's password is set, then this is a power on password and must be entered to boot or enter Setup. In Setup the User will have Administrator rights  Administrator Password User Password					→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit

### Administrator Password

Set Setup Administrator Password.

### User Password

Set User Password.

## Save & Exit Settings

Aptio Setup Utility

Main	Advanced	Chipset	Boot	Security	Save & Exit
Save Changes and Exit				→ ← Select Screen ↑ ↓ Select Item Enter: Select +- Change Opt F1: General Help F2: Previous Values F3: Optimized Default F4: Save & EXIT ESC: Exit	
Discard Changes and Exit					
Save Changes and Reset					
Discard Changes and Reset					
Save Options					
Save Changes					
Discard Changes					
Restore Defaults					
Save as User Defaults					
Restore User Defaults					
Boot Override					

### Save Changes and Exit

Exit system setup after saving the changes.

### Discard Changes and Exit

Exit system setup without saving any changes.

### Save Changes and Reset

Reset the system after saving the changes.

### Discard Changes and Reset

Reset system setup without saving any changes.

### Save Changes

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

### Discard Changes

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

### Restore Defaults

Restore/Load Defaults values for all the setup options.

### Save as User Defaults

Save the changes done so far as User Defaults.

### Restore User Defaults

Restore the User Defaults to all the setup options.

## Drivers Installation

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

Intel Chipset Software Installation Utility .....	40
VGA Drivers Installation .....	41
Realtek HD Audio Driver Installation .....	42
LAN Drivers Installation.....	43

**IMPORTANT NOTE:**

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

## **Intel Chipset Software Installation Utility**

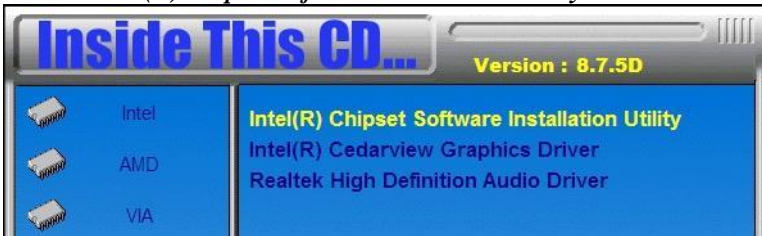
---

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.

1. Insert the disc that comes with the board. Click **Intel** and then **Intel(R) Cedarview Chipset Drivers**.



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



3. When the Welcome screen to the Intel® Chipset Device Software appears, click **Next** to continue.

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

5. On the Readme File 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

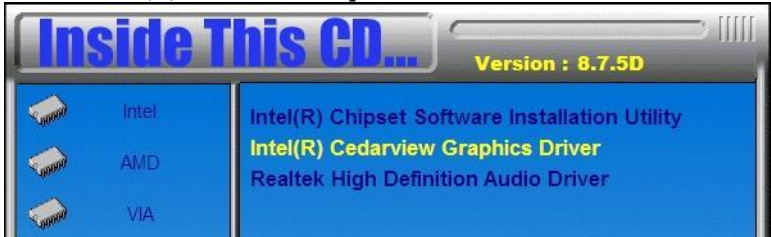
---

**IMPORTANT NOTE:** After installing the graphics driver, the default display is still LVDS. Both VGA/CRT and DVI will only be extended displays. To switch to any of these displays, use the hot key:

For VGA – **Ctrl+Alt+F1**

For DVI – **Ctrl+Alt+F3**

1. Click **Intel(R) Cedarview Graphics Driver**.



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



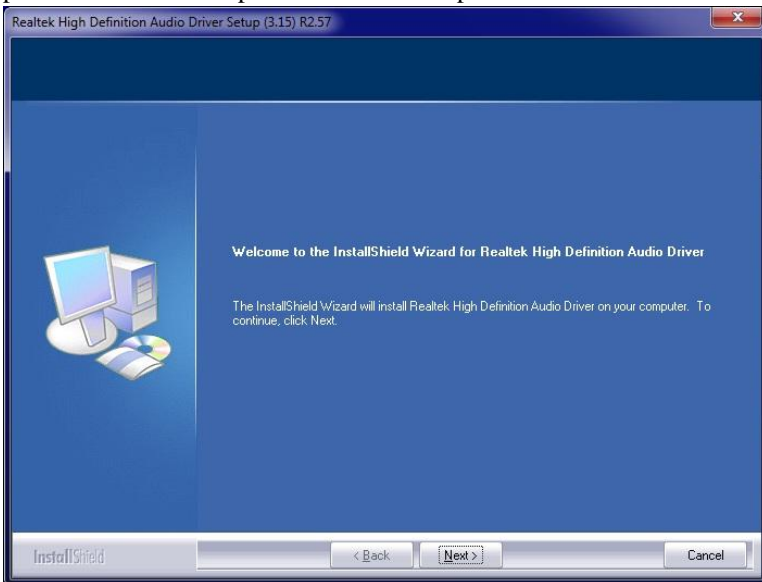
4. Click **Yes** to to agree with the license agreement and continue the installation.
5. On the Readme File Information screen, click **Next** to continue the installation of the Intel® Graphics Media Accelerator Driver.
6. On Setup Progress screen, click **Next** to continue.
7. Setup complete. Click **Finish** to restart the computer and for changes to take effect.

## Realtek HD Audio Driver Installation

1. Click *Realtek High Definition Audio Driver*.



2. On the Welcome to the InstallShield Wizard screen, click *Next* to proceed with and complete the installation process.



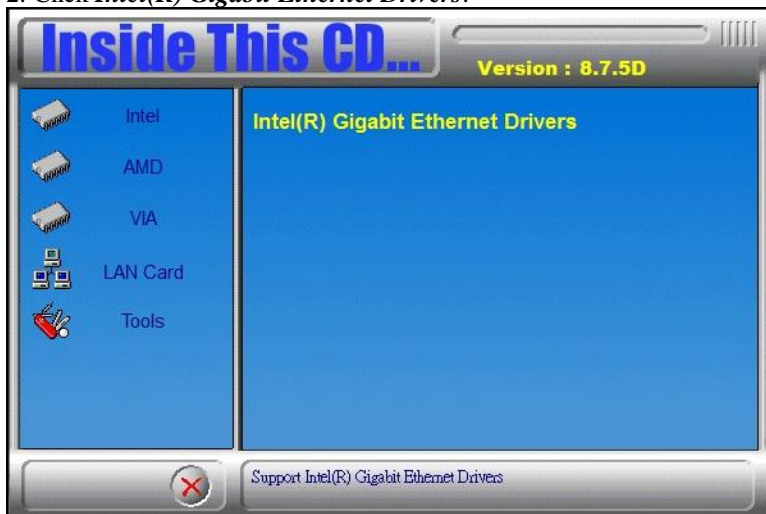
3. Restart the computer when prompted.

## LAN Drivers Installation

---

1. Insert the CD that comes with the board. Click *LAN Card* and then *Intel(R) LAN Controller Drivers*.

2. Click *Intel(R) Gigabit Ethernet Drivers*.



3. In the Welcome screen, click *Next*.

4. In the License Agreement screen, click *I accept the terms in license agreement* and *Next* to accept the software license agreement and proceed with the installation process.

5. Click the checkbox for **Drivers** in the Setup Options screen to select it and click **Next** to continue.

6. When the Ready to Install the Program screen appears, click *Install* to continue.

7. When InstallShield Wizard is complete, click *Finish*.

# Appendix

## A. I/O Port Address Map

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

<b>Address</b>	<b>Device Description</b>
0000-001F	Direct memory access controller
0000-001F	PCI bus
0020-0021	Programmable interrupt controller
0024-0025	Programmable interrupt controller
0028-0029	Programmable interrupt controller
002C-002D	Programmable interrupt controller
0030-0031	Programmable interrupt controller
0034-0035	Programmable interrupt controller
0038-0039	Programmable interrupt controller
003C-003D	Programmable interrupt controller
0040-0043	System timer
0050-0053	System timer
0060-0060	Standard PS/2 Keyboard
0064-0064	Standard PS/2 Keyboard
0070-0077	System CMOS/real time clock
0081-0091	Direct memory access controller
0093-009F	Direct memory access controller
00A0-00A1	Programmable interrupt controller
00A4-00A5	Programmable interrupt controller
00A8-00A9	Programmable interrupt controller
00AC-00AD	Programmable interrupt controller
00B0-00B1	Programmable interrupt controller
00B4-00B5	Programmable interrupt controller
00B8-00B9	Programmable interrupt controller
00BC-00BD	Programmable interrupt controller
00C0-00DF	Direct memory access controller
00F0-00F0	Numeric data processor
02E8-02EF	Communications Port (COM4)
02F8-02FF	Communications Port (COM2)



<b>Address</b>	<b>Device Description</b>
03B0-03BB	Intel(R) Graphics Media Accelerator 3600 Series
03C0-03DF	Intel(R) Graphics Media Accelerator 3600 Series
03E8-03EF	Communications Port (COM3)
03F8-03FF	Communications Port (COM1)
04D0-04D1	Programmable interrupt controller
0D00-FFFF	PCI bus
D000-DFFF	Intel(R) N10/ICH7 Family PCI Express Root Port - 27D2
E000-EFFF	Intel(R) N10/ICH7 Family PCI Express Root Port - 27D0
F000-F01F	Intel(R) N10/ICH7 Family SMBus Controller - 27DA
F020-F03F	Intel(R) N10/ICH7 Family USB Universal Host Controller - 27CB
F040-F05F	Intel(R) N10/ICH7 Family USB Universal Host Controller - 27CA
F060-F07F	Intel(R) N10/ICH7 Family USB Universal Host Controller - 27C9
F080-F09F	Intel(R) N10/ICH7 Family USB Universal Host Controller - 27C8
F0A0-F0AF	Intel(R) N10/ICH7 Family Serial ATA Storage Controller - 27C0
F0B0-F0B3	Intel(R) N10/ICH7 Family Serial ATA Storage Controller - 27C0
F0C0-F0C7	Intel(R) N10/ICH7 Family Serial ATA Storage Controller - 27C0
F0D0-F0D3	Intel(R) N10/ICH7 Family Serial ATA Storage Controller - 27C0
F0E0-F0E7	Intel(R) N10/ICH7 Family Serial ATA Storage Controller - 27C0
F0F0-F0F7	Intel(R) Graphics Media Accelerator 3600 Series

## 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.

<b>Level</b>	<b>Function</b>
IRQ 0	System timer
IRQ 1	Standard PS/2 Keyboard
IRQ 3	Communications Port (COM2)
IRQ 4	Communications Port (COM1)
IRQ 5	Communications Port (COM3)
IRQ 7	Communications Port (COM4)
IRQ 8	System CMOS/real time clock
IRQ 11	Intel(R) N10/ICH7 Family SMBus Controller - 27DA
IRQ 12	Microsoft PS/2 Mouse
IRQ 13	Numeric data processor
IRQ 16	Intel(R) N10/ICH7 Family USB Universal Host Controller - 27CB
IRQ 18	Intel(R) N10/ICH7 Family USB Universal Host Controller - 27CA
IRQ 19	Intel(R) N10/ICH7 Family Serial ATA Storage Controller - 27C0
IRQ 19	Intel(R) N10/ICH7 Family USB Universal Host Controller - 27C9
IRQ 22	High Definition Audio Controller
IRQ 23	Intel(R) N10/ICH7 Family USB2 Enhanced Host Controller - 27CC
IRQ 23	Intel(R) N10/ICH7 Family USB Universal Host Controller - 27C8
IRQ 4294967290	Intel(R) 82583V Gigabit Network Connection
IRQ 4294967291	Intel(R) 82583V Gigabit Network Connection #2
IRQ 4294967292	Intel(R) Graphics Media Accelerator 3600 Series
IRQ 4294967293	Intel(R) N10/ICH7 Family PCI Express Root Port - 27D2
IRQ 4294967294	Intel(R) N10/ICH7 Family PCI Express Root Port - 27D0

## C. Digital I/O Sample Code

File of the W627UHG.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 __W627UHG_H  
#define __W627UHG_H          1  
//-----  
#define W627UHG_INDEX_PORT (W627UHG_BASE)  
#define W627UHG_DATA_PORT (W627UHG_BASE+1)  
//-----  
#define W627UHG_REG_LD      0x07  
//-----  
#define W627UHG_UNLOCK     0x87  
#define W627UHG_LOCK       0xAA  
//-----  
unsigned int Init_W627UHG(void);  
void Set_W627UHG_LD( unsigned char);  
void Set_W627UHG_Reg( unsigned char, unsigned char);  
unsigned char Get_W627UHG_Reg( unsigned char);  
//-----  
#endif    //__W627UHG_H
```

File of the W627UHG.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 "W627UHG.H"  
#include <dos.h>  
//-----  
unsigned int W627UHG_BASE;  
void Unlock_W627UHG (void);  
void Lock_W627UHG (void);  
//-----  
unsigned int Init_W627UHG(void)  
{  
    unsigned int result;  
    unsigned char ucDid;  
  
    W627UHG_BASE = 0x4E;  
    result = W627UHG_BASE;  
  
    ucDid = Get_W627UHG_Reg(0x20);  
    if (ucDid == 0xA2)                                //W83627UHG??  
    {        goto Init_Finish;    }  
  
    W627UHG_BASE = 0x2E;  
    result = W627UHG_BASE;  
  
    ucDid = Get_W627UHG_Reg(0x20);  
    if (ucDid == 0xA2)                                //W83627UHG??  
    {        goto Init_Finish;    }  
  
    W627UHG_BASE = 0x00;  
    result = W627UHG_BASE;  
  
Init_Finish:  
    return (result);  
}  
//-----  
void Unlock_W627UHG (void)  
{  
    outportb(W627UHG_INDEX_PORT, W627UHG_UNLOCK);  
    outportb(W627UHG_INDEX_PORT, W627UHG_UNLOCK);  
}  
//-----  
void Lock_W627UHG (void)  
{  
    outportb(W627UHG_INDEX_PORT, W627UHG_LOCK);  
}  
//-----  
void Set_W627UHG_LD( unsigned char LD)
```

```
{
    Unlock_W627UHG();
    outportb(W627UHG_INDEX_PORT, W627UHG_REG_LD);
    outportb(W627UHG_DATA_PORT, LD);
    Lock_W627UHG();
}
//-----
void Set_W627UHG_Reg( unsigned char REG, unsigned char DATA)
{
    Unlock_W627UHG();
    outportb(W627UHG_INDEX_PORT, REG);
    outportb(W627UHG_DATA_PORT, DATA);
    Lock_W627UHG();
}
//-----
unsigned char Get_W627UHG_Reg(unsigned char REG)
{
    unsigned char Result;
    Unlock_W627UHG();
    outportb(W627UHG_INDEX_PORT, REG);
    Result = inportb(W627UHG_DATA_PORT);
    Lock_W627UHG();
    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 "W627UHG.H"  
//-----  
int main (void);  
  
void Dio5Initial(void);  
void Dio5SetOutput(unsigned char);  
unsigned char Dio5GetInput(void);  
void Dio5SetDirection(unsigned char);  
unsigned char Dio5GetDirection(void);  
//-----  
int main (void)  
{  
    char SIO;  
  
    SIO = Init_W627UHG();  
    if (SIO == 0)  
    {  
        printf("Can not detect Winbond 83627UHG, program abort.\n");  
        return(1);  
    }  
  
    Dio5Initial();  
  
    //for GPIO50..57  
    Dio5SetDirection(0x0F); //GP50..53 = input, GP54..57=output  
    printf("Current DIO direction = 0x%X\n", Dio5GetDirection());  
  
    printf("Current DIO status = 0x%X\n", Dio5GetInput());  
  
    printf("Set DIO output to high\n");  
    Dio5SetOutput(0x0F);  
  
    printf("Set DIO output to low\n");  
    Dio5SetOutput(0x00);  
  
    return 0;
```

```
}
//-----
void Dio5Initial(void)
{
    unsigned char ucBuf;

    Set_W627UHG_LD(0x08); //switch to logic device 8
    //enable the GP5 group
    ucBuf = Get_W627UHG_Reg(0x30);
    ucBuf |= 0x02;
    Set_W627UHG_Reg(0x30, ucBuf);
}
//-----
void Dio5SetOutput(unsigned char NewData)
{
    Set_W627UHG_LD(0x08); //switch to logic device 8
    Set_W627UHG_Reg(0xE1, NewData);
}
//-----
unsigned char Dio5GetInput(void)
{
    unsigned char result;

    Set_W627UHG_LD(0x08); //switch to logic device 8
    result = Get_W627UHG_Reg(0xE1);
    return (result);
}
//-----
void Dio5SetDirection(unsigned char NewData)
{
    //NewData : 1 for input, 0 for output
    Set_W627UHG_LD(0x08); //switch to logic device 8
    Set_W627UHG_Reg(0xE0, NewData);
}
//-----
unsigned char Dio5GetDirection(void)
{
    unsigned char result;

    Set_W627UHG_LD(0x08); //switch to logic device 8
    result = Get_W627UHG_Reg(0xE0);
    return (result);
}
//-----
```

## D. 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:

File of the W627UHG.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 "W627UHG.H"  
#include <dos.h>  
//-----  
unsigned int W627UHG_BASE;  
void Unlock_W627UHG (void);  
void Lock_W627UHG (void);  
//-----  
unsigned int Init_W627UHG(void)  
{  
    unsigned int result;  
    unsigned char ucDid;  
  
    W627UHG_BASE = 0x4E;  
    result = W627UHG_BASE;  
  
    ucDid = Get_W627UHG_Reg(0x20);  
    if (ucDid == 0xA2) //W83627UHG??  
    { goto Init_Finish; }  
  
    W627UHG_BASE = 0x2E;  
    result = W627UHG_BASE;  
  
    ucDid = Get_W627UHG_Reg(0x20);  
    if (ucDid == 0xA2) //W83627UHG??  
    { goto Init_Finish; }  
  
    W627UHG_BASE = 0x00;  
    result = W627UHG_BASE;
```



```
Init_Finish:
    return (result);
}
//-----
void Unlock_W627UHG (void)
{
    outportb(W627UHG_INDEX_PORT, W627UHG_UNLOCK);
    outportb(W627UHG_INDEX_PORT, W627UHG_UNLOCK);
}
//-----
void Lock_W627UHG (void)
{
    outportb(W627UHG_INDEX_PORT, W627UHG_LOCK);
}
//-----
void Set_W627UHG_LD( unsigned char LD)
{
    Unlock_W627UHG();
    outportb(W627UHG_INDEX_PORT, W627UHG_REG_LD);
    outportb(W627UHG_DATA_PORT, LD);
    Lock_W627UHG();
}
//-----
void Set_W627UHG_Reg( unsigned char REG, unsigned char DATA)
{
    Unlock_W627UHG();
    outportb(W627UHG_INDEX_PORT, REG);
    outportb(W627UHG_DATA_PORT, DATA);
    Lock_W627UHG();
}
//-----
unsigned char Get_W627UHG_Reg(unsigned char REG)
{
    unsigned char Result;
    Unlock_W627UHG();
    outportb(W627UHG_INDEX_PORT, REG);
    Result = inportb(W627UHG_DATA_PORT);
    Lock_W627UHG();
    return Result;
}
//-----
```

## APPENDIX

---

File of the W627UHG.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 __W627UHG_H  
#define __W627UHG_H            1  
//-----  
#define W627UHG_INDEX_PORT    (W627UHG_BASE)  
#define W627UHG_DATA_PORT     (W627UHG_BASE+1)  
//-----  
#define W627UHG_REG_LD        0x07  
//-----  
#define W627UHG_UNLOCK        0x87  
#define W627UHG_LOCK          0xAA  
//-----  
unsigned int Init_W627UHG(void);  
void Set_W627UHG_LD( unsigned char);  
void Set_W627UHG_Reg( unsigned char, unsigned char);  
unsigned char Get_W627UHG_Reg( unsigned char);  
//-----  
#endif    // __W627UHG_H
```

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 "W627UHG.H"  
//-----  
int main (void);  
  
void WDTInitial(void);  
void WDTEnable(unsigned char);  
void WDTDisable(void);  
  
//-----  
int main (void)  
{  
    char SIO;  
  
    SIO = Init_W627UHG();  
    if (SIO == 0)  
    {  
        .....printf("Can not detect Winbond 83627UHG, program abort.\n");  
        ..... return(1);  
    }  
  
    WDTInitial();  
  
    WDTEnable(10);  
  
    WDTDisable();  
  
    return 0;  
}  
//-----  
void WDTInitial(void)  
{  
    unsigned char bBuf;  
    Set_W627UHG_LD(0x08);.....//switch to logic device 8  
    bBuf = Get_W627UHG_Reg(0x30);  
    bBuf &= (~0x01);  
    Set_W627UHG_Reg(0x30, bBuf);.....//Enable WDIO  
}  
//-----  
void WDTEnable(unsigned char NewInterval)
```

```
{
    unsigned char bBuf;

    Set_W627UHG_LD(0x08);..... //switch to logic device 8
    Set_W627UHG_Reg(0x30, 0x01);..... //enable timer
    bBuf = Get_W627UHG_Reg(0xF5);
    bBuf &= (~0x08);
    Set_W627UHG_Reg(0xF5, bBuf);..... //count mode is second

    Set_W627UHG_Reg(0xF6, NewInterval); ..... //set timer
}
//-----
void WDTCDisable(void)
{
    Set_W627UHG_LD(0x08);.....
    Set_W627UHG_Reg(0xF6, 0x00);..... //clear watchdog timer
    Set_W627UHG_Reg(0x30, 0x00);..... //watchdog disabled
}
//-----
```