IDOOH-210-IR

User Manual

Revision	Release Date
VB1	2015/07/01

Table of Contents

Safety Information

Your IDOOH-210-IRis designed and tested to meet the latest standards of safety for information technology equipment. However, to ensure your safety, it is important that you read the following safety instructions

Setting up your system

- Read and follow all instructions in the documentation before you operate your system.
- Do not use this product near water.
- Set up the system on a stable surface. Do not secure the system on any unstable plane.
- Do not place this product on an unstable cart, stand, or table. The product may fall, causing serious damage to the product.
- Slots and openings on the chassis are for ventilation. Do not block or cover these openings. Make sure you leave plenty of space around the system for ventilation. ***Never insert objects of any kind into the ventilation openings***.
- This system should be operated from the type of power indicated on the marking label. If you are not sure of the type of power available, consult your dealer or local power company.
- Use this product in environments with ambient temperatures between -40˚C and 50˚C.*
- If you use an extension cord, make sure that the total ampere rating of the devices plugged into the extension cord does not exceed its ampere rating.
- DO NOT LEAVE THIS EQUIPMENT IN AN ENVIRONMENT WHERE THE STORAGE TEMPERATURE MAY GO BELOW -40° C OR ABOVE 60° C. THIS COULD DAMAGE THE EQUIPMENT. THE EQUIPMENT SHOULD BE IN A CONTROLLED ENVIRONMENT.

  *With 1120W/m² solar energy: 44 °C or with 900W/m² solar energy: 50 °C

Care during use

- Do not walk on the power cord or allow anything to rest on it.
- Do not spill water or any other liquids on your system.
- When the system is turned off, a small amount of electrical current still flows. Always unplug all power, and network cables from the power outlets before cleaning the system.
- If you encounter the following technical problems with the product, unplug the power cord and contact a qualified service technician or your retailer.
  - The power cord or plug is damaged.
  - Liquid has been spilled into the system.
  - The system does not function properly even if you follow the operating instructions.
  - The system was dropped or the cabinet is damaged.

Lithium-Ion Battery Warning

CAUTION: Danger of explosion if battery is incorrectly replaced. Replace only with the same or equivalent type recommended by the manufacturer. Dispose of used batteries according to the manufacturer's instructions.

NO DISASSEMBLY

The warranty does not apply to the products that have been disassembled by users.

WARNING
HAZARDOUS MOVING PARTS
KEEP FINGERS AND OTHER BODY PARTS AWAY

Acknowledgments

- AMI is a registered trademark of AMI Software International, Inc.
- Intel is registered trademarks of Intel Corporation.
- Microsoft Windows is a registered trademark of Microsoft Corporation.
- FINTEK is a registered trademark of FINTEK Electronics Corporation.
- REALTEK is a registered trademark of REALTEK Electronics Corporation.
- All other product names or trademarks are properties of their respective owners.

CHAPTER 1 INTRODUCTION

1.1 General Description

IDOOH-210-IR, an 21.5” ALL in ONE Panel PC utilizes Intel® E3800 series processor to provide high computing performance and low power consumption

The IDOOH-210-IR operates silently and reliably in harsh environments. It comes with two SODIMM slots to accommodate up to 8GB of DDR3L 1333MHz system memory and one 2.5” SATA SSD for data storage. It is provided with two Gigabit Ethernet, one RS-232/422/485 port. The unit is equipped with a front bezel with IR touch screen that has IP65-rated protection.

The IDOOH-210-IR supports a wide range 90~240VAC power and automatically temperature controlling system which makes it ideal for KIOSK, digital signage or any other industrial applications worldwide.

IDOOH-210-IR overview

1.2 System Specification

1.2.1 Hardware Specifications

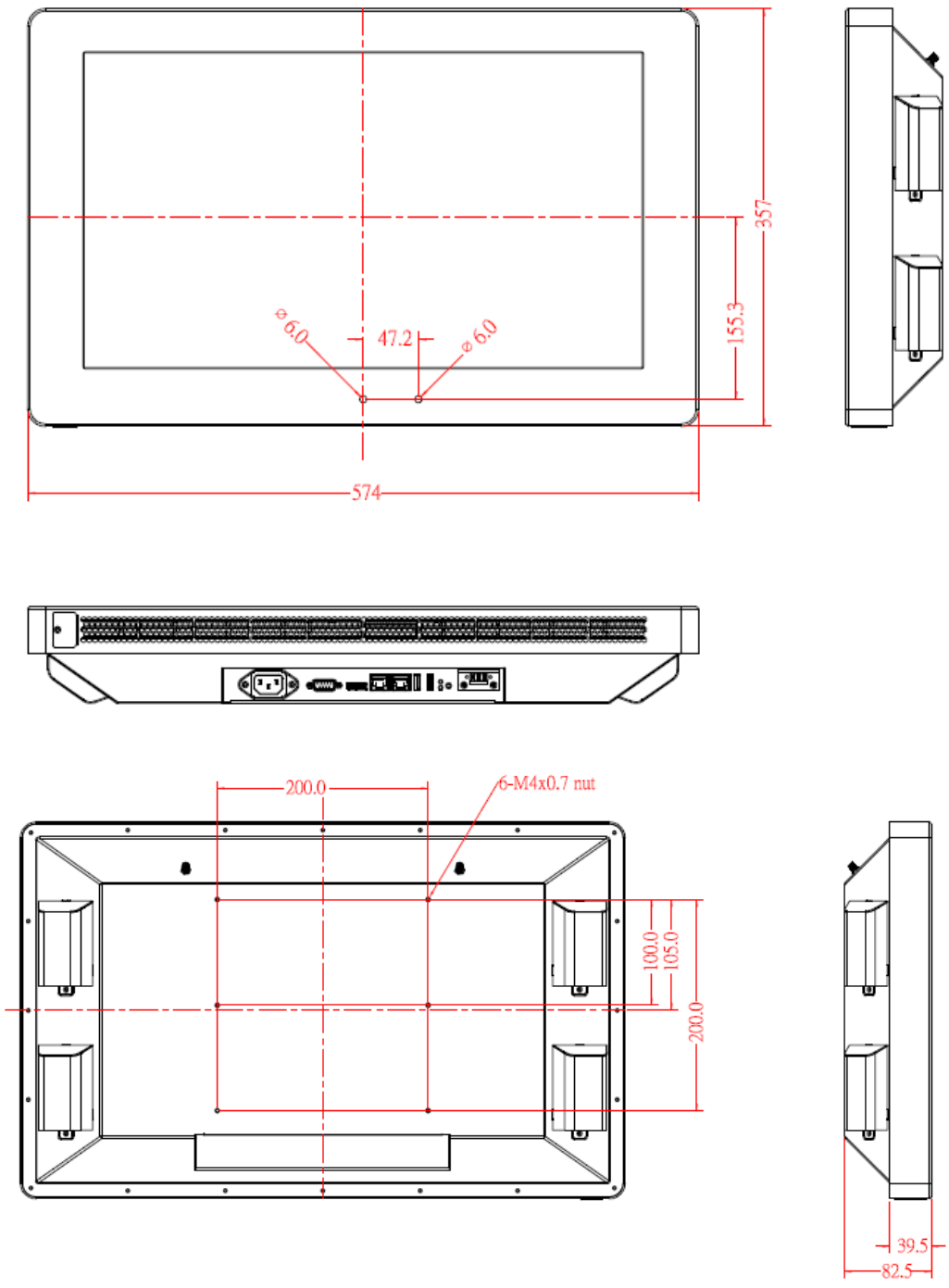
Model Name	IDOOH-210-IR
CPU	Intel® Atom™ QC E3845 (1.91GHz, 2MB cache)
Chipset	Integrated in Intel® Atom™ SoC
Memory	2x DDR3L-1333 SO-DIMM, up to 8GB, Default 4GB(4GBx1)
I/O Interface	1 x USB 3.0 flag type Blue color 1 x USB 2.0 Type A flag type 1 x D-SUB9 RS-232/422/485 COM1 1 x DP port 2 x Gigabit LAN (RJ45) 1 x DC power connector 1 x Power on/off switch, power on LED / HDD LED
Storage	1 x 2.5” SATA HDD easy accessible, default 64G SSD
Expansion Slots	None
Power Supply	90~240VAC
LCD Size	21.5”
LCD Color	16.7M
LCD Resolution	1920x1080
LCD Brightness	Up to 1200nits (auto-dimming)
LCD View Angle (H°/V°)	140/120
Touch Screen	IR touch
Construction	Aluminum front bezel and back cover
Mounting	VESA 200x100/200x200 mm
Dimensions (W)x(D)x(H) mm	574x357x82.5
Operating Temperature	-40°C**~ 50°C(With SSD)
Storage Temperature	-20°C ~ 60°C
Relative Humidity	10%~90% (non-condensing)
Protection Class	IP65 front bezel
Operating System Support	Windows 8 32/64bit, Windows 7 Pro for Embedded 64bit, WES7 64bit

***Based on internal temperature controlling system.*

·This specification is subject to change without prior notice

1.2.2 Dimensions

IDOOH-210-IR



1.2.3 I/O View

IDOOH-210-IR I/O side

1.3 Accessory List

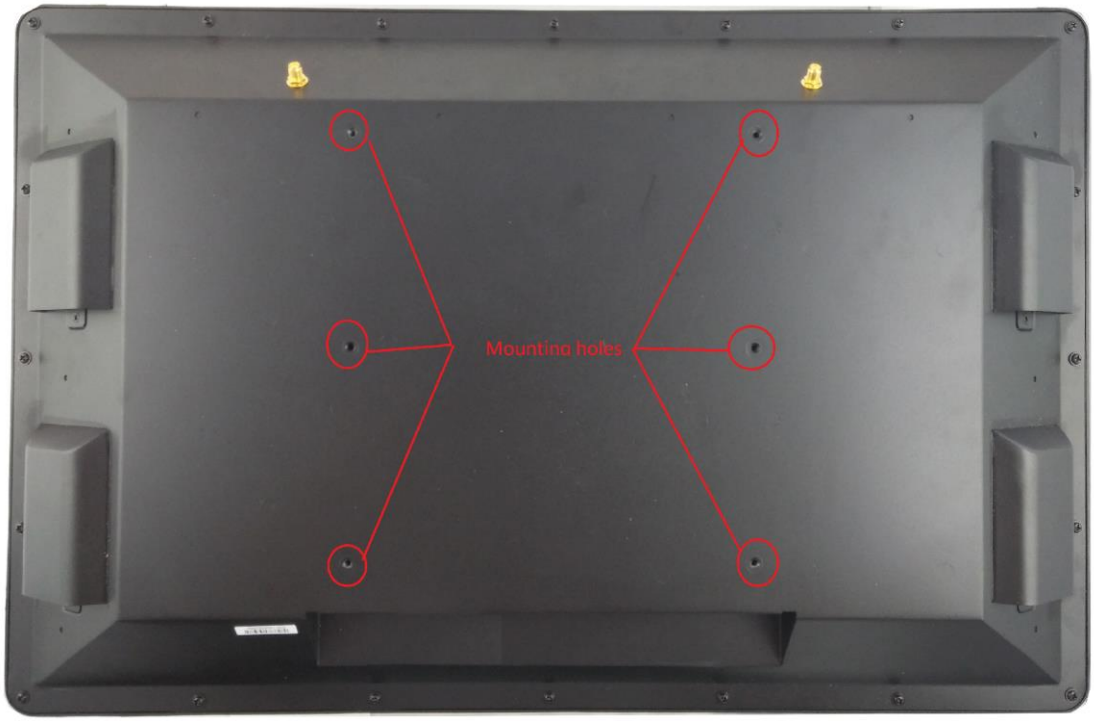
Part No.	Description	Quantity
1	Wi-Fi/BT Antenna	2 pc
2	DVD	1 pc
3	Motherboard manual	1 pc
4	Power Cord	1 pc

1.4 Installation

VESA mounting

IDOOH-210-IR supports standard VESA 200x200 or 200x100 mm VESA mounting.

Please use 4 x M4 screw to fix your mounting accessories.



Power on

Please just connect your power cord to power inlet.

IDOOH-210-IR supports auto- power on, you can change the setting under

BIOS menu:Advance-iSmart controller

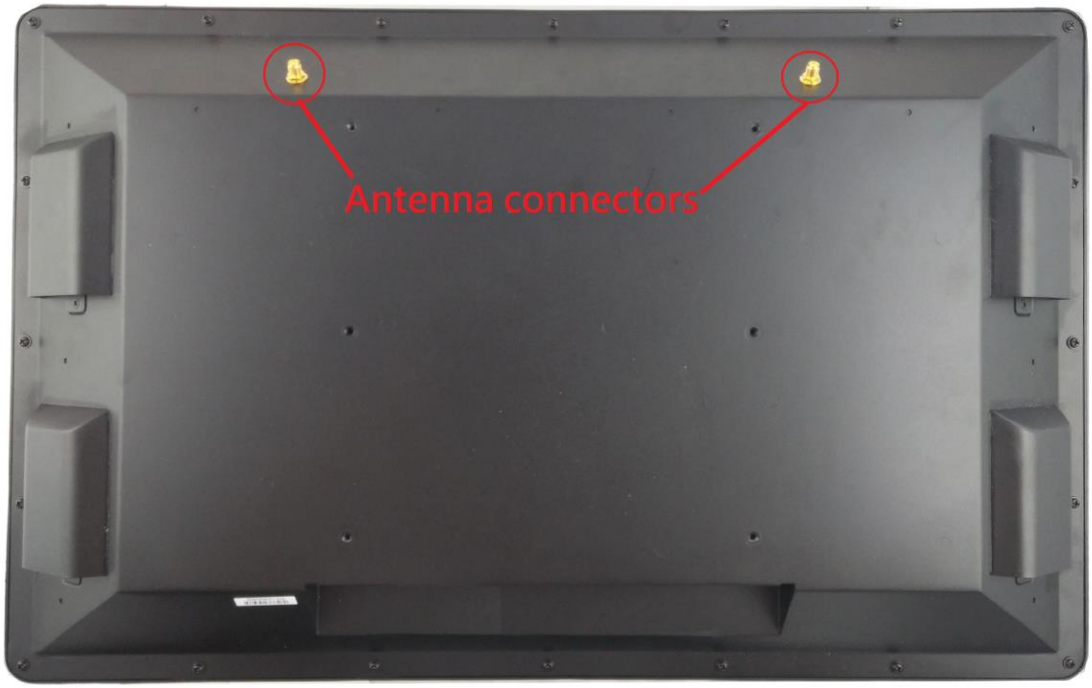
Power-On after Power failure

If your setting is Enable, the system will auto-turn on when power linked.

If your setting is disable, please push the power button hole to turn on/off the system

Additional Antenna

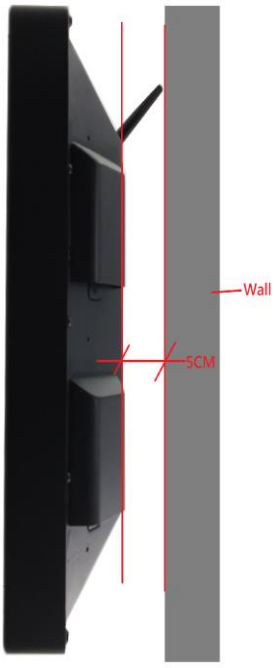
IDOOH-210-IR comes with 2 IP65 wireless antennas, just fix to the connector of the back-top side of the system.



Please note:

Due to the attached anteena is waterproof type which cannot adjust position, you have to keep 5cm space from backcover to backwall.

Or, you can order adjustable anteena from the market or our iBASE reference part No. A055RFA02C2M20800P



CHAPTER 2 MOTHERBOARD INTRODUCTION

Warning: IDOOH-210-IR is a finished end product, we don't suggest you change any setting of motherboard by yourself. That might cause any unexpected problem. If need, you can contact our FAE to support in advance.

2.1 Introduction

IB897-I45P is a 3.5-inch single board computer based on the Intel® AtomTM E3845 series processors which be used in IDOOH-210-IR. It supports two DDR3L (1.35V) SODIMM sockets for a maximum memory capacity of 8GB.

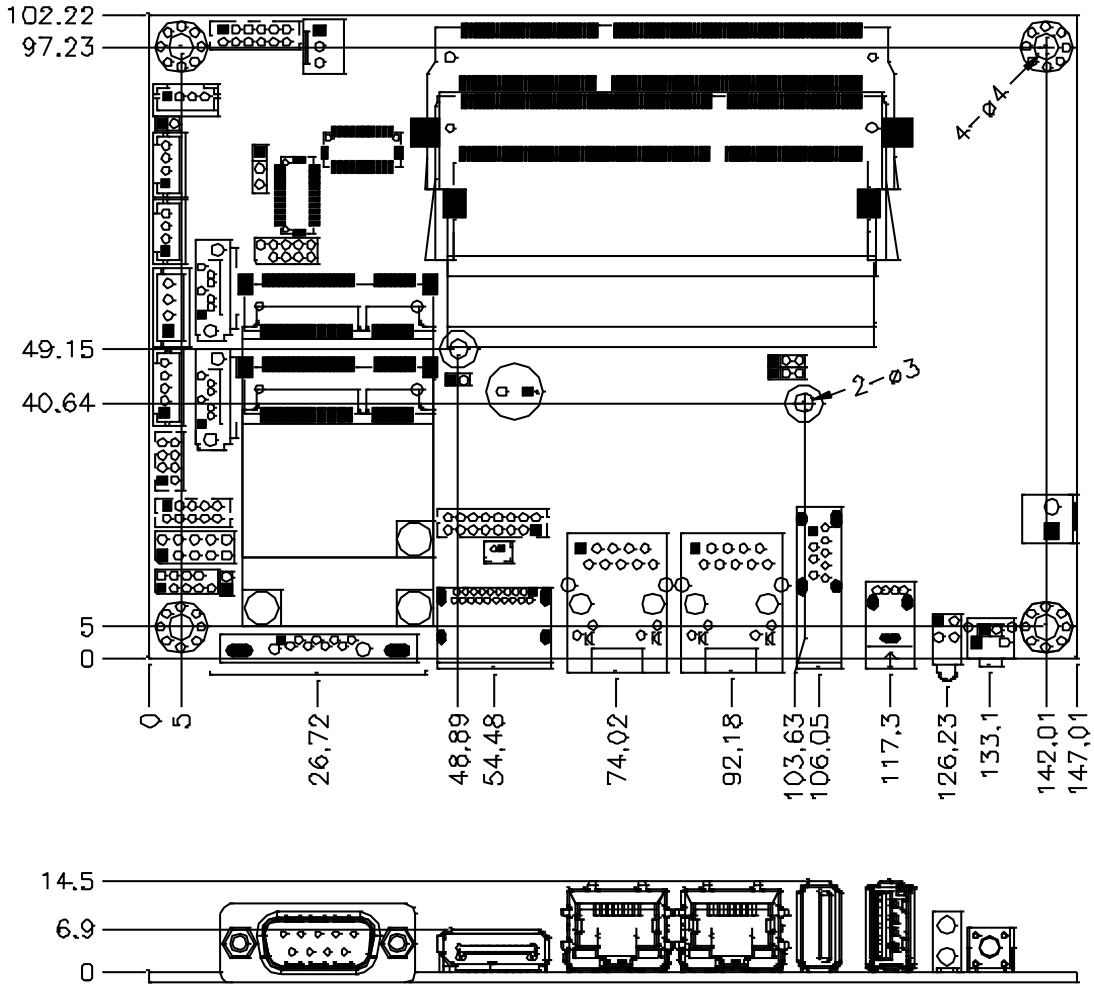
IB897 features the Intel® Gen7 w/4EUs graphics engines and has both CRT and DisplayPort video display interface, and 24-bit LVDS dual channel interface with the use of the NXP PTN3460 device.

Onboard connections are available for two SATAII ports, two COM ports, one USB 3.0 ports, four USB2.0 ports, audio, two Mini PCI-e(x1) slots, and Micro SD. Power input is made with a +9~+30V DC in interface.

Product Name	**IB897-I45P**
Form Factor	3.5" disk size SBC
SoC Type/Speed	Intel® Atom™ QC E3845 /2MB cache/1.91GHz Package = FCBGA1170, 25mmx27mm, 22nm,Tj= -40 ° C to +110 ° C
BIOS	AMI BIOS
Memory	Intel® Atom™ SoC integrated memory controller Support DDR3L (**1.35V only**), Non-ECC memory only 2 x DDR3 SO-DIMM socket [IB897-I45_P/IB897-I27_P], 8GB max. 1 x DDR3 SO-DIMM socket [IB897-I15_P], 4GB max.
VGA	Intel® Gen7 w/4 EUs graphics engines DisplayPort x 1 [Support to 2560x1200@60Hz] CRT x 1 via pin header [Support to 1920x1080@60Hz]
LVDS	24-bit dual channel via NXP PTN3460 thru eDP (Support to1920x1200 @ 60 Hz)
LAN	Intel® I210**I**T PCIe Gigabit LAN x 2
USB	Intel® Atom™ SoC built-in USB host controller Support USB 2.0 x 4 ports; USB 3.0 x 1 port,

	extra USB 2.0 x4 ports (Thru SMSC HUB USB2514)
Serial ATA	Intel® Atom™ SoC built-in SATA II controller, supports 2 ports
Audio	Intel® Atom™ SoC 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
LPC I/O	Nuvoton NCT5523D [64-pin LQFP, 7x7x1.4mm] - COM #1 (RS232/422/485) [EXAR SP339EER1 x 1 for jumper-less] - COM #2 (RS-232 only) [Hardware Monitor]: 2x Thermal inputs; 2x Voltage monitoring
Digital IO	4 in & 4 out
Expansion Slots	Mini PCI-e socket x**2** (1xFull-sized+1xHalf-sized,) **Full length MiniPCIe (1x) support mSATA**
Edge Connector	DB9 for COM1, DisplayPort, RJ45 x 2 for LAN 1 & 2 USB 2.0 vertical connector x 1, USB 3.0 vertical connector x 1 LED indicators (red+green) x1 for power and HDD status & power button x 1(IB897-I45/I27/I15) 4-pin header for LED indicator & 2-pin header for power button via cable
Onboard Header/ Connector	2x8 pin header for CRT; 2x4 pin header for 2x USB 2.0 DF20 socket connector x2 for 24-bit dual channel LVDS 4-pin box header for backlight/brightness control (PWM) 2x6 pin box header for Audio, 4-pin header for speaker 2x5 pin box header for COM2 2x5 pin headers for LPC (80-port card debugging purpose) Mini PCI-e(1x) connector x**2**, 5 pins box header for smart battery SATA connector x **2** for SATA device 4-pins power connector (JST type, for SATA device) 2-pins connector for power input, Micro SD slot (type 3.3V)
Watchdog	Yes (256 segments, 0, 1, 2…255 sec/min)
Power Connector	9V ~ 30V DC-in thru onboard 2-pin connector
Others	iSMART 2.0 [Auto-scheduler / Power resume]
OS Support	Windows 8.1 / Embedded; Windows 7 / Embedded, Linux
RoHS / REACH/ CE / FCC	Yes / Yes / Yes / Class B
Operating Temp.	-40 ˚ C to +85 ˚ C
Board Size	102mm x 147mm

Board Dimensions



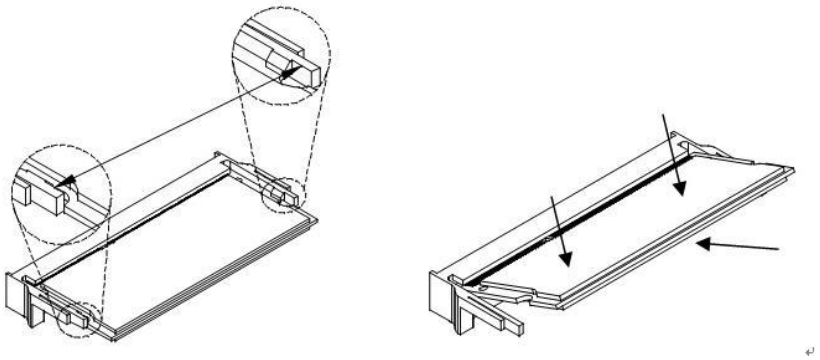
2.2 Installing the Memory

The IB897 board supports TWO DDR3L memory socket for a maximum total memory of 8GB DDR3L memory type.

Installing and Removing Memory Modules

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

1. Hold the DDR3 module so that the key of the DDR3 module 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.

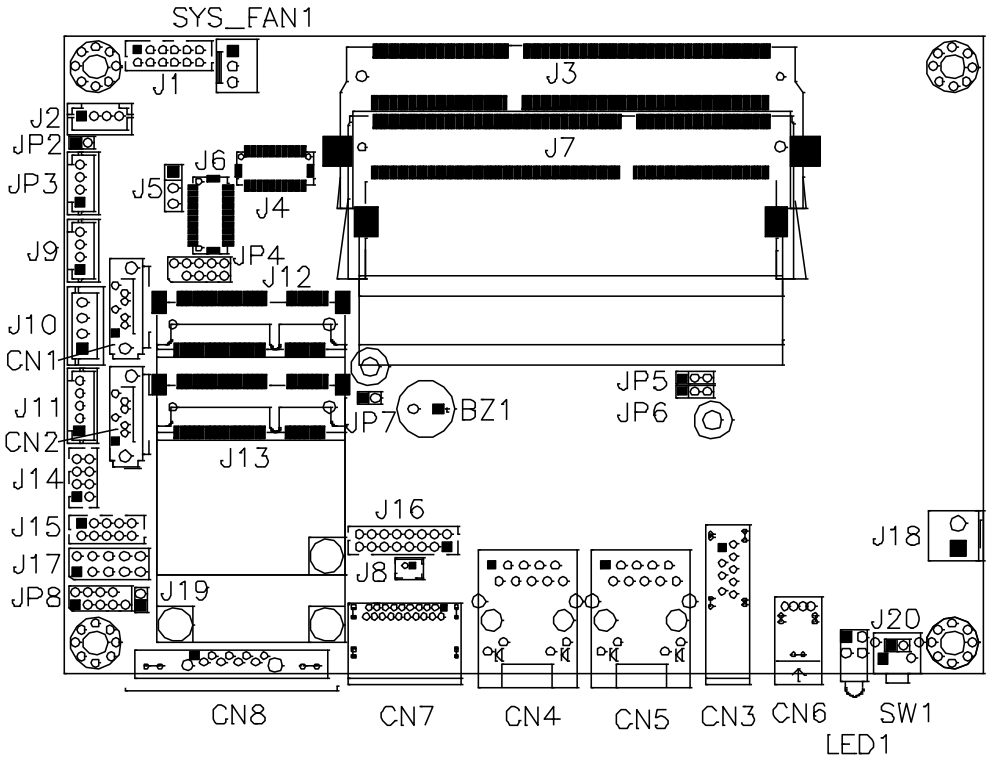


***** Channel-A slot must be installed with memory module for booting up*****

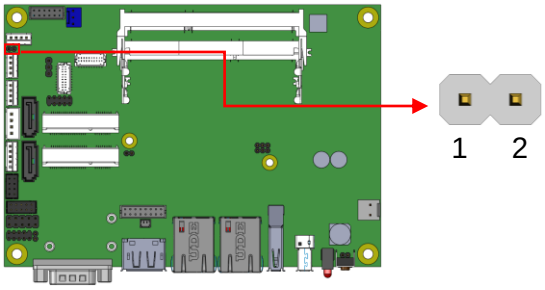
2.3 Setting Jumpers

Jumpers are used on IB897 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 IB897 and their respective functions.

Jumper Locations on IB897

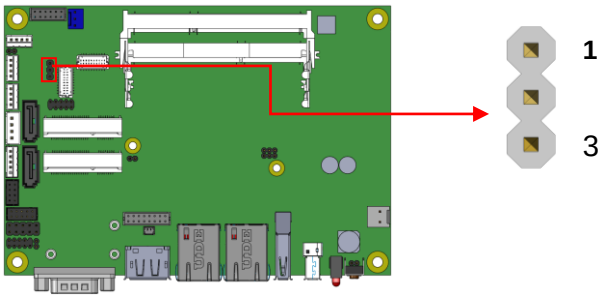


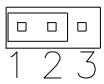
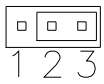
JP2: LVDS Panel Brightness Control Selection



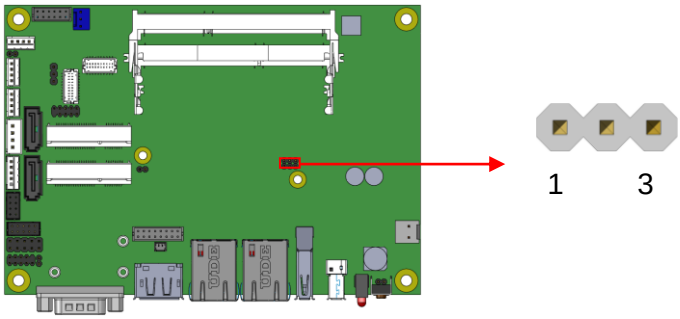
JP2	Brightness Control (PWM mode)
Open	3.3V
Close	5V(Default)

J5: LVDS Panel Power Selection



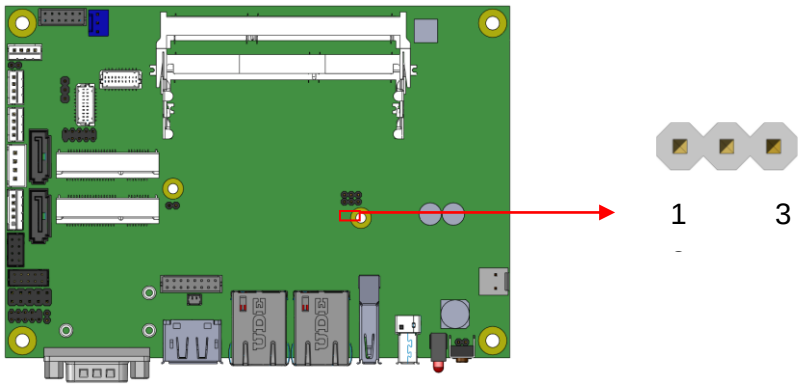
J5	Setting	Panel Voltage
	Pin 1-2 Short/Closed	3.3V (default)
	Pin 2-3 Short/Closed	5V

JP5: Clear ME Contents



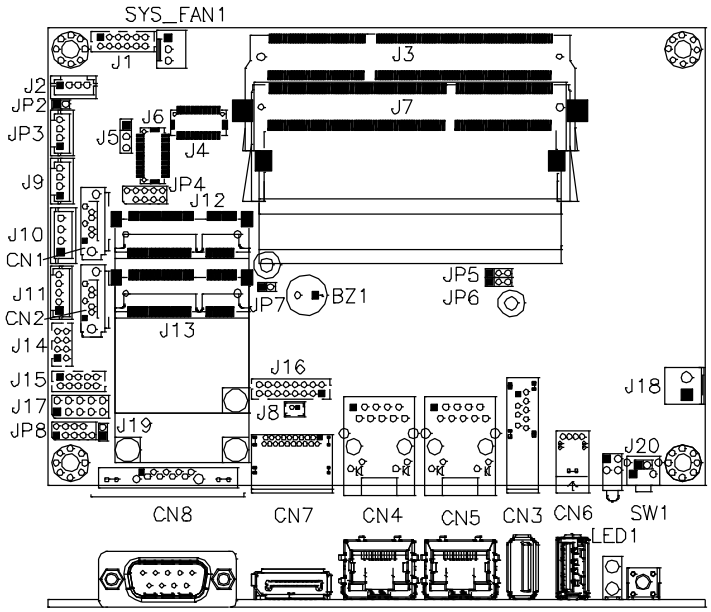
JP5	Setting	Function
	Pin 1-2 Short/Closed	Normal
	Pin 2-3 Short/Closed	Clear ME REGISTER

JP6: Clear CMOS Contents

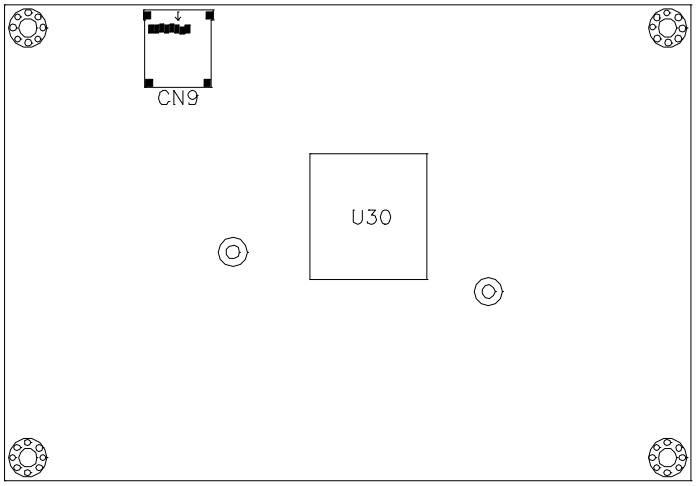


JP6	Setting	Function
	Pin 1-2 Short/Closed	Normal
	Pin 2-3 Short/Closed	Clear CMOS

Connector Locations



Bottom side



CN3: USB3.0 Connector

CN4, CN5: Gigabit LAN Connector

CN4: Intel® I210IT Connector

CN5: Intel® I210IT Connector

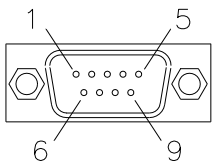
CN6: USB2.0 Connector

CN7: DP Connector

CN8: DB9 Connector (COM1)

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 is to be configured with BIOS Selection.



Pin #	Signal Name		
	RS-232	R2-422	RS-485
1	DCD	TX-	DATA-
2	RX	TX+	DATA+
3	TX	RX+	NC
4	DTR	RX-	NC
5	Ground	Ground	Ground
6	DSR	NC	NC
7	RTS	NC	NC
8	CTS	NC	NC
9	RI	NC	NC
10	NC	NC	NC

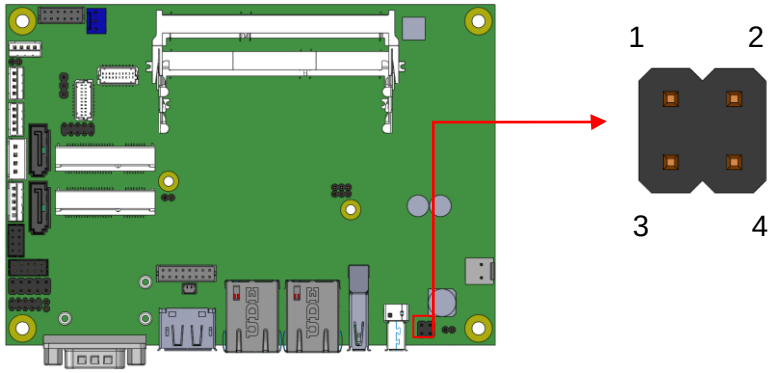
CN9: Micro SD (3.3V) Connector

SW1: Power Switch

LED1:

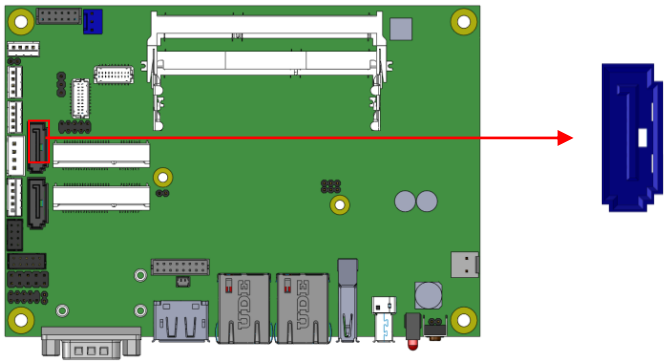
Power LED and HDD LED Connector

The green LED at the bottom is power LED. The red LED on top is the HDD LED.

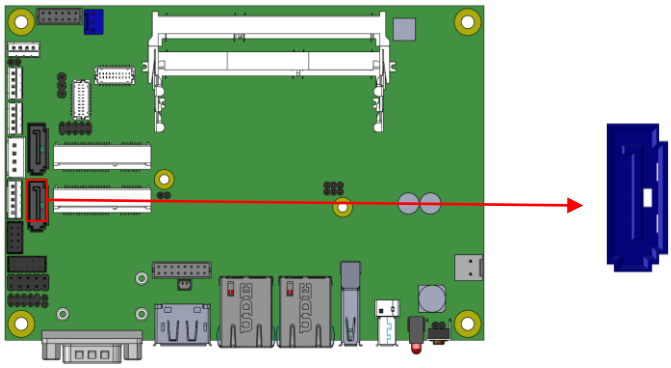


Signal Name	Pin #	Pin #	Signal Name
VCC3	1	2	HDD_LED
VCC5	3	4	GND

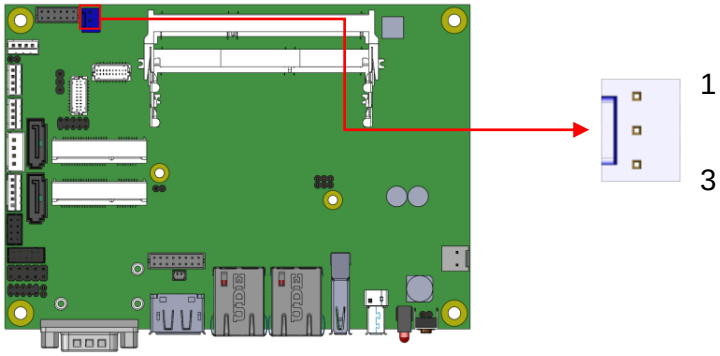
CN1: SATAII /share mSATA/ Connectors



CN2: SATAII Connectors

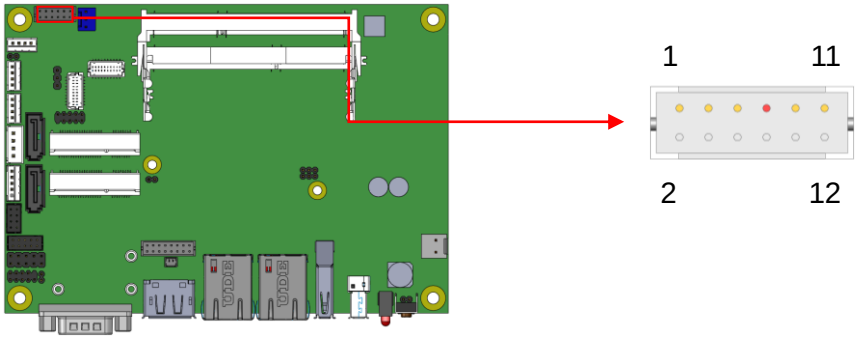


SYS_FAN1: SYSTEM Fan Power Connector



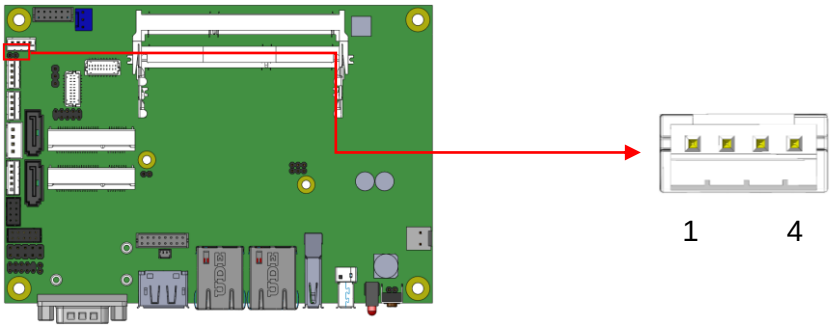
Pin #	Signal Name
1	Ground
2	+12V(500mA)
3	Rotation detection

J1: Audio Connector (DF11-12DP-2DSA)



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

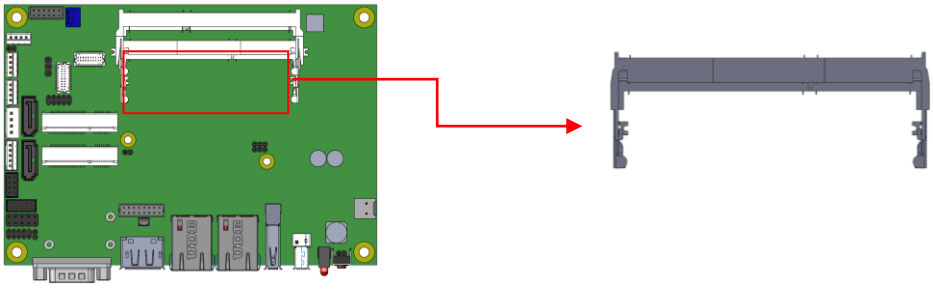
J2: Amplify Connector (JST B4B-PH-K-S)



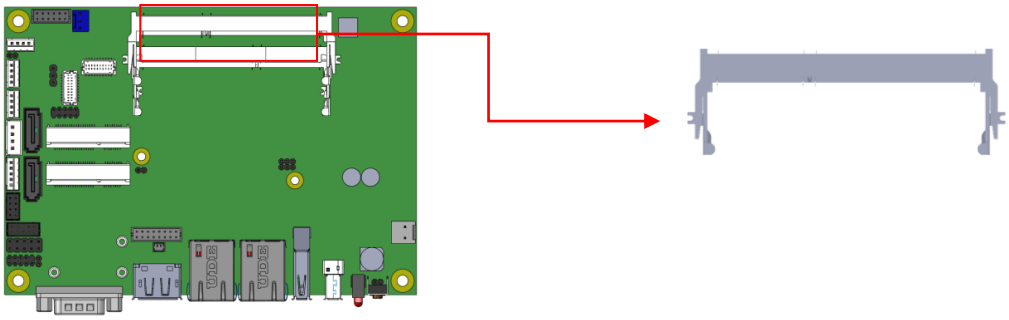
Pin #	Signal Name
1	OUTL+
2	OUTL-
3	OUTR-
4	OUTR+

J7: DDR3L SO-DIMM(CH-A) Sockets

**** Please note CH-A must be installed for booting up****

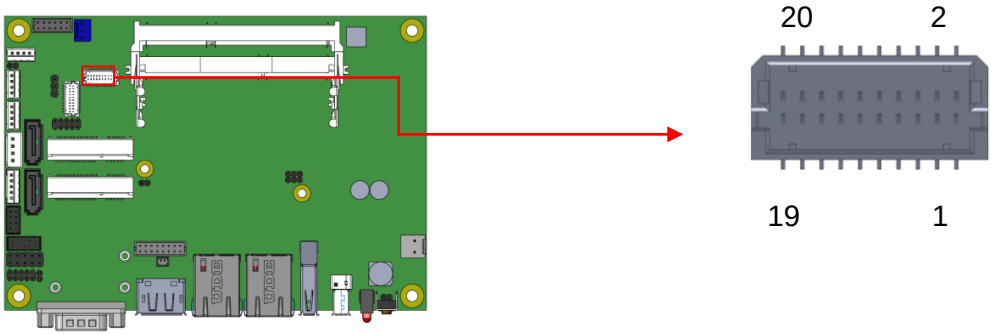


J3: DDR3L SO-DIMM(CH-B) Sockets

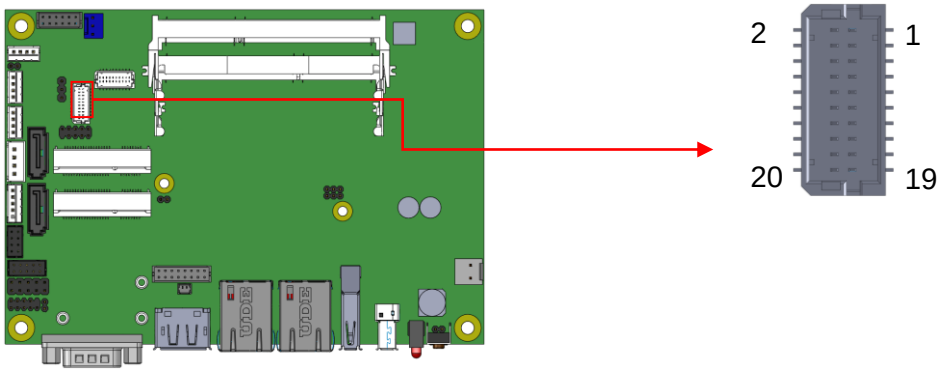


J4, J6: LVDS Connectors, (DF20G-20DP-1V)

J4: First Channel LVDS

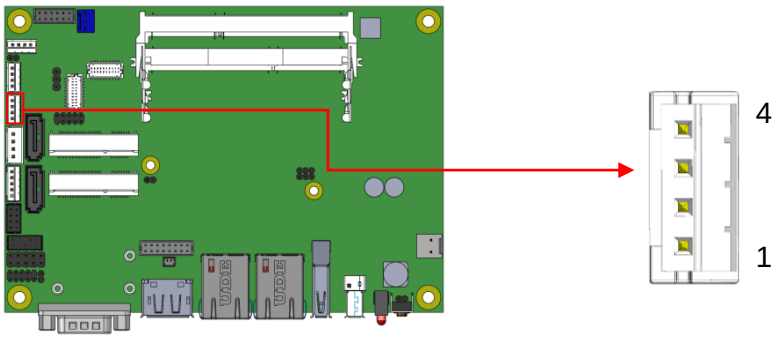


J6: Second Channel LVDS

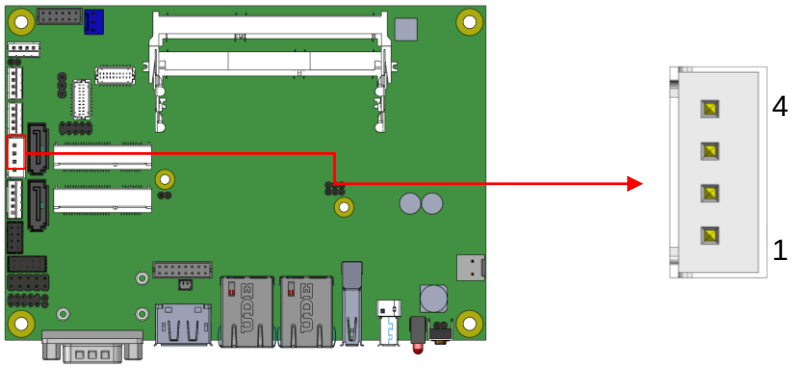


Signal Name	Pin #	Pin #	Signal Name
TX0N	2	1	TX0P
Ground	4	3	Ground
TX1N	6	5	TX1P
Ground	8	7	Ground
TX2N	10	9	TX2P
Ground	12	11	Ground
CLKN	14	13	CLKP
Ground	16	15	Ground
TX3N	18	17	TX3P
Power(1A)	20	19	Power

J9: MCU Flash Connector (factory use only)

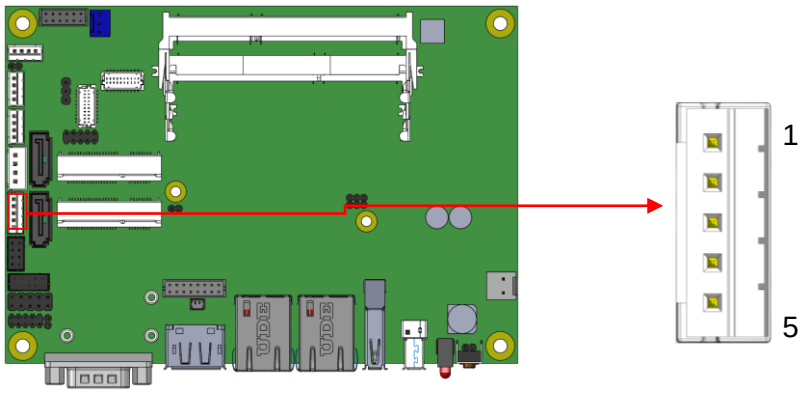


J10: SATA HDD Power Connectors(JST B4B-XH-A)



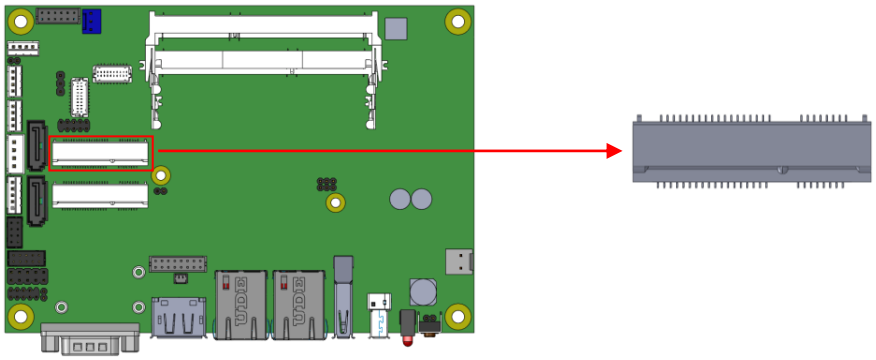
Pin #	Signal Name
1	+5V(1A)
2	Ground
3	Ground
4	+12V(1A)

J11: Smart Battery(JST B5B-PH-K-S)

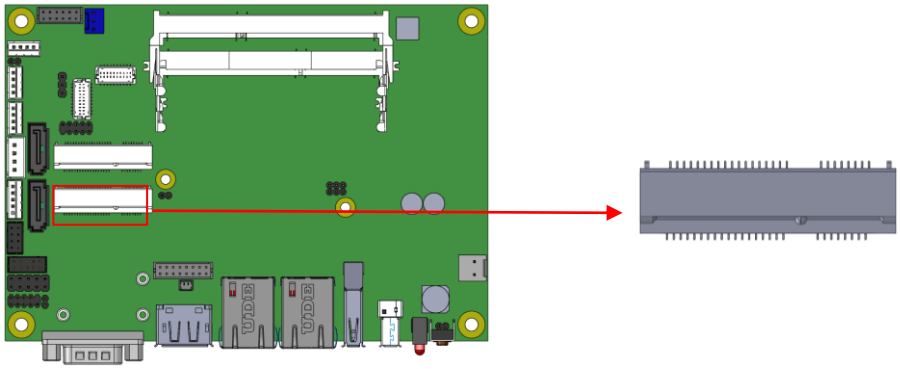


Pin #	Signal Name
1	RST#
2	ICHSWI#
3	Ground
4	SMB_DATA
5	SMB_CLK

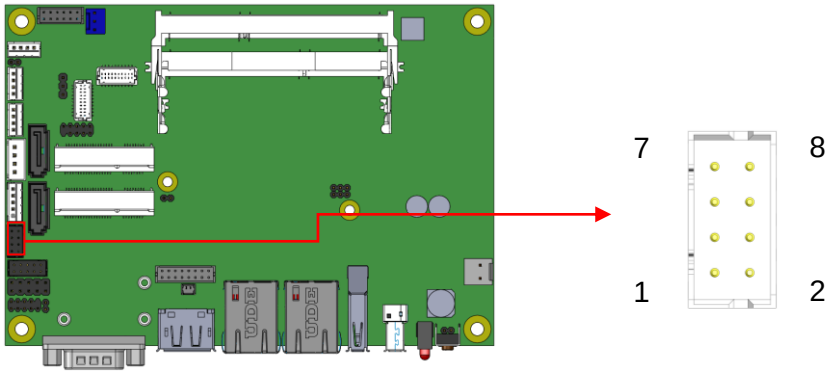
J12: Mini PCIE Connector (share mSATA)



J13: Mini PCIE Connector (Half Size)

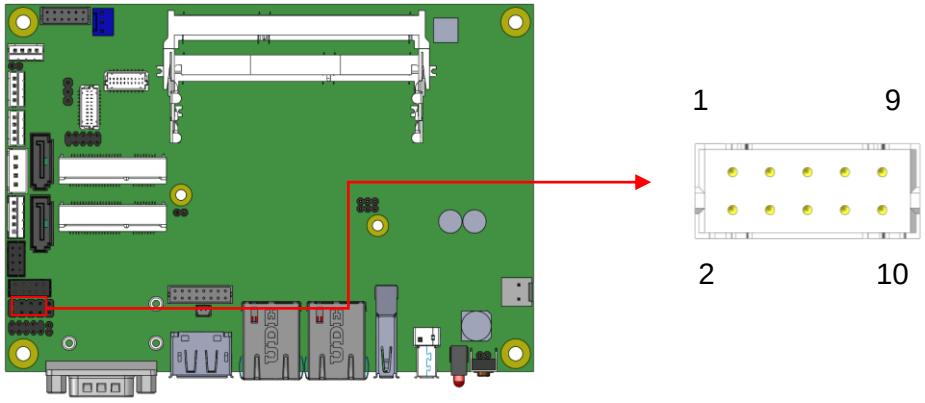


J14: USB 2.0 Connector(DF11-8DP-2DSA)



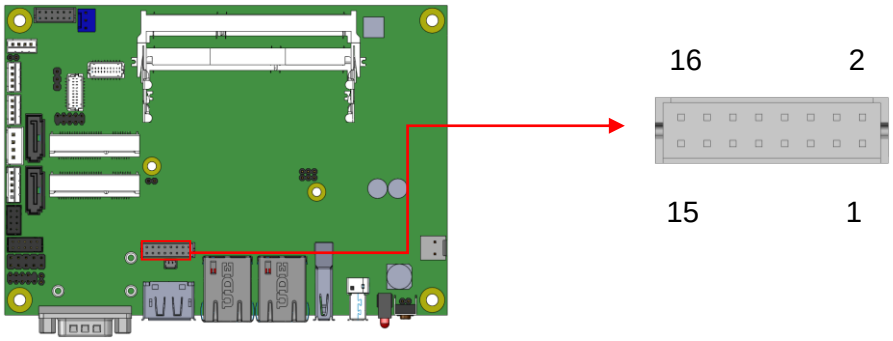
Signal Name	Pin #	Pin #	Signal Name
Vcc	1	2	Ground
D0-	3	4	D1+
D0+	5	6	D1-
Ground	7	8	Vcc

J15: COM2/RS232 Serial Port(DF11-10DP-2DSA)



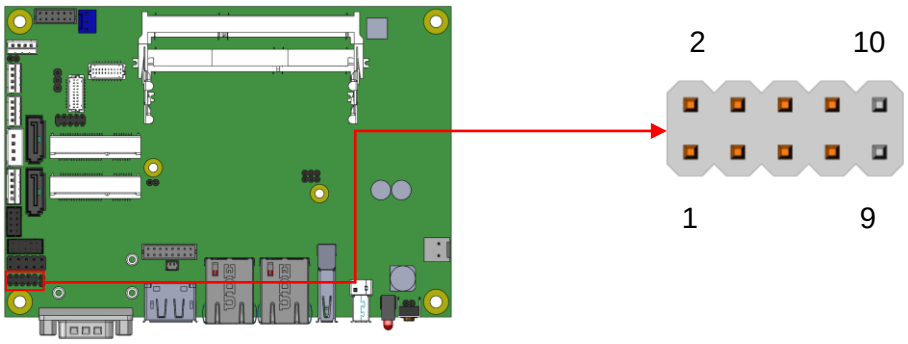
Signal Name	Pin #	Pin #	Signal Name
DCD, Data carrier detect	1	2	RXD, Receive data
TXD, Transmit data	3	4	Data terminal ready
GND, ground	5	6	DSR, Data set ready
RTS, Request to send	7	8	CTS, Clear to send
RI, Ring indicator	9	10	Not Used

J16: VGA Connector (DF11-16DP-2DSA)



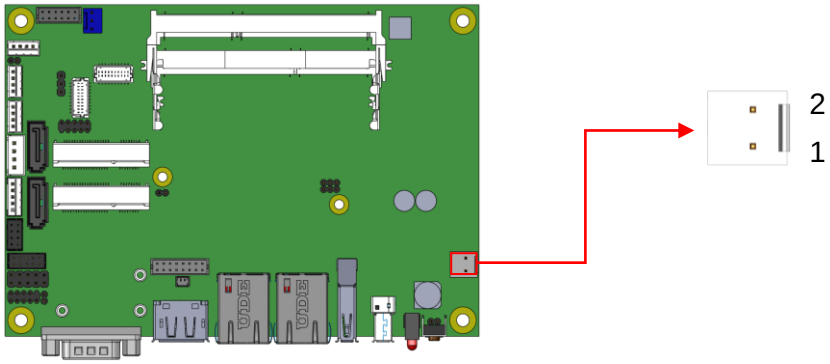
Signal Name	Pin #	Pin #	Signal Name
+5V	2	1	Red
Ground	4	3	Green
N.C	6	5	Blue
DDCDATA	8	7	N.C
H_SYNC	10	9	GND
V_SYNC	12	11	GND
DDCCLK	14	13	GND
N.C.	16	15	GND

J17: Digital I/O(signal level 5V)Connector(2.54mm)



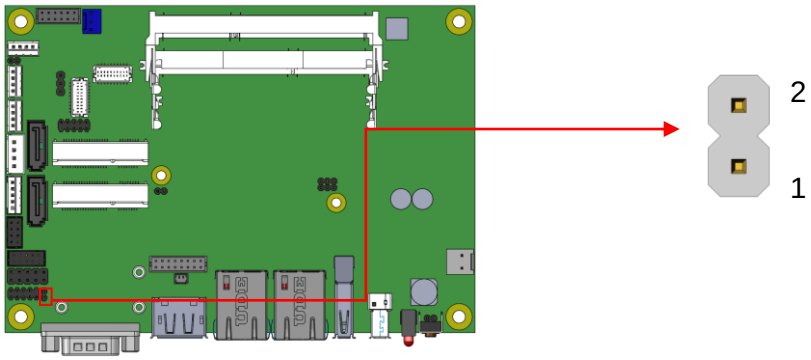
Signal Name	Pin #	Pin #	Signal Name
GND	1	2	VCC(500mA)
OUT3	3	4	OUT1
OUT2	5	6	OUT0
IN3	7	8	IN1
IN2	9	10	IN0

J18: Board Input Power Connector(HK_WAFER396-2S-WV)



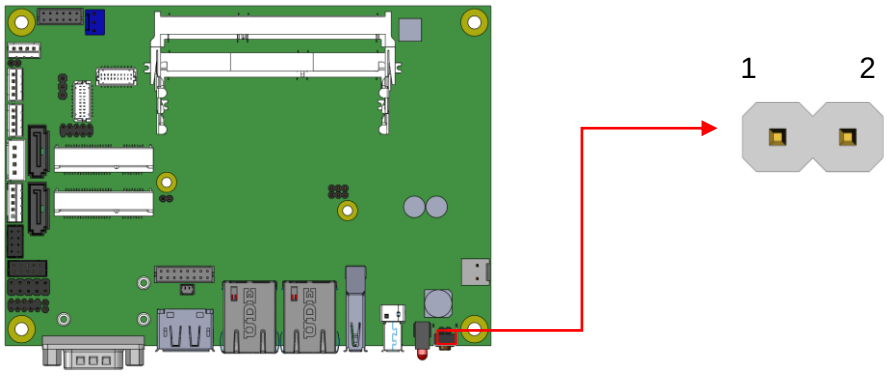
Pin #	Signal Name
1	+9V to +30V(10A)
2	GND

J19: Reset Switch(2mm)



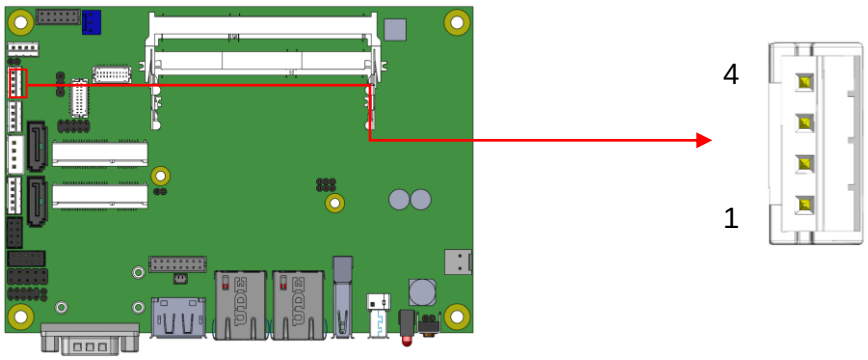
Pin #	Signal Name
1	Reset Switch
2	Ground

J20: Power Switch(2mm)



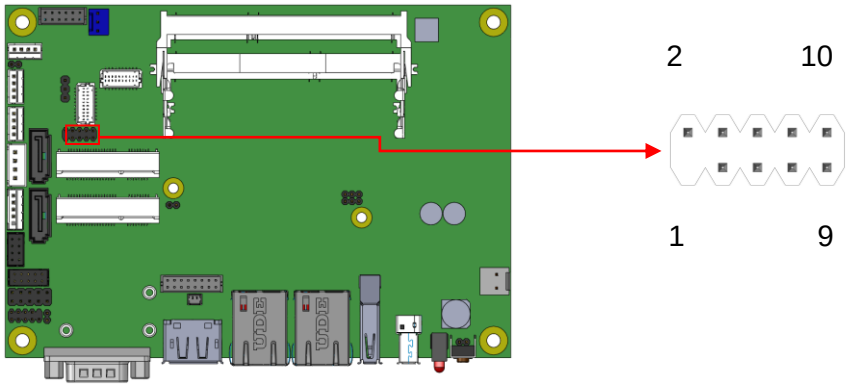
Pin #	Signal Name
1	Power Switch
2	Ground

JP3: LCD Backlight Connector(JST B4B-PH-K-S)

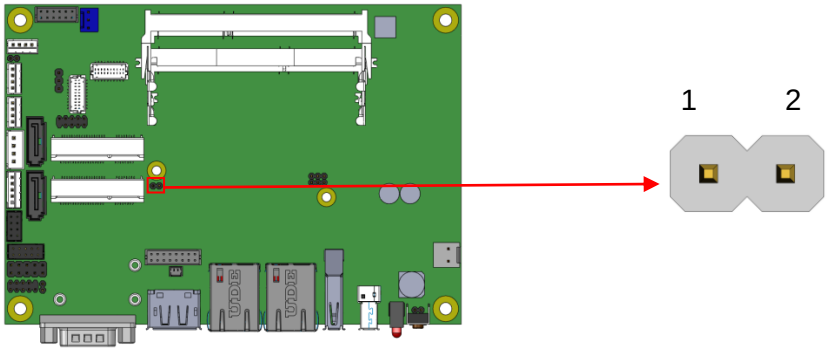


Pin #	Signal Name
1	+12V(1A)
2	Backlight Enable
3	Brightness Control
4	Ground

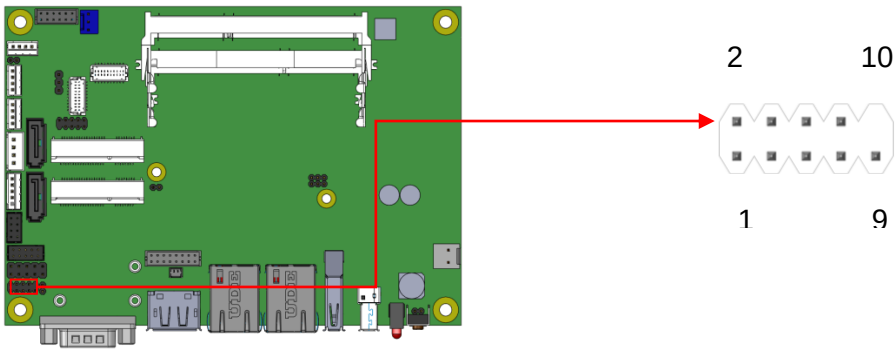
JP4: SPI Flash Connector (factory use only)



JP7: Factory use only



JP8: Debug 80 Port Connector (factory use only)



CHAPTER 3 BIOS SETUP

3.1 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.

3.2 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:

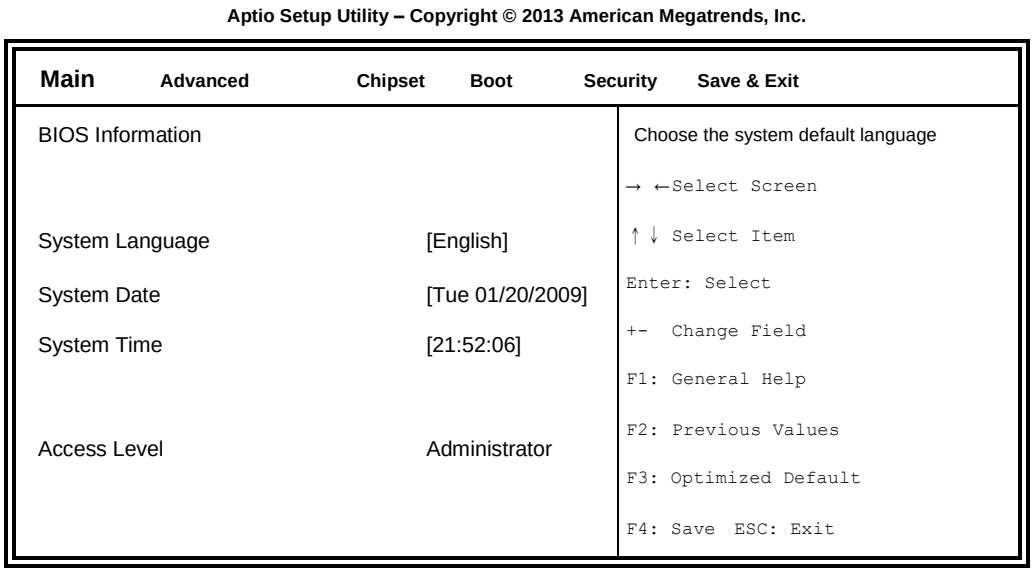
```
Press <DEL> to Enter Setup
```

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.*

3.3 Main Settings



System Language

Choose the system default language.

System Date

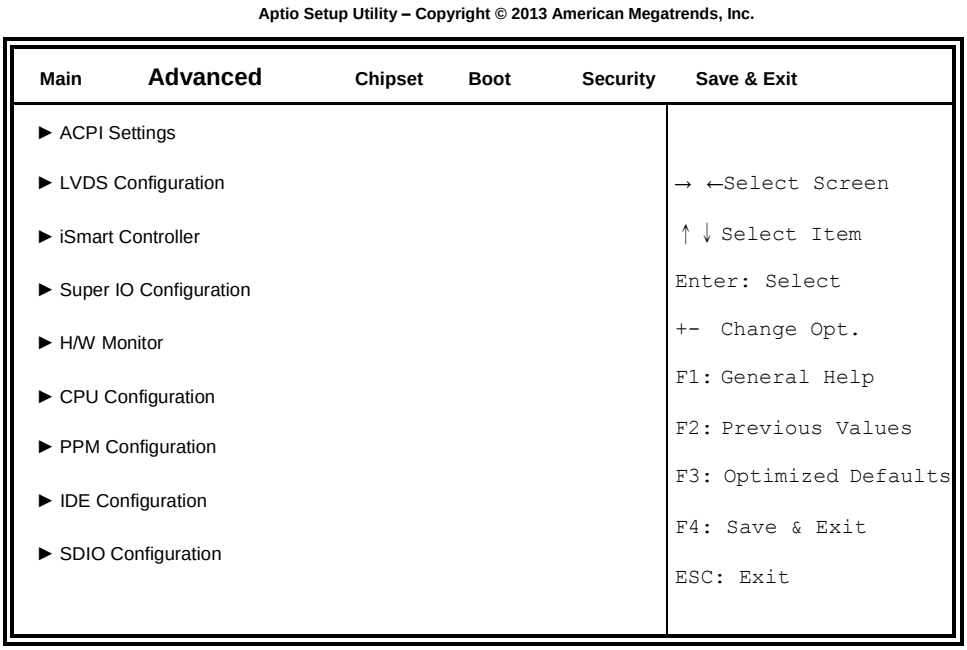
Set the Date. Use Tab to switch between Data elements.

System Time

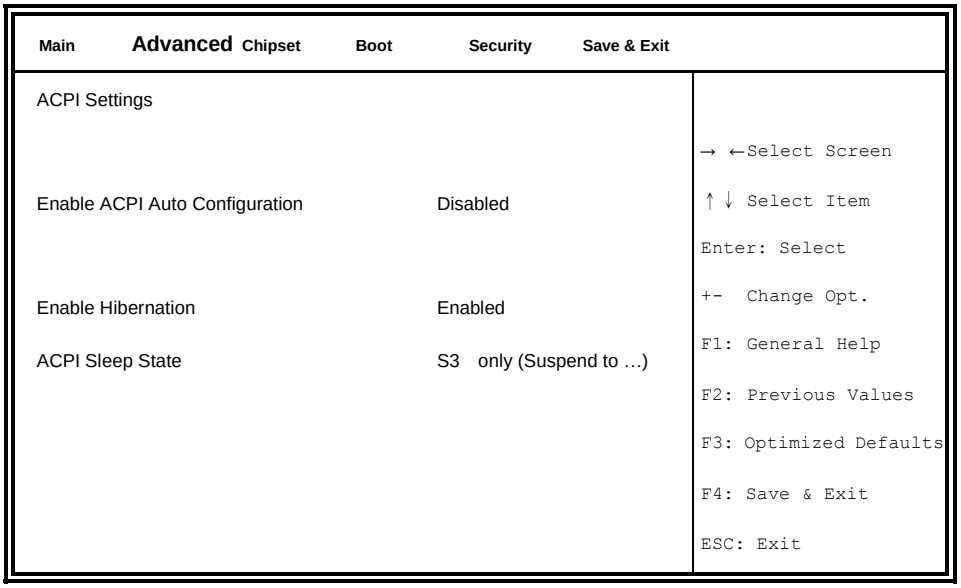
Set the Time. Use Tab to switch between Data elements.

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.



ACPI Settings



Enabled ACPI Auto Configuration

Enables or Disables BIOS ACPI Auto Configuration.

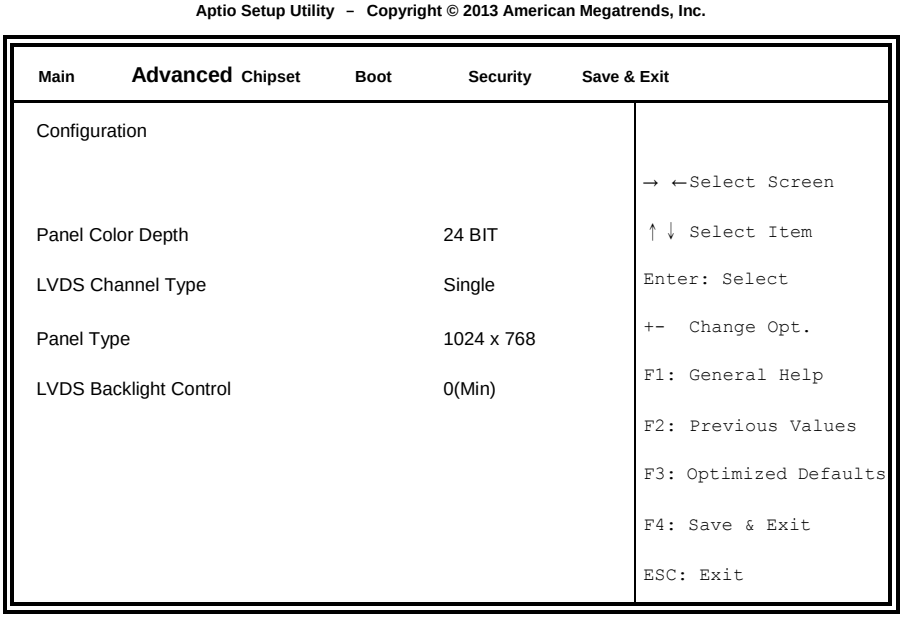
Enable Hibernation

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

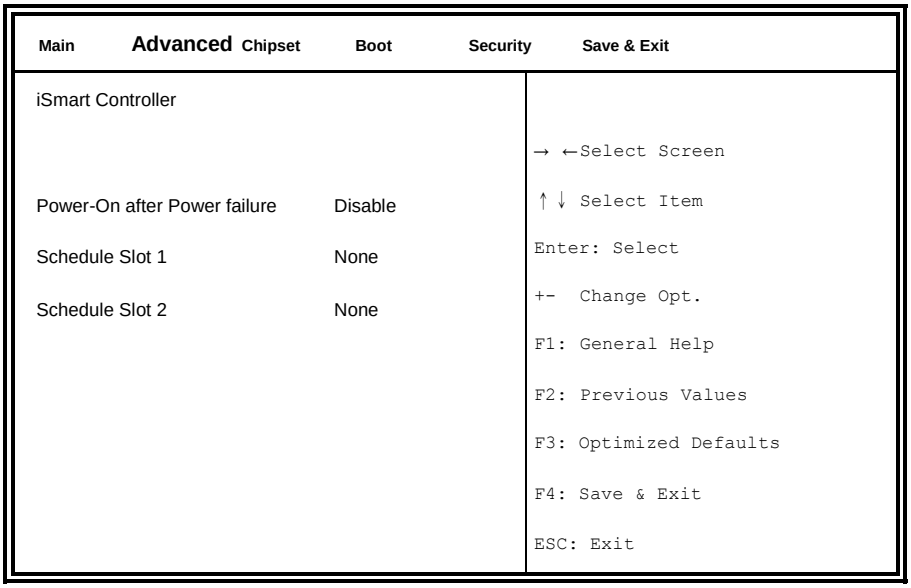
ACPI Sleep State

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

LVDS Configuration



iSmart Controller



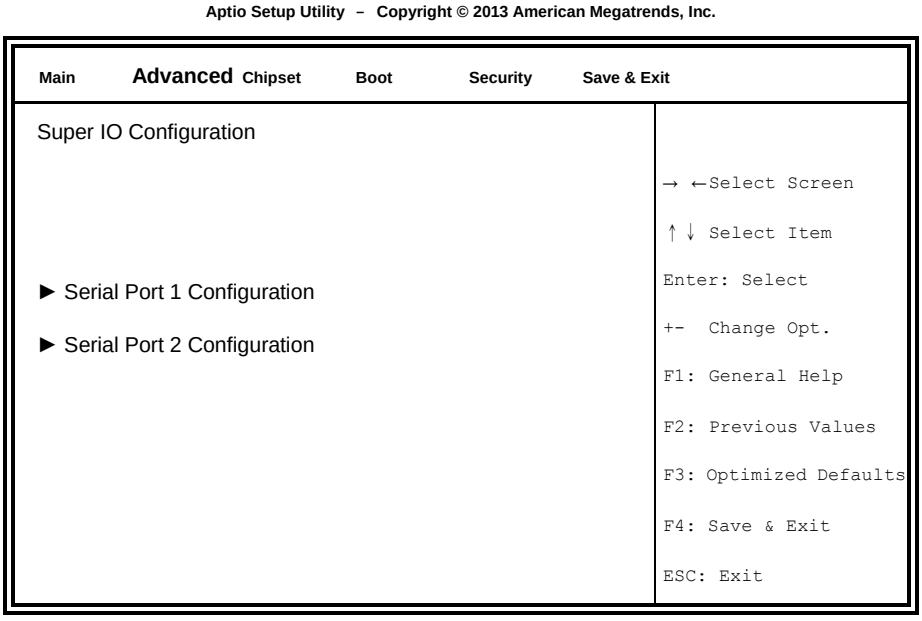
Power-On after Power failure

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

Schedule Slot 1 / 2

Setup the hour/minute for system power on.

Super IO Configuration



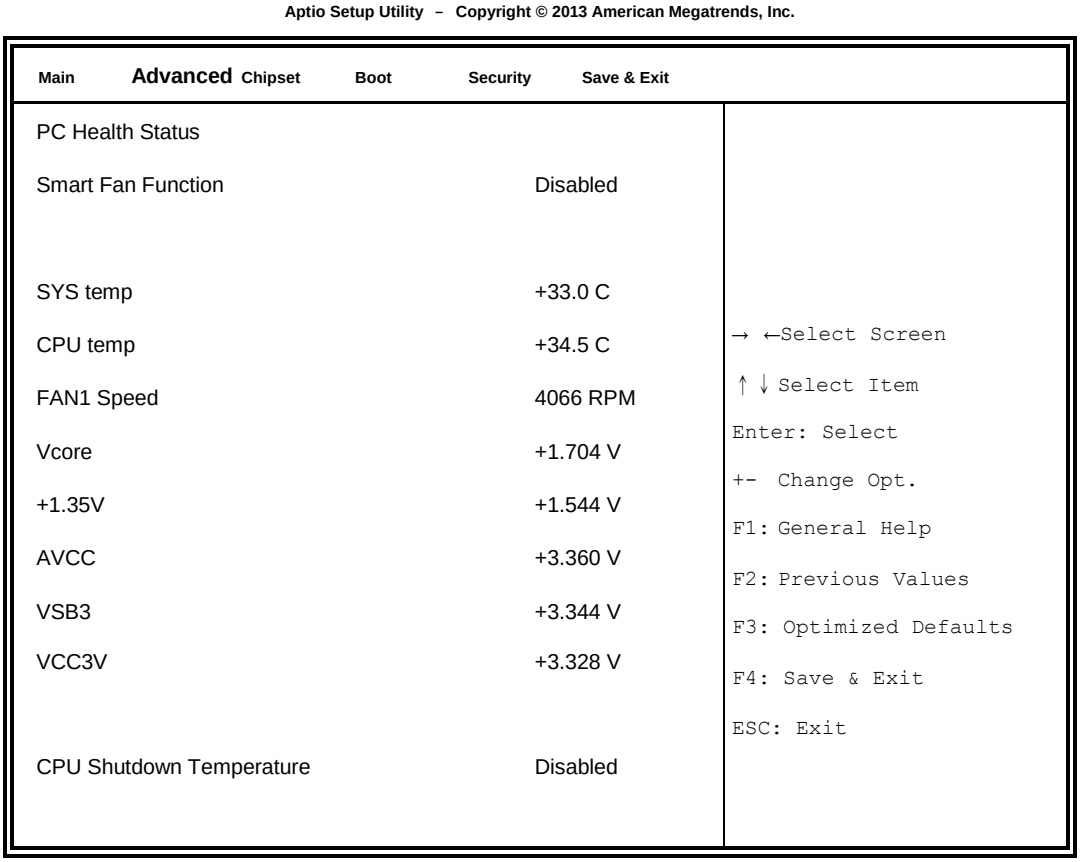
Serial Port 1 Configuration

Set parameters of serial port 1(COMA)

Serial Port 2 Configuration

Set parameters of serial port 2(COMA)

H/W Monitor



Smart Fan Function

This field enables or disables the smart fan feature.

Disabled (default)

50 ℃

60 ℃

70 ℃

80 ℃

90 ℃

Shutdown Temperature

This field enables or disables the Shutdown Temperature

Disabled (default)

70 ℃/158 F

75 ℃/167 F

80 ℃/176 F

85 ℃/185 F

90 ℃/194 F

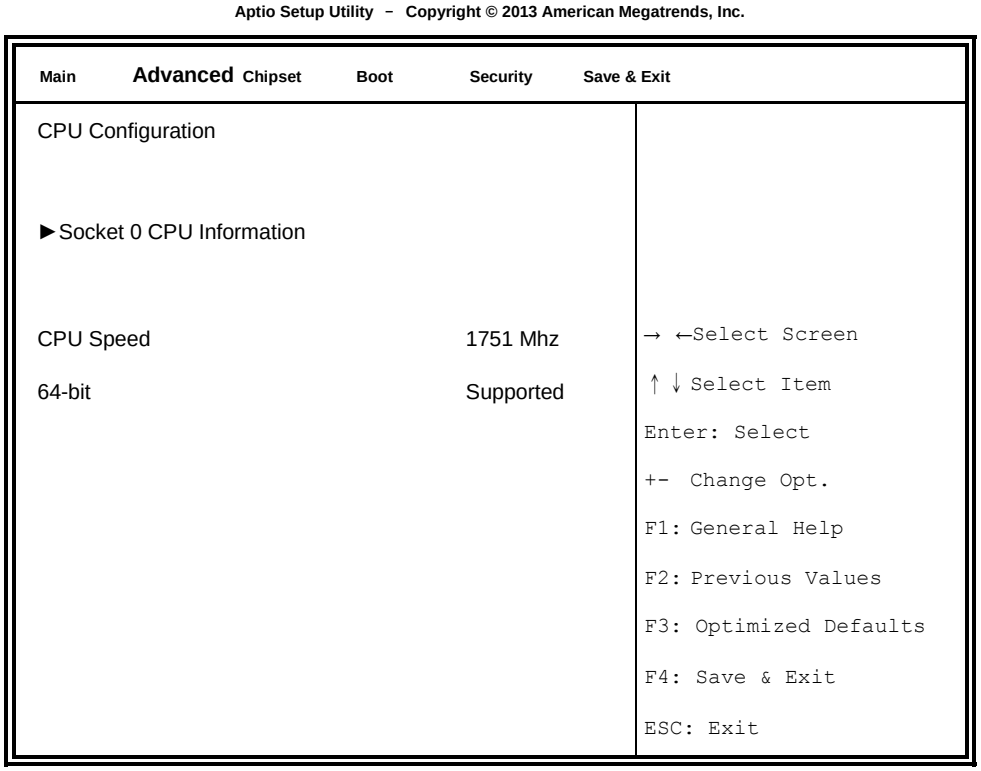
90 ℃/203 F

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 Configuration

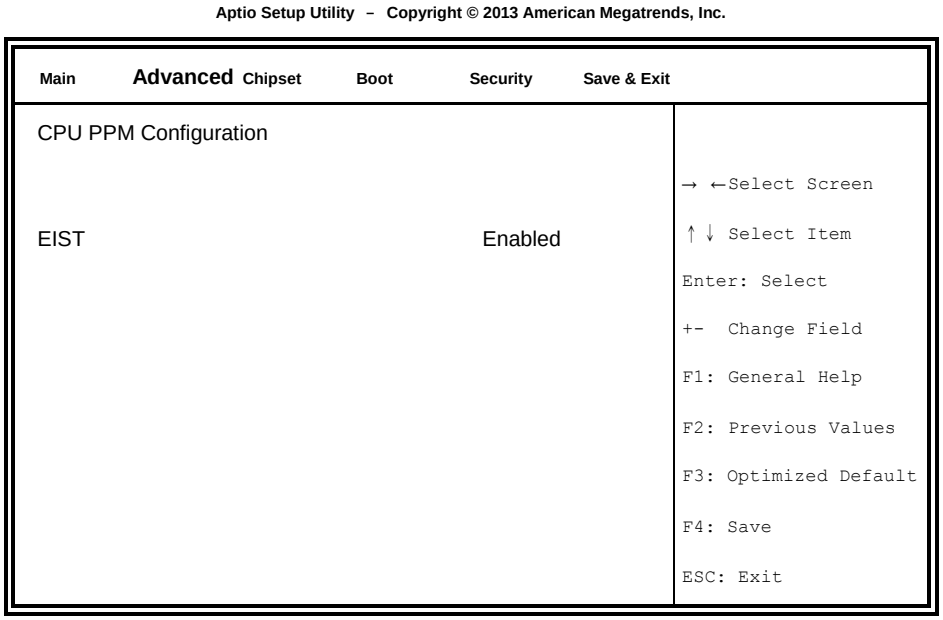
This section shows the CPU configuration parameters.



Socket 0 CPU Information

Socket specific CPU Information.

CPU PPM Configuration

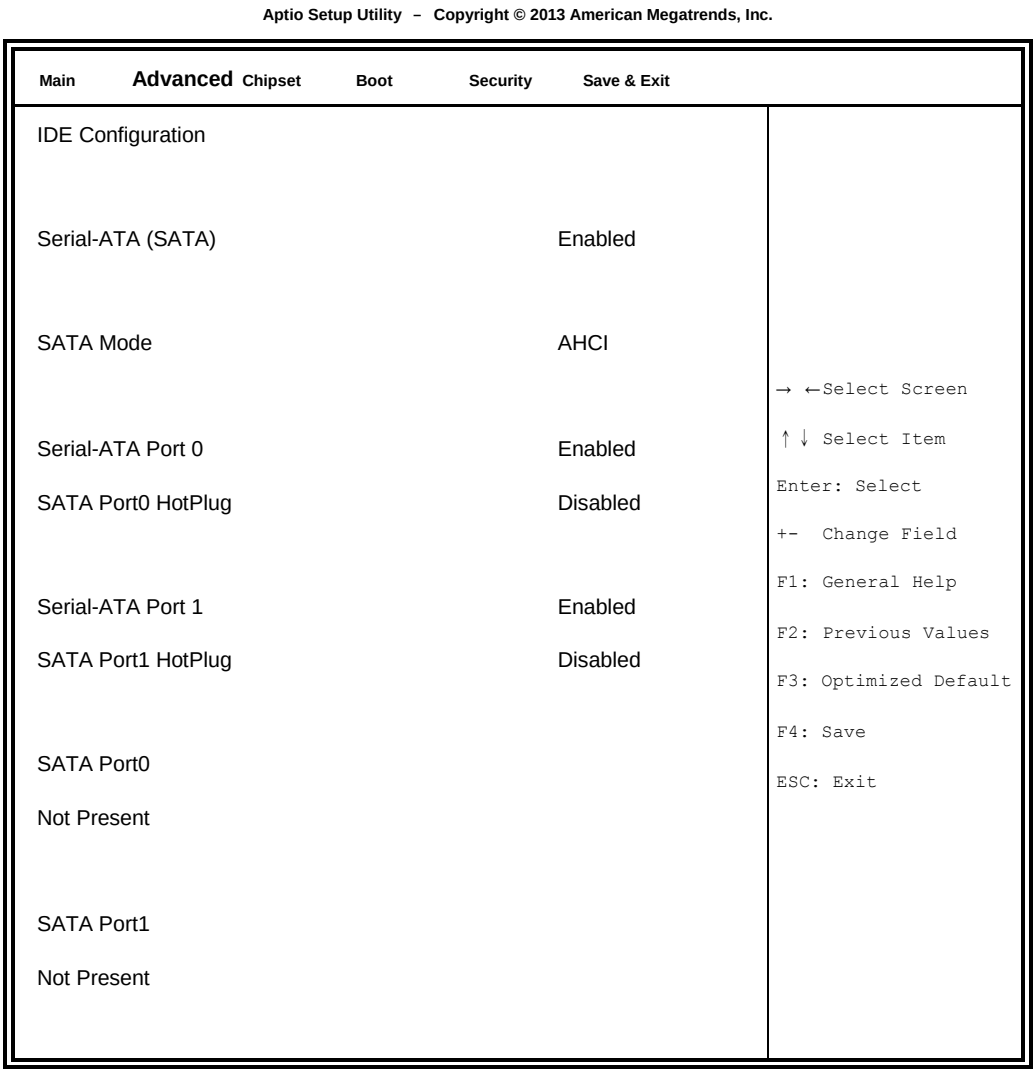


EIST

Enable/Disable Intel SpeedStep.

IDE Configuration

SATA Devices Configuration.



Serial-ATA(SATA)

Enabled / Disabled Serial ATA

SATA Mode

Select IDE / AHCI Mode

Serail –ATA Port 0

Enabled / Disabled Serial Port 0

SATA Port0 HotPlug

Enabled / Disabled SATA Port 0 HotPlug

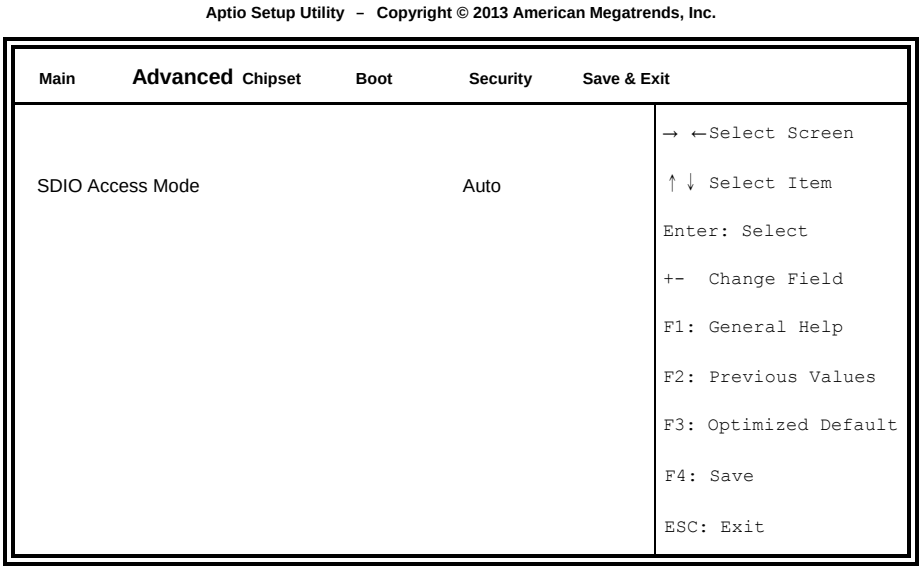
Serail –ATA Port 1

Enabled / Disabled Serial Port 1

SATA Port1 HotPlug

Enabled / Disabled SATA Port 1 HotPlug

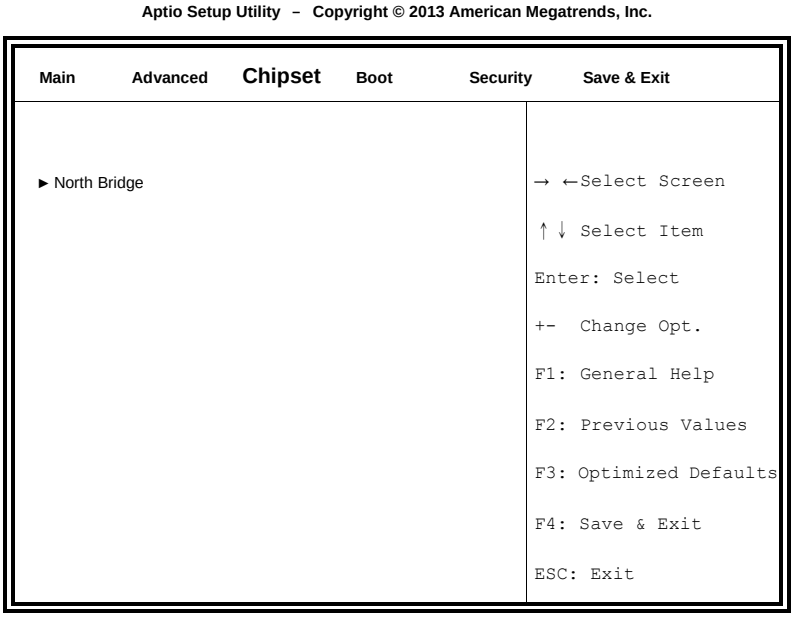
SDIO Configuration



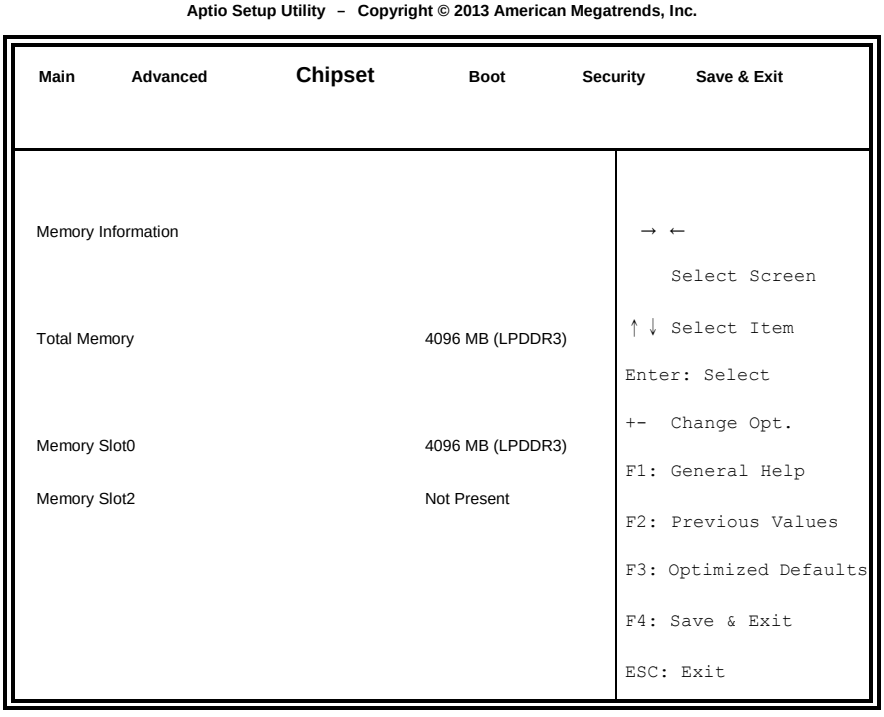
SDIO Access Mode

Auto Option: Access SD device in DMA mode if controller supports it. Otherwise, in PIO mode. DMA options: Access SD device in DMA mode. PIO Option: Access PIO device in DMA

Chipset Settings

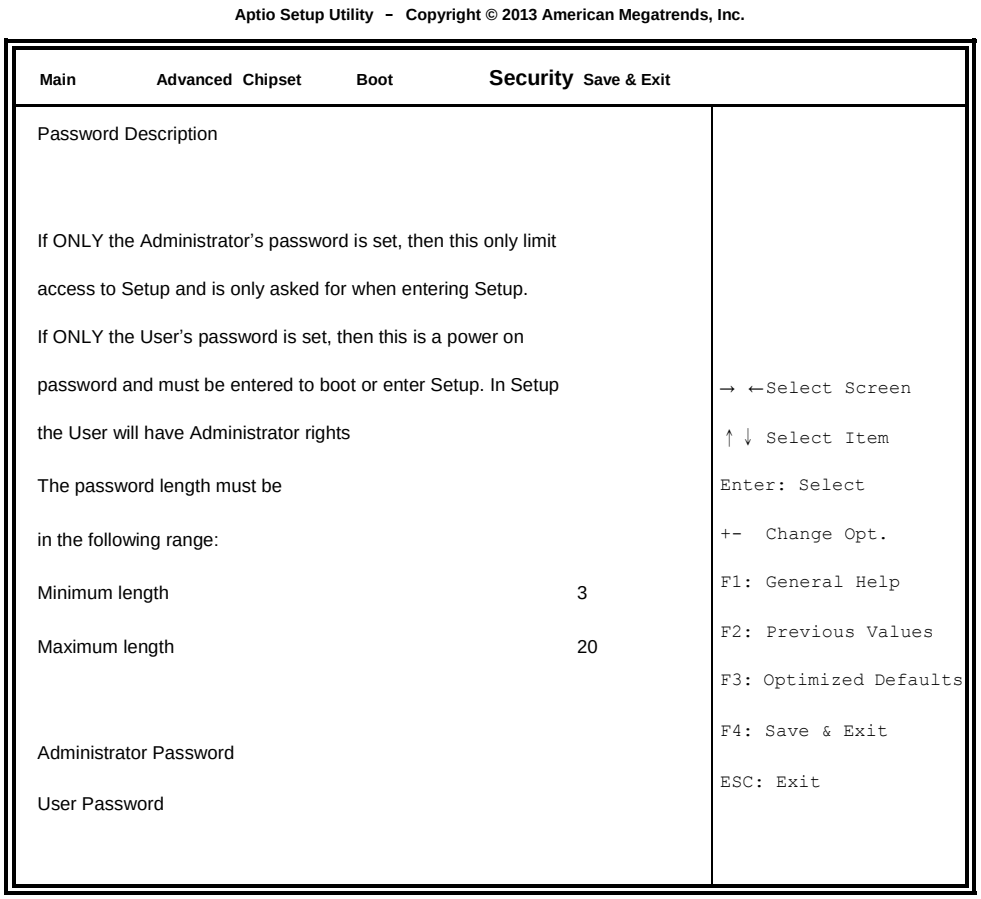


North Bridge



3.4 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.

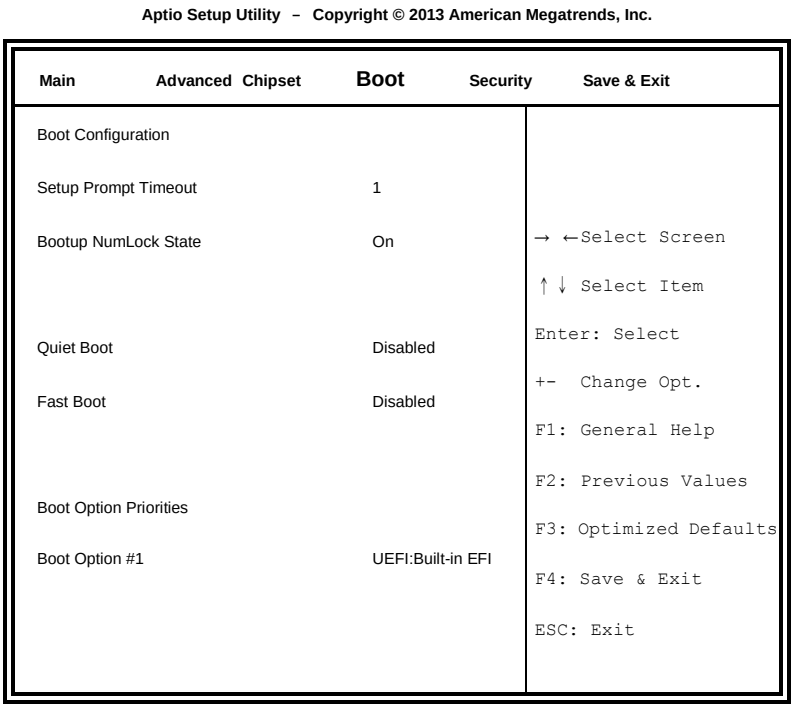


Administrator Password

Set Administrator Password.

Boot Settings

This section allows you to configure the boot settings.



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 or disables Quiet Boot option.

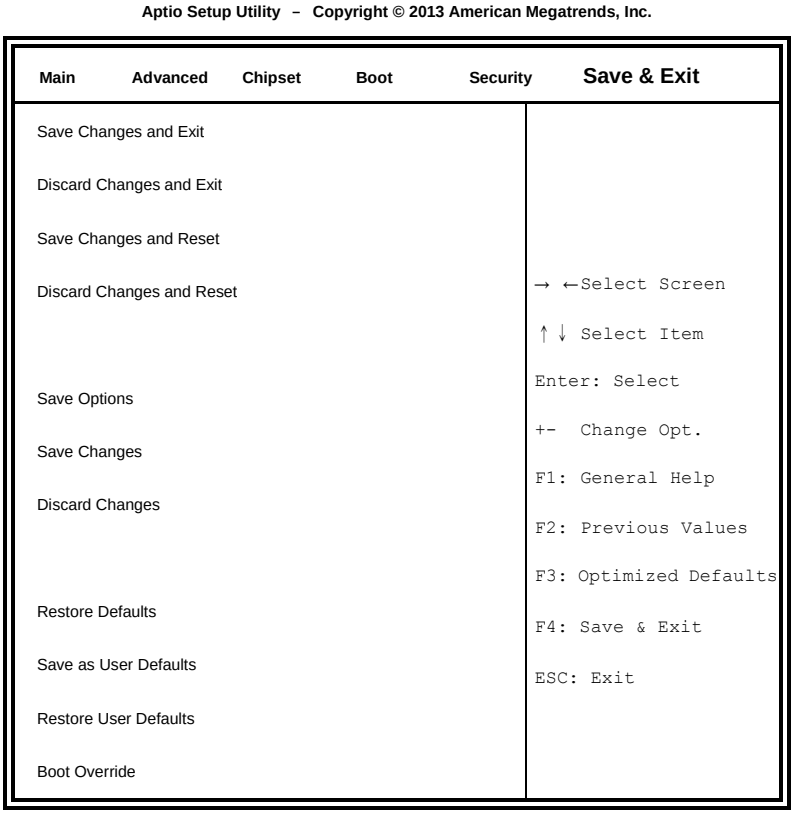
Fast Boot

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

Boot Option Priorities

Sets the system boot order.

Save & Exit Settings



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.

CHAPTER 4 DRIVERS INSTALLATION

This section describes the installation procedures for software and drivers. The software and drivers are included with the motherboard

IMPORTANT NOTE:

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

4.1 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 DVD that comes with the board. Click ***Intel*** and then ***Intel(R) Baytrail Chipset***.

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

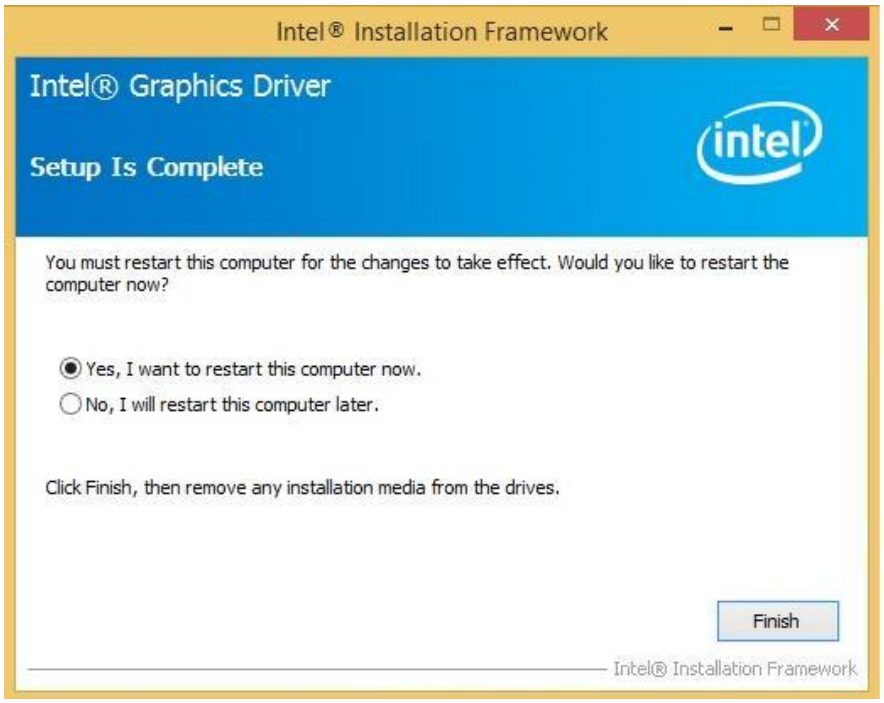
4.2 VGA Drivers Installation

1. Insert the DVD that comes with the board. Click ***Intel*** and then ***Intel(R) Baytrail Chipset***.Click ***Intel(R) Baytrail Graphics Driver***

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

3. Click ***Yes*** to accept the license agreement and continue the installation.

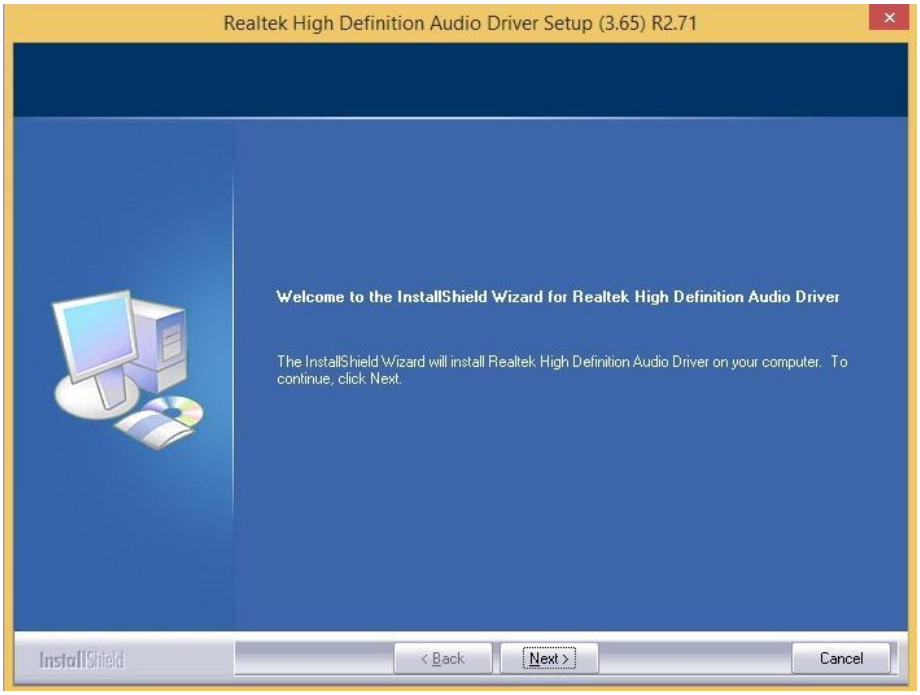
4. Setup complete. Click ***Finish*** to restart the computer and for changes to take effect.



4.3 Realtek High Definition Audio Driver Installation

1. Insert the DVD that comes with the board. Click ***Intel*** and then ***Intel(R) Baytrail Chipset***. Click ***Realtek High Definition Audio Driver.***

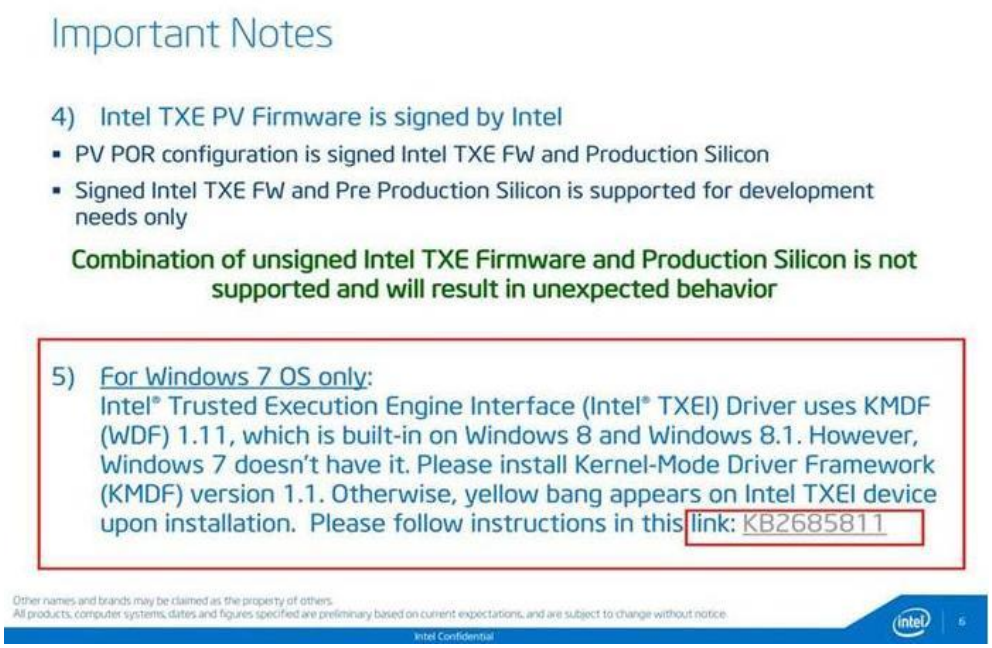
2. On the Welcome screen, click ***Next*** to proceed with the installation.



3. InstallShield Wizard is complete. Click ***Finish*** to restart the computer and for changes to take effect.

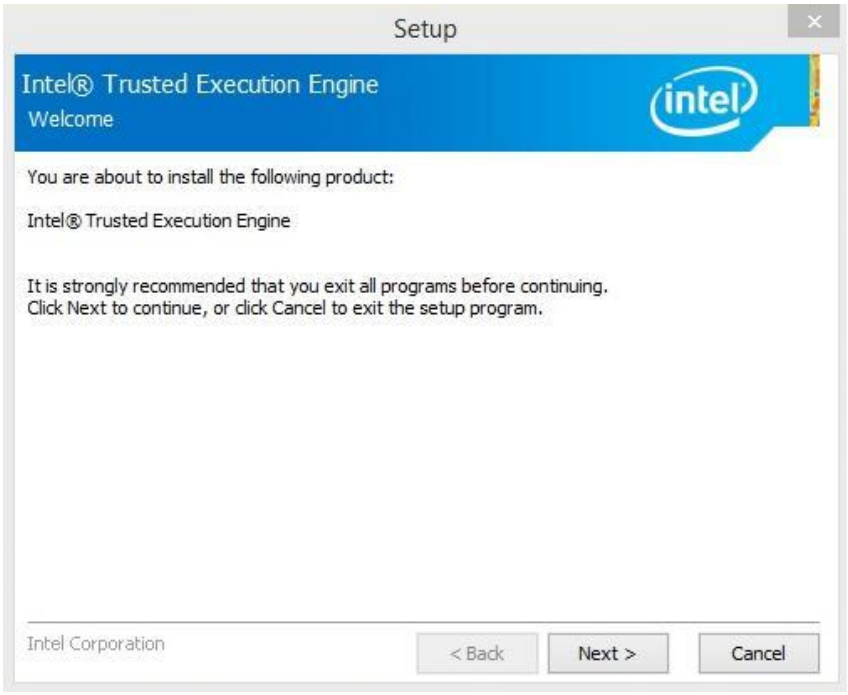
4.4 Intel Trusted Execution Engine Installation

Note :Windows 7 OS only



1. Insert the DVD that comes with the board. Click ***Intel*** and then ***Intel(R) Baytrail Chipset***. Click ***Intel(R) Baytrail Graphics Driver.***

2. On the Setup Welcome screen, click ***Next*** to proceed with the installation process.



3. Click ***Next*** accept the license agreement and continue the installation.

4. Installation of the Intel Trusted Execution Engine is now 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.

Address	Device Description
0000h-001Fh	Direct memory access controller
0000h-001Fh	PCI bus
0040h-0043h	System timer
0050h-0053h	System timer
0070h-0077h	System CMOS/real time clock
0081h-0091h	Direct memory access controller
0093h-009Fh	Direct memory access controller
00C0h-00DFh	Direct memory access controller
00F0h-00F0h	Numeric data processor
02F8h-02FFh	Communications Port (COM2)
03B0h-03BBh	Intel(R) HD Graphics 4600
03C0h-03DFh	Intel(R) HD Graphics 4600
03F8h-03FFh	Communications Port (COM1)
0D00h-FFFFh	PCI bus
E000h-EFFFh	Intel(R) 8 Series/C220 Series PCI Express Root Port #7 - 8C1C
F000h-F03Fh	Intel(R) HD Graphics 4600
F040h-F05Fh	Intel(R) 8 Series/C220 Series SMBus Controller - 8C22
F060h-F07Fh	Intel(R) 8 Series/C220 Series SATA AHCI Controller - 8C02
F0A0h-F0A3h	Intel(R) 8 Series/C220 Series SATA AHCI Controller - 8C02
F0B0h-F0B7h	Intel(R) 8 Series/C220 Series SATA AHCI Controller - 8C02

F0C0h-F0C3h	Intel(R) 8 Series/C220 Series SATA AHCI Controller - 8C02
F0D0h-F0D7h	Intel(R) 8 Series/C220 Series SATA AHCI Controller - 8C02
F0E0h-F0E7h	Intel(R) Active Management Technology - SOL (COM3)

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
IRQ3	Serial Port #2
IRQ4	Serial Port #1
IRQ 10	Intel(R) 8 Series/C220 Series SMBus Controller - 8C22
IRQ 13	Numeric data processor
IRQ 16	High Definition Audio Controller
IRQ 16	Intel(R) 8 Series/C220 Series USB EHCI #2 - 8C2D
IRQ 16	Intel(R) Management Engine Interface
IRQ 19	Intel(R) 8 Series/C220 Series SATA AHCI Controller - 8C02
IRQ 19	Intel(R) Active Management Technology - SOL (COM3)
IRQ 22	High Definition Audio Controller
IRQ 23	Intel(R) 8 Series/C220 Series USB EHCI #1 - 8C26

C. Digital I/O Sample Code

File of the NCT5523D.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 __NCT5523D_H

#define __NCT5523D_H                                    1

//---------------------------------------------------------------------------

#define     NCT5523D_INDEX_PORT                         (NCT5523D_BASE)

#define     NCT5523D_DATA_PORT                          (NCT5523D_BASE+1)

//---------------------------------------------------------------------------

#define     NCT5523D_REG_LD                             0x07

//---------------------------------------------------------------------------

#define     NCT5523D_UNLOCK                             0x87

#define     NCT5523D_LOCK                               0xAA

//---------------------------------------------------------------------------

unsigned int Init_NCT5523D(void);

void Set_NCT5523D_LD( unsigned char);

void Set_NCT5523D_Reg( unsigned char, unsigned char);

unsigned char Get_NCT5523D_Reg( unsigned char);

//---------------------------------------------------------------------------

#endif          //__NCT5523D_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 "NCT5523D.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_NCT5523D();
      if (SIO == 0)
      {
                printf("Can not detect Nuvoton NCT5523D, program abort.\n");
                return(1);
       }

    Dio5Initial();

    //for GPIO20..27
    Dio5SetDirection(0x0F);    //GP20..23 = input, GP24..27=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;

      ucBuf = Get_NCT5523D_Reg(0x1C);
      ucBuf &= ~0x02;
      Set_NCT5523D_Reg(0x1C, ucBuf);

       Set_NCT5523D_LD(0x07);
       //switch to logic device 7                              //enable the GP2 group
      ucBuf = Get_NCT5523D_Reg(0x30);
      ucBuf |= 0x04;
      Set_NCT5523D_Reg(0x30, ucBuf);
}
//---------------------------------------------------------------------------
void Dio5SetOutput(unsigned char NewData)
{
      Set_NCT5523D_LD(0x07);                                   //switch to logic device 7
      Set_NCT5523D_Reg(0xE1, NewData);
}
//---------------------------------------------------------------------------
unsigned char Dio5GetInput(void)
{
    unsigned char result;

    Set_NCT5523D_LD(0x07);                                     //switch to logic device 7
    result = Get_NCT5523D_Reg(0xE1);
    return (result);
}
//---------------------------------------------------------------------------
void Dio5SetDirection(unsigned char NewData)
{
    //NewData : 1 for input, 0 for output
    Set_NCT5523D_LD(0x07);                                     //switch to logic device 7
    Set_NCT5523D_Reg(0xE8, NewData);
}
//---------------------------------------------------------------------------
unsigned char Dio5GetDirection(void)
```

```
{
    unsigned char result;

    Set_NCT5523D_LD(0x07);                          //switch to logic device 7
    result = Get_NCT5523D_Reg(0xE8);
    return (result);
}
//---------------------------------------------------------------------------
```

```
File of the NCT5523D.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 "NCT5523D.H"
#include <dos.h>
//---------------------------------------------------------------------------
unsigned int NCT5523D_BASE;
void Unlock_NCT5523D (void);
void Lock_NCT5523D (void);
//---------------------------------------------------------------------------
unsigned int Init_NCT5523D(void)
{
    unsigned int result;
    unsigned char ucDid;

    NCT5523D_BASE = 0x4E;
    result = NCT5523D_BASE;

    ucDid = Get_NCT5523D_Reg(0x20);
    if (ucDid == 0xC4)                                  //NCT5523D??
{    goto Init_Finish;      }

    NCT5523D_BASE = 0x2E;
    result = NCT5523D_BASE;

    ucDid = Get_NCT5523D_Reg(0x20);
    if (ucDid == 0xC4)                                  //NCT5523D??
{   goto Init_Finish;   }

  NCT5523D_BASE = 0x00;
  result = NCT5523D_BASE;

Init_Finish:
      return (result);
```

```
}
//---------------------------------------------------------------------------
void Unlock_NCT5523D (void)
{
    outportb(NCT5523D_INDEX_PORT, NCT5523D_UNLOCK);
    outportb(NCT5523D_INDEX_PORT, NCT5523D_UNLOCK);
}
//---------------------------------------------------------------------------
void Lock_NCT5523D (void)
{
    outportb(NCT5523D_INDEX_PORT, NCT5523D_LOCK);
}
//---------------------------------------------------------------------------
```

```
void Set_NCT5523D_LD( unsigned char LD)
{
    Unlock_NCT5523D();
    outportb(NCT5523D_INDEX_PORT, NCT5523D_REG_LD);
    outportb(NCT5523D_DATA_PORT, LD);
    Lock_NCT5523D();
}
//----------------------------------------------------------------------------
void Set_NCT5523D_Reg( unsigned char REG, unsigned char DATA)
{
    Unlock_NCT5523D();
    outportb(NCT5523D_INDEX_PORT, REG);
    outportb(NCT5523D_DATA_PORT, DATA);
    Lock_NCT5523D();
}
//----------------------------------------------------------------------------
unsigned char Get_NCT5523D_Reg(unsigned char REG)
{
    unsigned char Result;
    Unlock_NCT5523D();
    outportb(NCT5523D_INDEX_PORT, REG);
    Result = inportb(NCT5523D_DATA_PORT);
    Lock_NCT5523D();
    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 NCT5523D.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 __NCT5523D_H
#define __NCT5523D_H                                    1
//-----------------------------------------------------------------------------
#define     NCT5523D_INDEX_PORT                         (NCT5523D_BASE)
#define     NCT5523D_DATA_PORT                          (NCT5523D_BASE+1)
//-----------------------------------------------------------------------------
#define     NCT5523D_REG_LD                             0x07
//-----------------------------------------------------------------------------
#define NCT5523D_UNLOCK                                 0x87
#define NCT5523D_LOCK                                   0xAA
//-----------------------------------------------------------------------------
unsigned int Init_NCT5523D(void);
void Set_NCT5523D_LD( unsigned char);
void Set_NCT5523D_Reg( unsigned char, unsigned char);
unsigned char Get_NCT5523D_Reg( unsigned char);
//-----------------------------------------------------------------------------
#endif      //__NCT5523D_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 "NCT5523D.H"
//---------------------------------------------------------------------------
int main (void);

void WDTInitial(void);
void WDTEnable(unsigned char);
void WDTDisable(void);

//---------------------------------------------------------------------------
int main (void)
{
  char SIO;

  SIO = Init_NCT5523D();
  if (SIO == 0)
  {
    printf("Can not detect Nuvoton NCT5523D, program abort.\n");
    return(1);
  }

WDTInitial();
WDTEnable(10);
WDTDisable();
return 0;
```

```
}
//---------------------------------------------------------------------------
void WDTInitial(void)
{
unsigned char bBuf;
Set_NCT5523D_LD(0x08);                          //switch to logic device 8
bBuf = Get_NCT5523D_Reg(0x30);
bBuf &= (~0x01);
Set_NCT5523D_Reg(0x30, bBuf);                   //Enable WDTO
}
//---------------------------------------------------------------------------


void WDTEnable(unsigned char NewInterval)
{
    unsigned char bBuf;

    Set_NCT5523D_LD(0x08);                      //switch to logic device 8
    Set_NCT5523D_Reg(0x30, 0x01);               //enable timer
    bBuf = Get_NCT5523D_Reg(0xF0);
    bBuf &= (~0x08);
    Set_NCT5523D_Reg(0xF0, bBuf);               //count mode is second

    Set_NCT5523D_Reg(0xF1, NewInterval);        //set timer
}
//---------------------------------------------------------------------------
void WDTDisable(void)
{
    Set_NCT5523D_LD(0x08);                      //switch to logic device 8
    Set_NCT5523D_Reg(0xF1, 0x00);               //clear watchdog timer
    Set_NCT5523D_Reg(0x30, 0x00);               //watchdog disabled
}
//---------------------------------------------------------------------------
```

```
File of the NCT5523D.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 "NCT5523D.H"
#include <dos.h>
//---------------------------------------------------------------------------
unsigned int NCT5523D_BASE;
void Unlock_NCT5523D (void);
void Lock_NCT5523D (void);
//---------------------------------------------------------------------------
unsigned int Init_NCT5523D(void)
{
   unsigned int result;
   unsigned char ucDid;

   NCT5523D_BASE = 0x4E;
   result = NCT5523D_BASE;
   ucDid = Get_NCT5523D_Reg(0x20);
   if (ucDid == 0xC4)                                    //NCT5523D??
   {     goto Init_Finish;     }

  NCT5523D_BASE = 0x2E;
  result = NCT5523D_BASE;
  ucDid = Get_NCT5523D_Reg(0x20);
  if (ucDid == 0xC4)                                     //NCT5523D??
{     goto Init_Finish;     }

NCT5523D_BASE = 0x00;
result = NCT5523D_BASE;

Init_Finish:
```

```
    return (result);
}
//---------------------------------------------------------------------------
void Unlock_NCT5523D (void)
{
    outportb(NCT5523D_INDEX_PORT, NCT5523D_UNLOCK);
    outportb(NCT5523D_INDEX_PORT, NCT5523D_UNLOCK);
}
//---------------------------------------------------------------------------
void Lock_NCT5523D (void)
{
    outportb(NCT5523D_INDEX_PORT, NCT5523D_LOCK);
}
//---------------------------------------------------------------------------
```

```
void Set_NCT5523D_LD( unsigned char LD)
{
   Unlock_NCT5523D();
   outportb(NCT5523D_INDEX_PORT, NCT5523D_REG_LD);
   outportb(NCT5523D_DATA_PORT, LD);
   Lock_NCT5523D();
}
//---------------------------------------------------------------------------
void Set_NCT5523D_Reg( unsigned char REG, unsigned char DATA)
{
    Unlock_NCT5523D();
    outportb(NCT5523D_INDEX_PORT, REG);
    outportb(NCT5523D_DATA_PORT, DATA);
    Lock_NCT5523D();
}
//---------------------------------------------------------------------------
unsigned char Get_NCT5523D_Reg(unsigned char REG)
{
    unsigned char Result;
    Unlock_NCT5523D();
    outportb(NCT5523D_INDEX_PORT, REG);
    Result = inportb(NCT5523D_DATA_PORT);
    Lock_NCT5523D();
    return Result;
}
//---------------------------------------------------------------------------
```

Optional Wi-Fi Module

IDOOH-210-IR offers optional Wi-FI module, please check with iBASE to get details information.