ROBOX300

Compact AMR designed controller

User's Manual

USER'S MANUAL

AXIOMTEK

www.axiomtek.com

Disclaimers

This manual has been carefully checked and believed to contain accurate information. Axiomtek Co., Ltd. assumes no responsibility for any infringements of patents or any third party's rights, or any liability arising from such uses.

Axiomtek does not warrant or assume any legal liability or responsibility for the accuracy, completeness or usefulness of any information in this document. Axiomtek does not make any commitment to update any information in this manual.

Axiomtek reserves the right to change or revise this document and/or product at any time without notice.

No part of this document may be reproduced, stored in a retrieval system, or transmitted in any forms or by any means, electronic, mechanical, photocopying, recording, among others, without prior written permissions of Axiomtek Co., Ltd.

Oct. 2024, Version A2

Printed in Taiwan

Safety Precautions

Before getting started, please read the following important safety precautions.

1. The ROBOX300 does not come with an operating system which must be loaded first before installation of any software into the computer.
2. Be sure to ground yourself to prevent static charge when installing any internal components. Use a wrist grounding strap and place all electronic components in any static-shielded devices. Most electronic components are sensitive to static electrical charge.
3. Disconnect the power cord from the ROBOX300 prior to making any installation. Be sure both the system and all external devices are turned OFF. Sudden surge of power could ruin sensitive components. Make sure the ROBOX300 is properly grounded.
4. Make sure the voltage of the power source is correct before connecting it to any power outlet.
5. Turn OFF the system power before cleaning. Clean the system using a cloth only. Do not spray any liquid cleaner directly onto the screen.
6. Do not leave equipment in an uncontrolled environment where the storage temperature is below -40℃ or above 80℃ as it may damage the equipment.
7. Do not open the system's back cover. If opening the cover for maintenance is a must, only a trained technician is allowed to do so. Integrated circuits on computer boards are sensitive to static electricity. To avoid damaging chips from electrostatic discharge, observe the following precautions:
   - Before handling a board or integrated circuit, touch an unpainted portion of the system unit chassis for a few seconds. This will help discharge any static electricity on the human body.
   - When handling boards and components, wear a wrist grounding strap available from most electronic component stores.
8. Follow these simple precautions to protect yourself from harm and the products from damage.
   - To avoid electrical shock, always disconnect the power from the PC chassis before manual handling. Do not touch any components on motherboard while the PC is powered on.
   - Disconnect the power before making any configuration changes. A sudden rush of power after connecting a jumper or installing a card may damage sensitive electronic components.

Battery Information

Batteries, battery packs, and accumulators must not be disposed of as unsorted household waste. Please use the public collection system to return, recycle, or treat such items in compliance with local regulations.

※ Cautions are included to help prevent hardware damage and data losses.

For example,

The battery is at risk of exploding if incorrectly installed. Do not attempt to recharge, force open, or heat the battery. Replace the battery only with the same or equivalent type recommended by the manufacturer. Discard used batteries according to the manufacturer's instructions."

※ As for the RTC battery used in the system, when the battery voltage drops below 2.5V, the REAL-TIME CLOCK (RTC) RAM in the CMOS will be cleared, including the date, time, system password, and system configuration parameters. At this point, the battery must be replaced, and all parameters need to be reconfigured. Please refer to Chapter 2.4 for instructions on how to replace the battery.

Classifications

1. Degree of production against electric shock: not classified
2. Degree of protection against ingress of water: IP40
3. Equipment not suitable for use in the presence of a flammable anesthetic mixture with air, oxygen or nitrous oxide.
4. Mode of operation: Continuous

General Cleaning Tips

Please keep the following precautions in mind while understanding the details fully before and during any cleaning of the computer and any components within.

A piece of dry cloth is ideal to clean the device.

1. Be cautious of any tiny removable components when using a vacuum cleaner to absorb dirt on the floor.
2. Turn the system off before cleaning up the computer or any components within.
3. Avoid dropping any components inside the computer or getting circuit board damp or wet.
4. For cleaning, be cautious of all kinds of cleaning solvents or chemicals which may cause allergy to certain individuals.
5. Keep foods, drinks or cigarettes away from the computer.

Cleaning Tools:

Although many companies have created products to help improve the process of cleaning computer and peripherals, users can also use household items accordingly for cleaning. Listed below are items available for cleaning computer or computer peripherals.

Pay special attention to components requiring designated products for cleaning as mentioned below.

- Cloth: A piece of cloth is the best tool to use when rubbing up a component. Although paper towels or tissues can be used on most hardware as well, it is recommended to use a piece of cloth.
- Water or rubbing alcohol: A piece of cloth may be somewhat moistened with water or rubbing alcohol before being rubbed on the computer. Unknown solvents may be harmful to plastic parts.
- Absorb dust, dirt, hair, cigarette and other particles outside of a computer can be one of the best methods of cleaning a computer. Over time these items may restrict the airflow in a computer and cause circuitry to corrode.
- Cotton swabs: Cotton swabs moistened with rubbing alcohol or water are applicable to reach areas in a keyboard, mouse and other areas.
- Foam swabs: If possible, it is better to use lint free swabs such as foam swabs.

.

【Note】: ***It is strongly recommended that customer should shut down the system before starting to clean any single components.***

Please follow the steps below:

1. Close all application programs;
2. Close operating software;
3. Turn off power switch;
4. Remove all devices;
5. Pull out power cable.

Scrap Computer Recycling

Please inform the nearest Axiomtek distributor as soon as possible for suitable solutions in case computers require maintenance or repair; or for recycling in case computers are out of order.

Trademarks Acknowledgments

Table of Contents

This page is intentionally left blank.

SECTION 1
INTRODUCTION

This section contains general information and detailed specifications of the ROBOX300. Section 1 consists of the following sub-sections:

- General Descriptions
- System Specifications
- Dimensions
- I/O Outlets
- Packing List
- Model List

1.1 General Descriptions

The ROBOX300 Series features 11th gen Intel® Core™ i5® Quad-core ULT processor (Tiger Lake UP3), high performance yet low power consumption, fan-less slim type design, -40°C to +60°C extended operating temperature, and 9V to 60V wide range DC power input with industrial-grade reliability. Highly integrated and with rich IO configuration, the ROBOX300 is perfectly suitable for Edge Computing, Machine vision, Embedded controller, Robotics applications.

Features

- 11th gen Intel® Core™ i5 Quad-core ULT processor (Tiger Lake UP3)
- Dual-channel DDR4-3200 SO-DIMM for up to 64GB of memory
- Supports 3x 2.5 GbE, 6 USB, 6 COM

- -40°C to +60°C wide operating temperatures
- 9 to 60 VDC wide range DC power input
- Supports dual displays with 2 HDMI
- Intel® Iris® Xe integrated graphics
- Trusted platform module (TPM 2.0 onboard)

Reliable and Stable Design
The embedded system supports 11th gen Intel® Core™ i5 Quad-core ULT processor, along with the features of high performance, industrial-grade operation temperature/power input and multi-functional design that make it the best solution factory automation, machine vision, Edge Computing, Robotics applications.

Rich IO Connectivity
The ROBOX300 comes with rich I/O interfaces including 6 RS-232/422/485 ports, 1 RS232 D-SUB console port and 2 CAN, 4 USB 3.2 ports, 2 USB 2.0 ports,3 2.5G high speed ethernet,2 HDMI 1.4b, one 8-CH TTL DIO, and one front access SIM slot socket supported.

Embedded O.S. Supported
The ROBOX300 with 11th generation platform supports Ubuntu 22.04 LTS

1.2 System Specifications

1.2.1 CPU

- **CPU**
  - Intel® Core™ i5-1145G7E
- **Chipset**
  - SoC
- **BIOS**
  - American Megatrends Inc. UEFI (Unified Extensible Firmware Interface BIOS.
- **System Memory**
  - 2 x 260-pin DDR4-3200 SO-DIMM, up to 64GB

1.2.2 I/O System

- **Display**
  - 2 x HDMI 1.4b (3840 x 2160@30 Hz)
- **Ethernet**
  - 3 x 2.5GbE (3 x Intel® I226-IT)
- **USB Ports**
  - 4 x USB 3.2 Gen 1

- 2 x USB 2.0

- **Serial Ports**
  - 6 x RS-232/422/485 (COM2/COM3/COM4/COM5/COM6/COM7), Baud rate max. up to 115200
  - 1 x RS-232 (COM1 console DB9), Baud rate max. up to 115200
  - 2 x CAN 2.0b

- **DIO Port**
  - 1 x 8-CH TTL DIO (Default 4 in & 4 out)

- **Mini PCIe Interface**
  - 1 x Full-size PCI Express Mini Card (USB + PCIe signal)
  - 1 x M.2 Key E 2230 (for Wi-Fi)
  - 1 x M.2 Key M 2280 (for NVMe storage , PCIex4 signal only)

- **Storage**
  - 1 x 2.5" SATA HDD/SSD drive bay, up to 9.5mm in height

- **Indicator**
  - 1 x Green LED as indicator for PWR status
  - 1 x Amber LED as indicator for HDD/SSD active

- **Switch**
  - 1 xPWR switch
  - 1 x Reset button

- **Antenna & SIM**
  - 6 x SMA type connector openings for antenna
  - 1 x front access SIM slot + 1 internal SIM slot

- **TPM 2.0**
  - 1 x ST33HTPH2X32AHD8

1.2.3 System Specifications

- **Watchdog Timer**
  - 1-255 seconds or minutes; up to 255 levels.

- **Power Supply**
  - 9V-60V DC input

- **Operation Temperature**

  Ethernet version :
  - -40°C to +60°C (-40°F to +140°F) with 0.5 m/s air flow

    (with W.T. DRAM & SSD,CPU TDP 15W)

- **Storage Temperature**
  - -40℃ to +80℃ (-40 ºF to +176ºF)

- **Humidity**
  - 10% to 95% (non-condensation)

- **Shock**
  - IEC 60068-2-27 (w/SSD: 50G, half sine,11 ms duration)

- **Vibration Endurance**
  - IEC 60068-2-64 (w/SSD: 3Grms STD, random, 5 - 500 Hz,1 hr/axis)

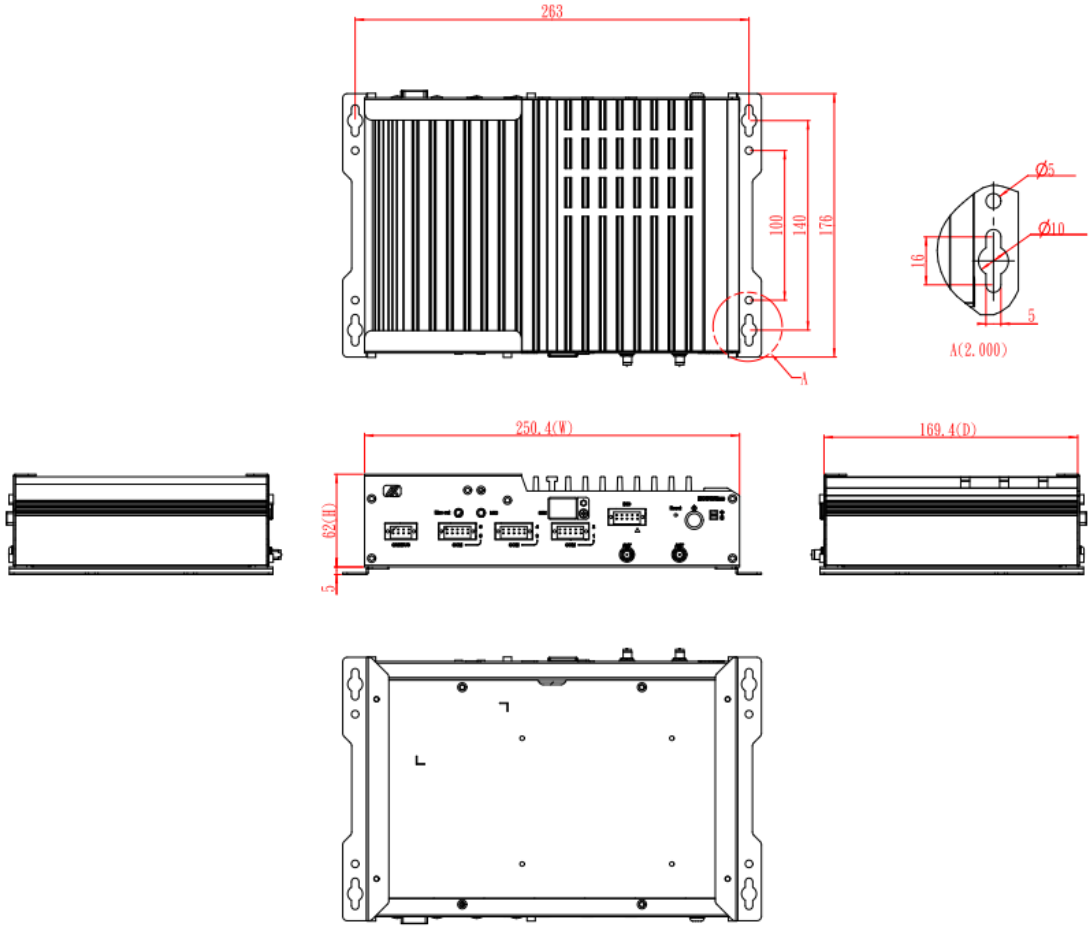
- **Weight**
  - 2.5 kg (5.51 lb) without package
  - 3.3 kg (7.27 lb) with package

- **Dimension**
  - 250mm (W) x 170mm (D) x 62mm (H)

1.3 Dimensions

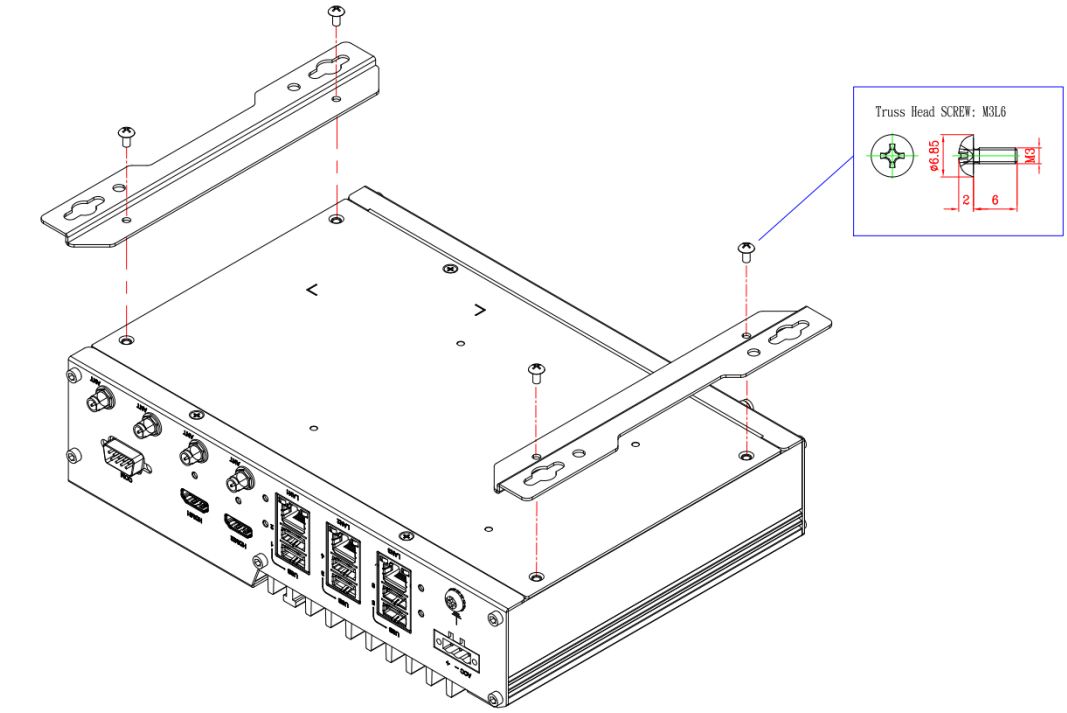
The following diagrams show dimensions and outlines of the ROBOX300.

1.3.1 ROBOX300 Dimensions

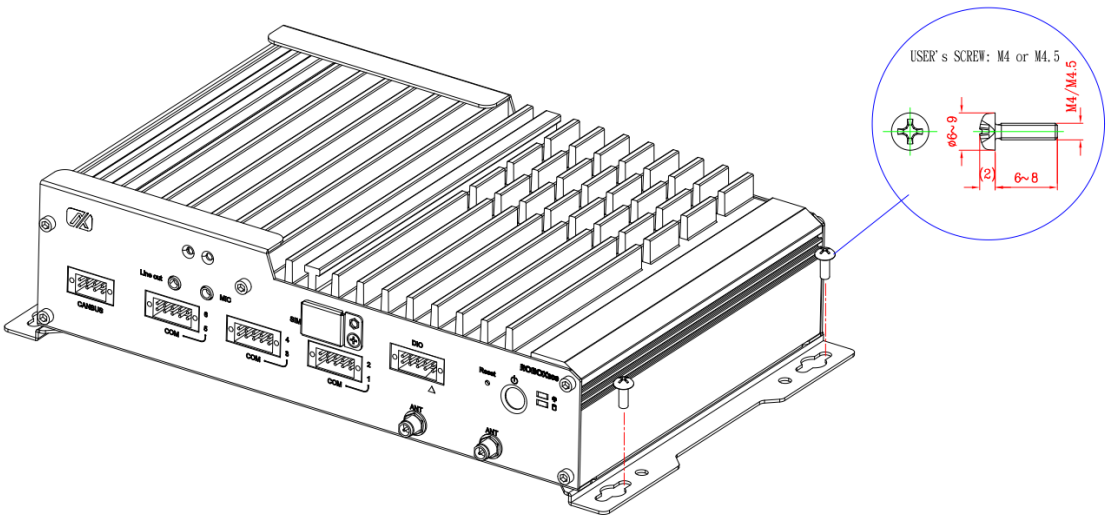


Wall-mount Bracket Assembly

From the accessories box, users can get 4pcs of truss head M3L6 screws for fixing the wall mount kit.



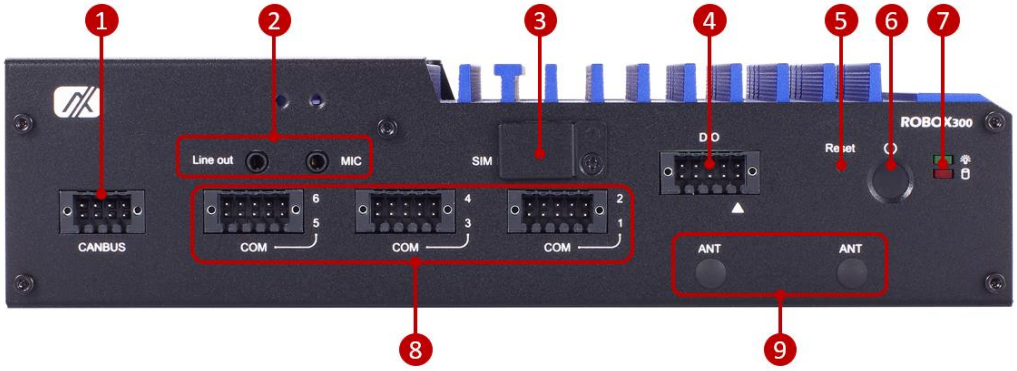
Note : If users install the screws in drywall, use the hollow wall anchors to ensure that the unit does not pull away from the wall due to prolonged strain between the cable and the power connector.



1.4 I/O Outlets

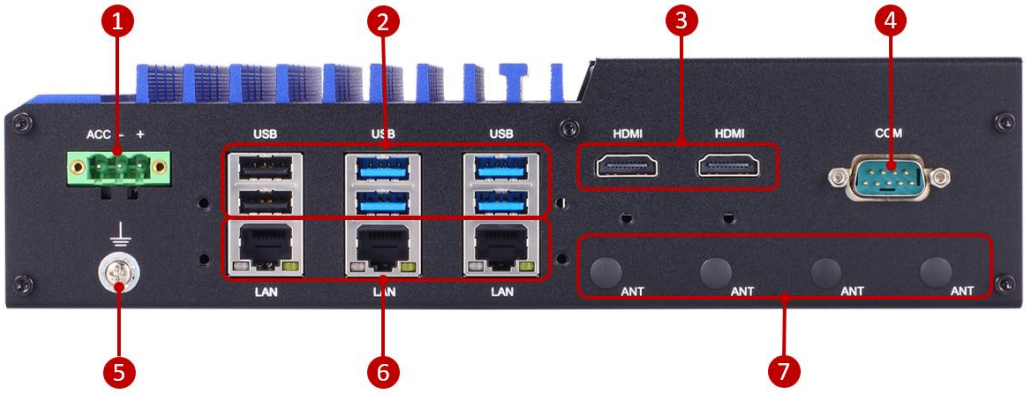
The following figures show I/O outlets on the ROBOX300.

Front View



1	**2 x CAN A/B**	**7**	**LEDs**
2	**Line out, Mic in**	**8**	**6 x COM (Isolated RS-485/422/232)**
3	**SIM card slot**	**9**	**2 x Antenna opening**
4	**8-bit GPIO**		
5	**1 x Reset button**		
6	**1 x Power button**		

Rear View



1	1 x Phoenix type power input	7	4 x Antenna opening
2	4 x USB 3.2 Gen 1 2 x USB 2.0		
3	2 x HDMI		
4	RS-232 (Console port)		
5	Ground Screw		
6	3 x LAN		

1.5 Packing List

The ROBOX300 comes with the following bundle package:

- **ROBOX300 system unit x 1**
- **DRAM Thermal Pad x 3**
- **DRAM Bracket x 1**
- **3-pin Terminal block connector x 1**
- **2x5-pin Terminal block connector x 4**
- **2x4-pin Terminal block connector x 1**
- **Foot Pad x 4**
- **Screw pack x 1**
- **M.2 Bracket x 1**
- **M.2 Thermal pad x 3**

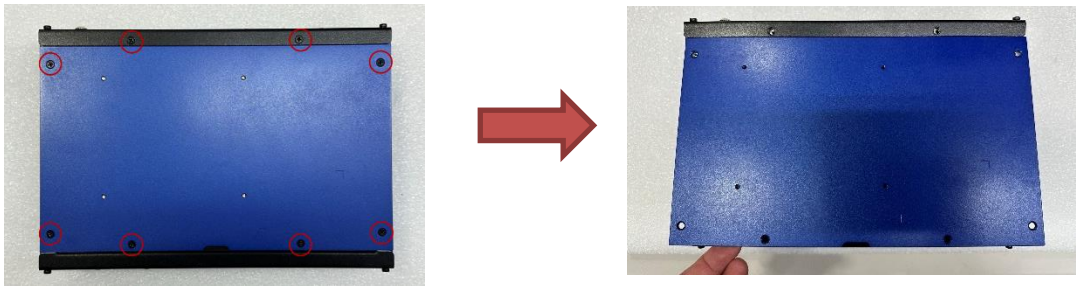
This page is intentionally left blank.

SECTION 2
HARDWARE INSTALLATION

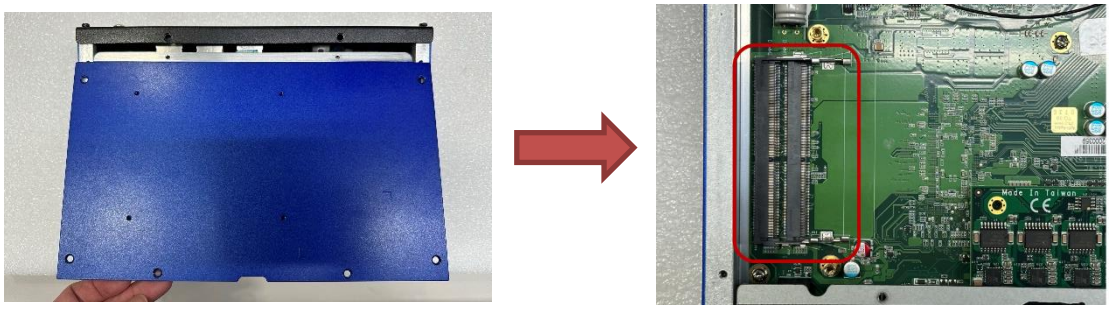
The ROBOX300 is convenient for various hardware configurations, such as CPU, DRAM, HDD (Hard Disk Drive), SSD (Solid State Drive), PCI Express Mini card modules. Section 2 contains guidelines for hardware installation.

2.1 Installation of SO-DIMM

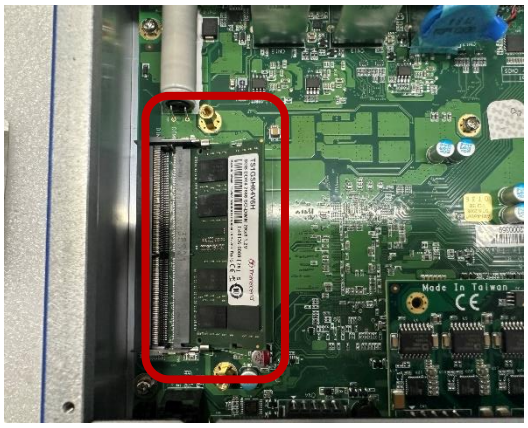
Step 1 Turn off the system and unplug the power cord.
Step 2 Turn the system unit upside down and loosen 8 screws on the bottom cover of the chassis and pull up the bottom cover.

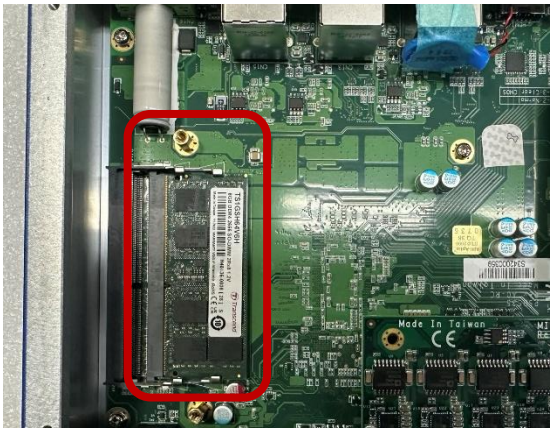


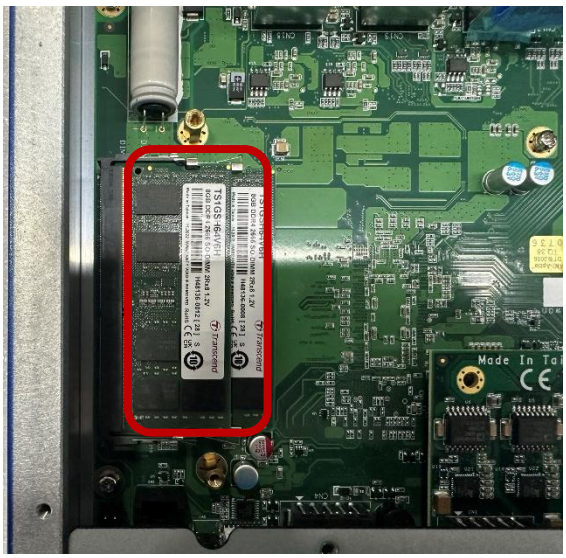
Step 3 Pull and open the bottom cover back, then located the dual DDR4 SO-DIMM sockets on main board as red marked.



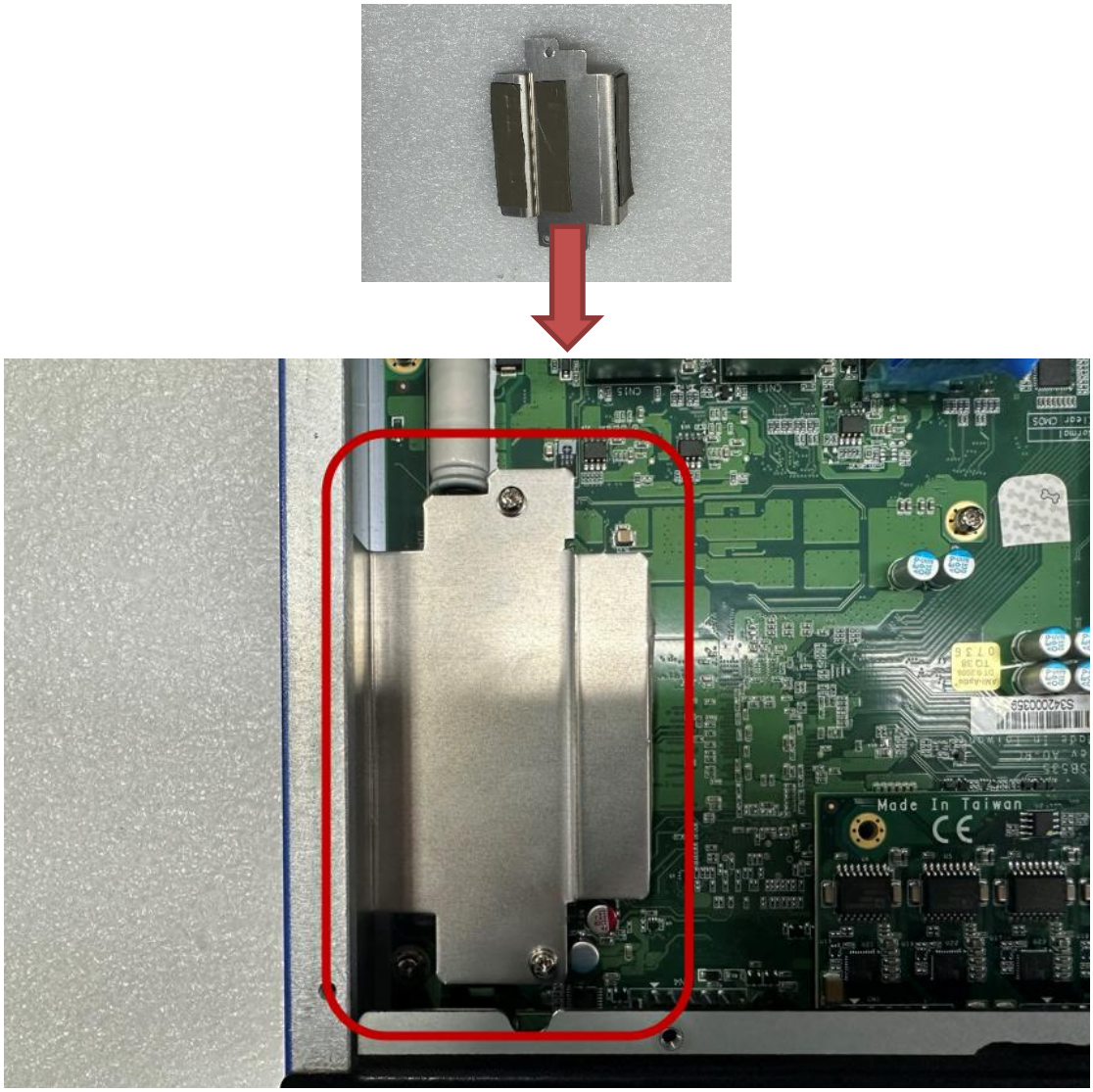
Step 4 Locate the memory module, place the thermal pad on the bottom of the DRAM socket, then insert a gold colored contact into the socket and push the module down until it is locked in place by the two end latches. And then place another thermal pad on top of the DRAM module.


Made In Taiwan


Made In Taiwan



Step 5 Place a thermal pad on the DRAM bracket and screw it on top of the DRAM.

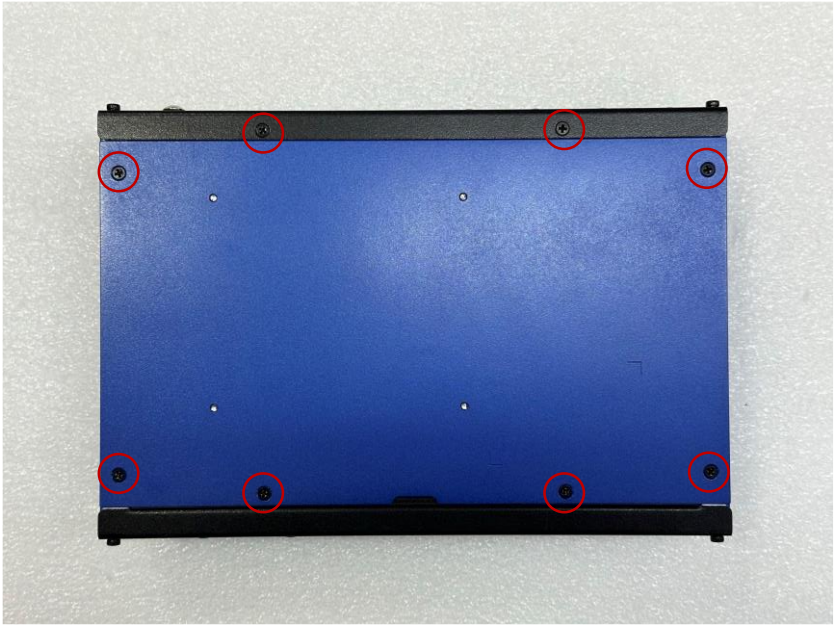


Step 6 Put the bottom cover back and fasten two screws back onto the system.

2.2 Installation of Mini PCIe Module (Full-Size)

Step 1 **Turn off the system and unplug the power cord.**

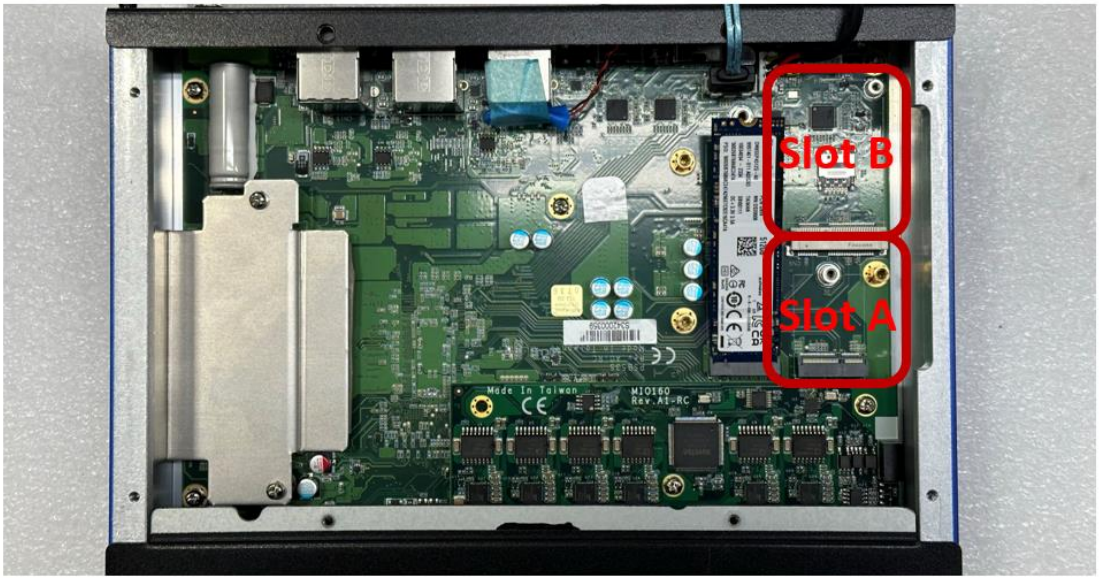
Step 2 **Turn the system upside down to locate screws at the bottom and loosen 8 screws as red marked.**



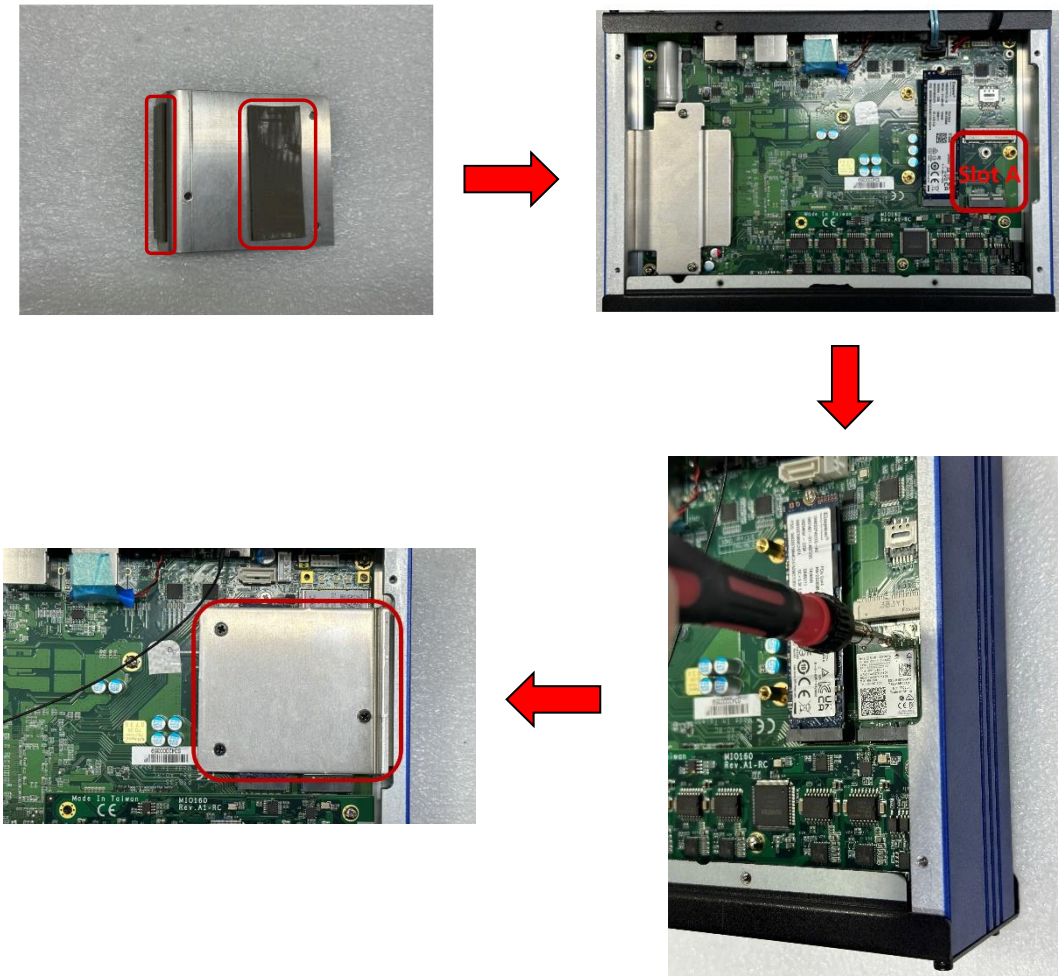
Step 3 **The ROBOX300 has two mini card slots:**

Slot A : M.2 Key E for Wi-Fi (CN7) (USB + PCIe signal)

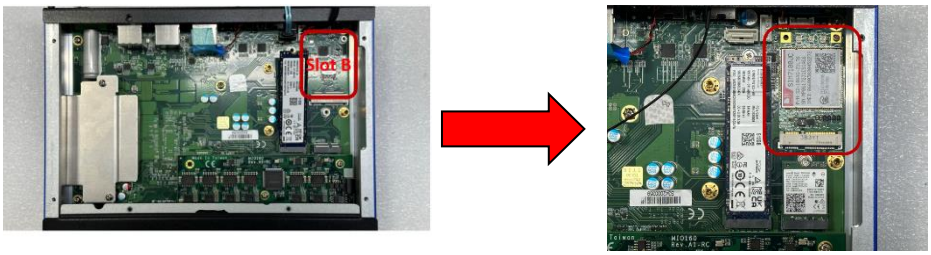
Slot B : Mini PCIe slot for Wi-Fi or LTE (CN8) (PCIex1 signal)



Step 4 **Slot A is used as the M.2 wireless mini card slot, assembly the pad on the bracket and insert the M.2 wireless mini PCIe module into the slot and fasten the screw, then secure the thermal pad bracket on top of the mini card module.**


Slot A

Step 5 Slot B is used as the wireless mini card slot, insert the Wi-Fi or LTE mini PCIe module into the slot and fasten the screw.



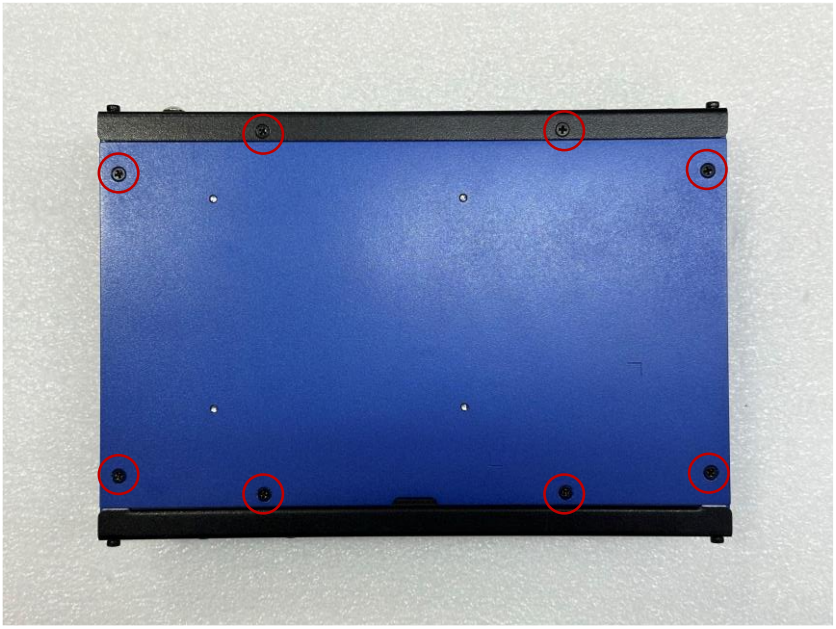
Step 6 And then connect the cable to antenna opening.

Step 7 Put the bottom cover and fasten all screws back onto the system.

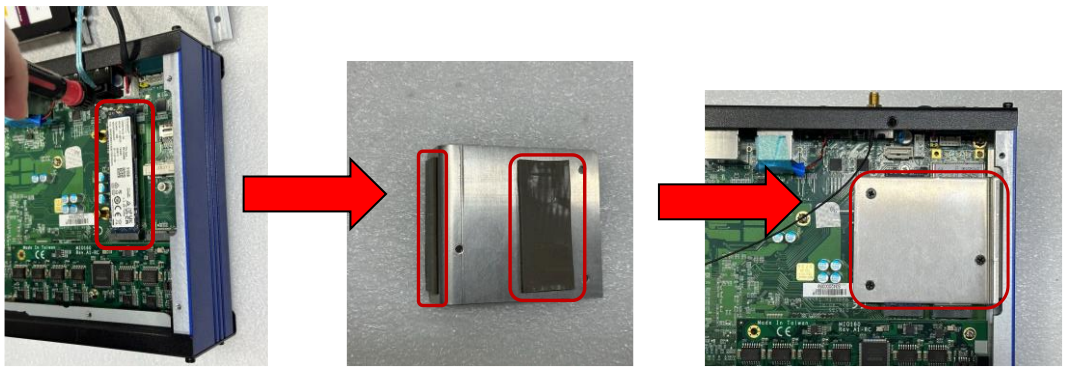
2.3 Installation of NVMe storage (M.2 Key M)

Step 1 **Turn off the system and unplug the power cord.**

Step 2 **Turn the system upside down to locate screws at the bottom and loosen 8 screws as red marked.**

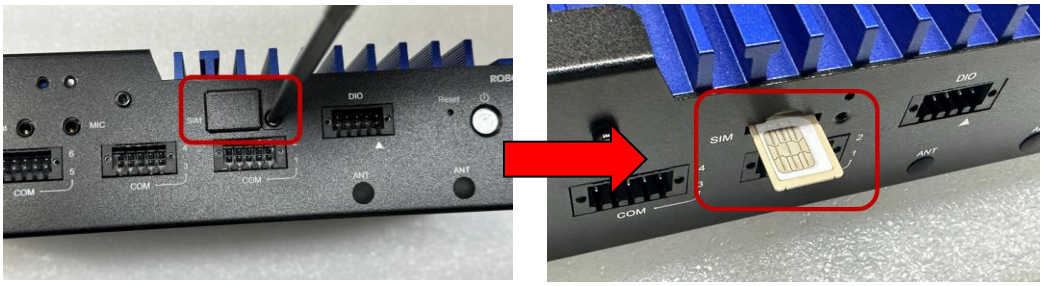


Step 3 **Located the M.2 NVMe slot as red marked, assembly the pad on the bracket and insert the M.2 NVMe module into the slot and fasten the screw, then secure the thermal pad bracket on top of the NVMe module.**

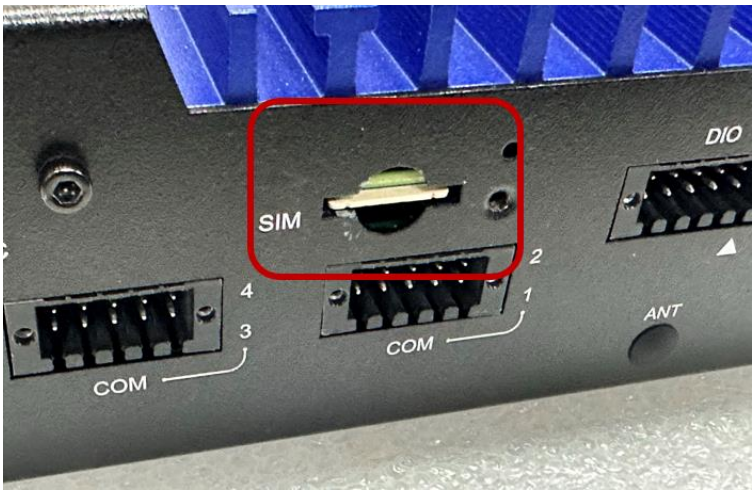


2.4 Installation of SIM Card

Step 1 Turn off the system and unplug the power cord.
Step 2 Loosen the screw of the SIM slot (Mini SIM) cover on chassis.



Step 3 Make sure the SIM card direction is correct and insert the SIM card firmly.

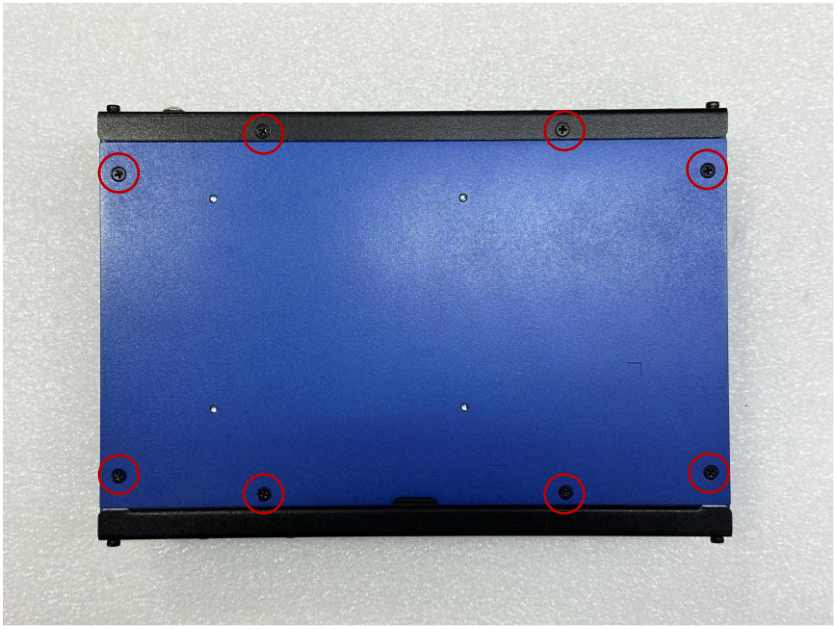


SIM card direction (contact side facing upward)

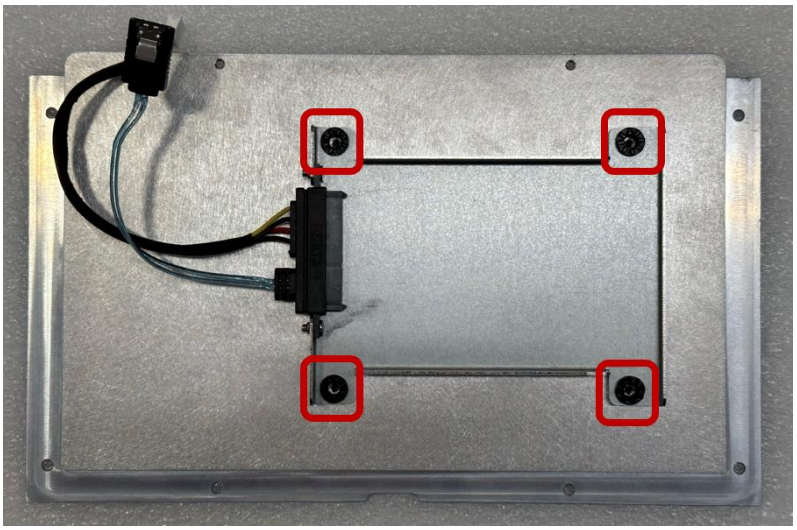
2.5 Installation of SATA SSD

Step 1 Turn off the system and unplug the power cord.

Step 2 Turn the system upside down to locate screws at the bottom and loosen 8 screws as red marked.

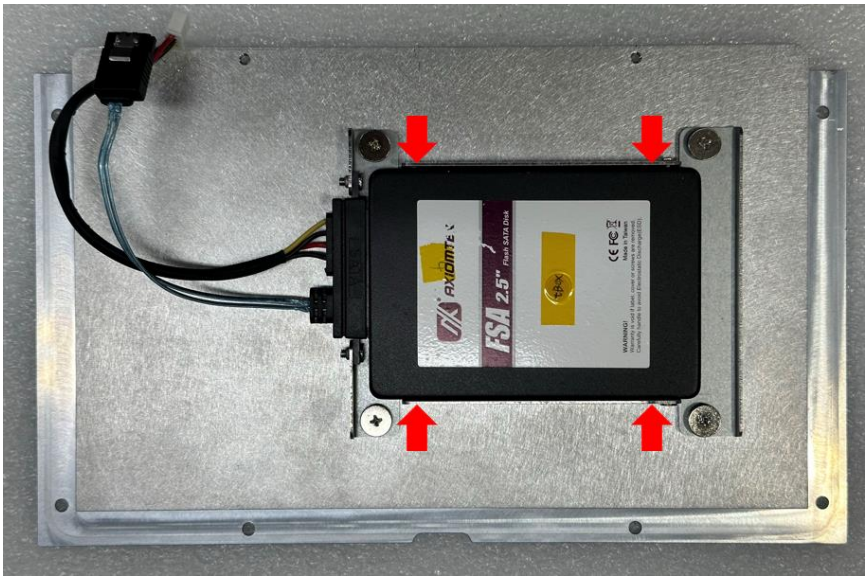


Step 3 Put the SATA SSD tray on the back of top cover, fasten the 4 screws of the tray.



Step 4 Put the SATA SSD and insert the SSD to the SATA connector.

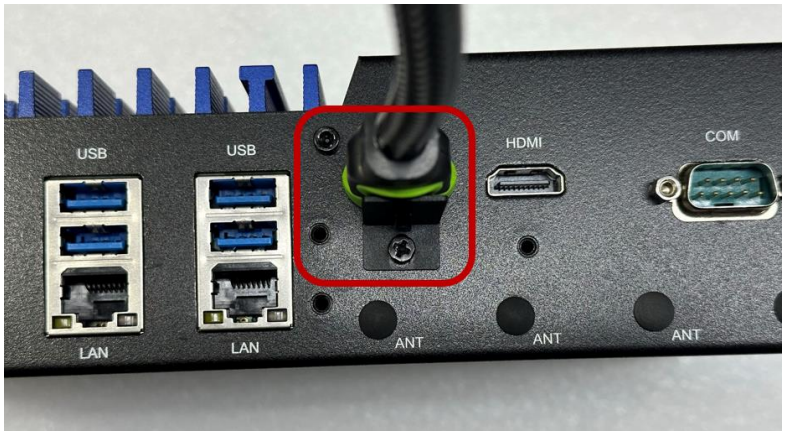
Step 5 Fasten the 4 side screws of SATA SSD to fix it.



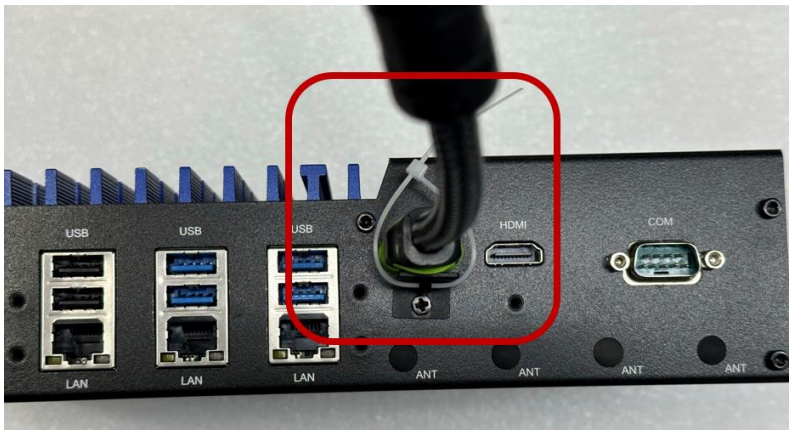
2.6 Installation of Cable Bracket

2.6.1 HDMI Cable Bracket

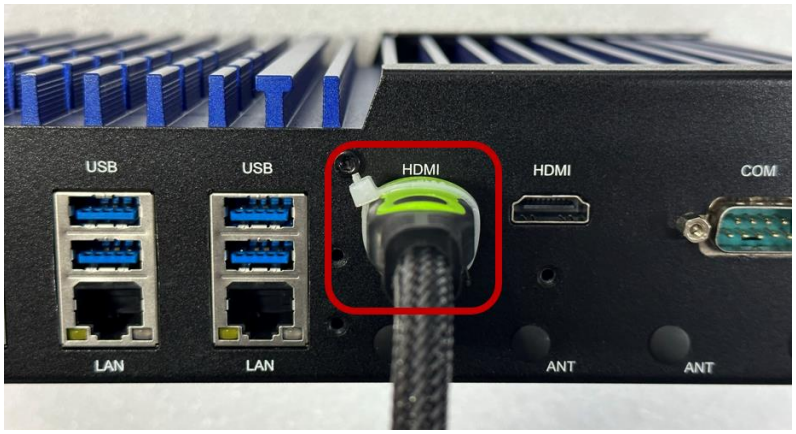
Step 1 Put the HDMI cable bracket below the HDMI connector and fasten the screw of it.



Step 2 Thread the cable tie through the hoop and tighten it.

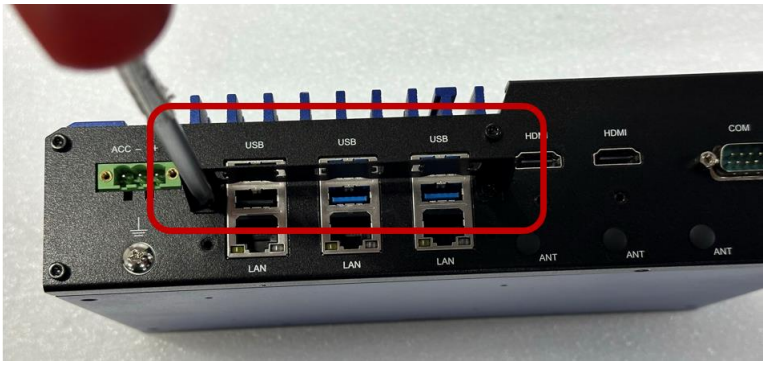


Step 3 After confirming that the cable tie is tight, cut off its end.

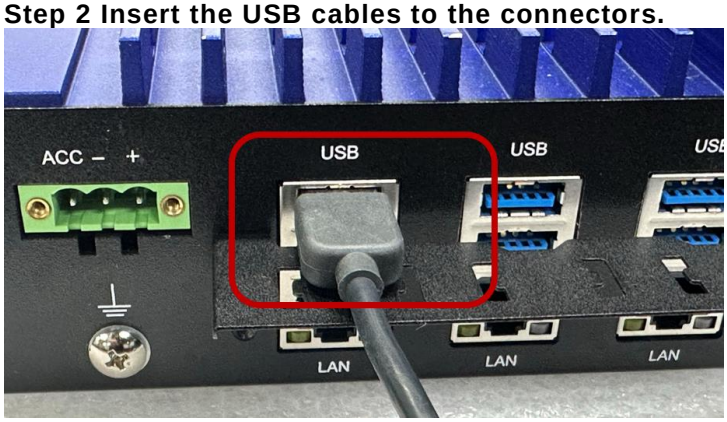


2.6.2 USB Cable Bracket

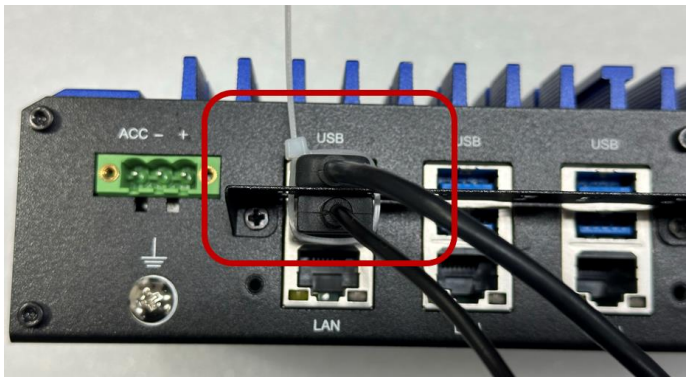
Step 1 Put the USB cable bracket below the USB connectors and fasten the screw of it.



Step 2 Insert the USB cables to the connectors.

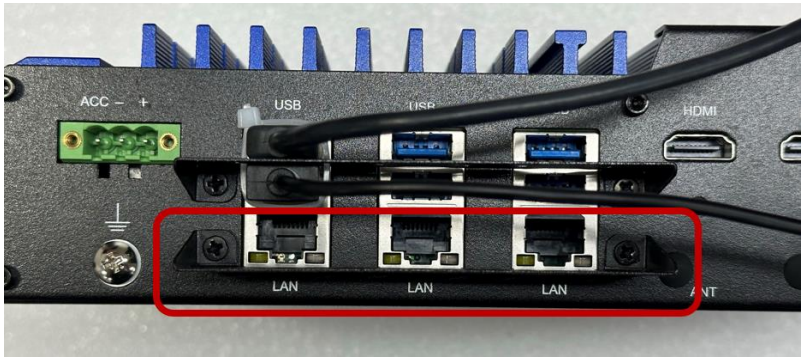


Step 3 Thread the cable tie through the hoop and tighten it.

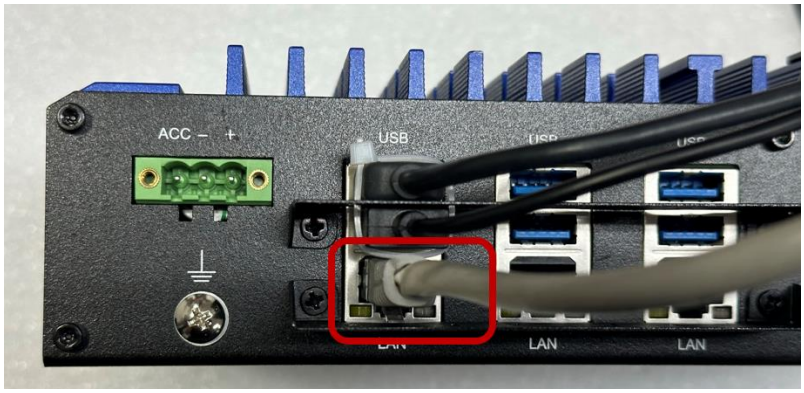


2.6.3 LAN Cable Bracket

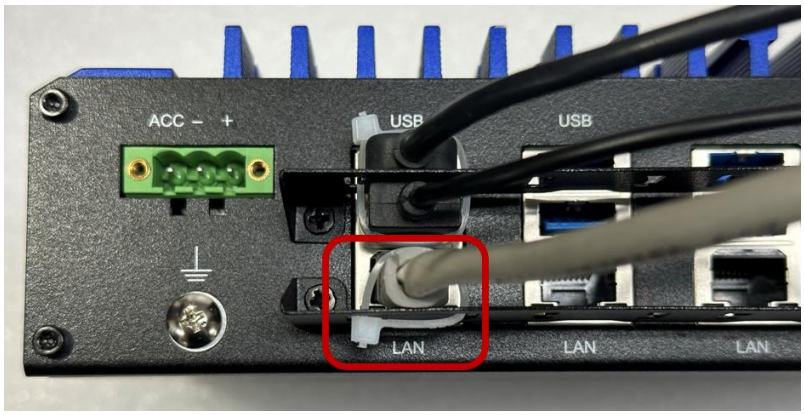
Step 1 Put the LAN cable bracket below the RJ-45 connectors and fasten the screw of it.



Step 2 Insert the LAN cable to the connector.

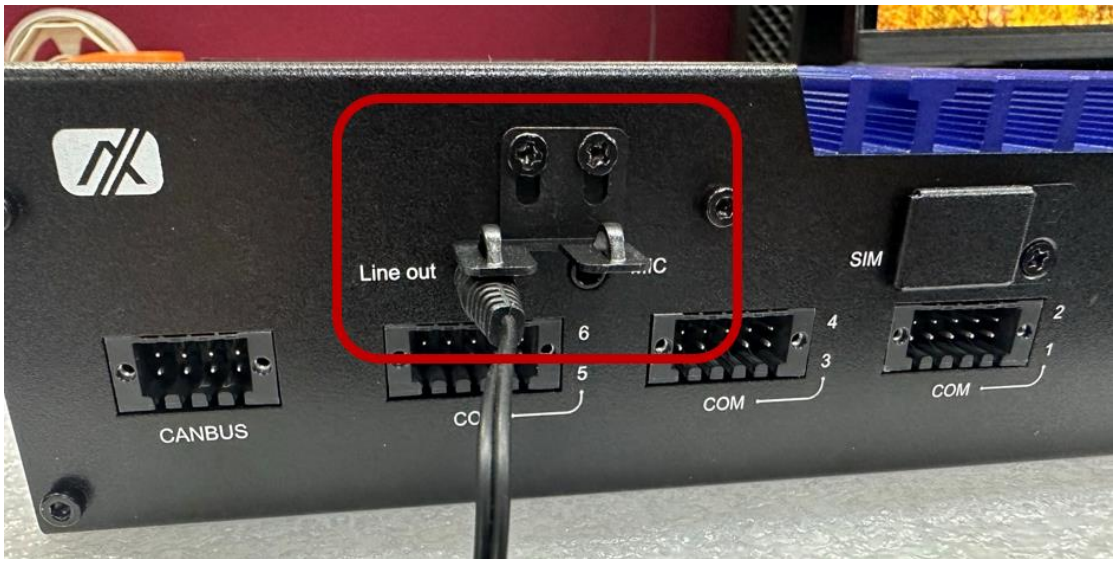


Step 3 Thread the cable tie through the hoop and tighten it.



2.6.4 Audio Cables Bracket

Step 1 Put the audio cable bracket up of the 3.5mm connectors and fasten the screw of it.



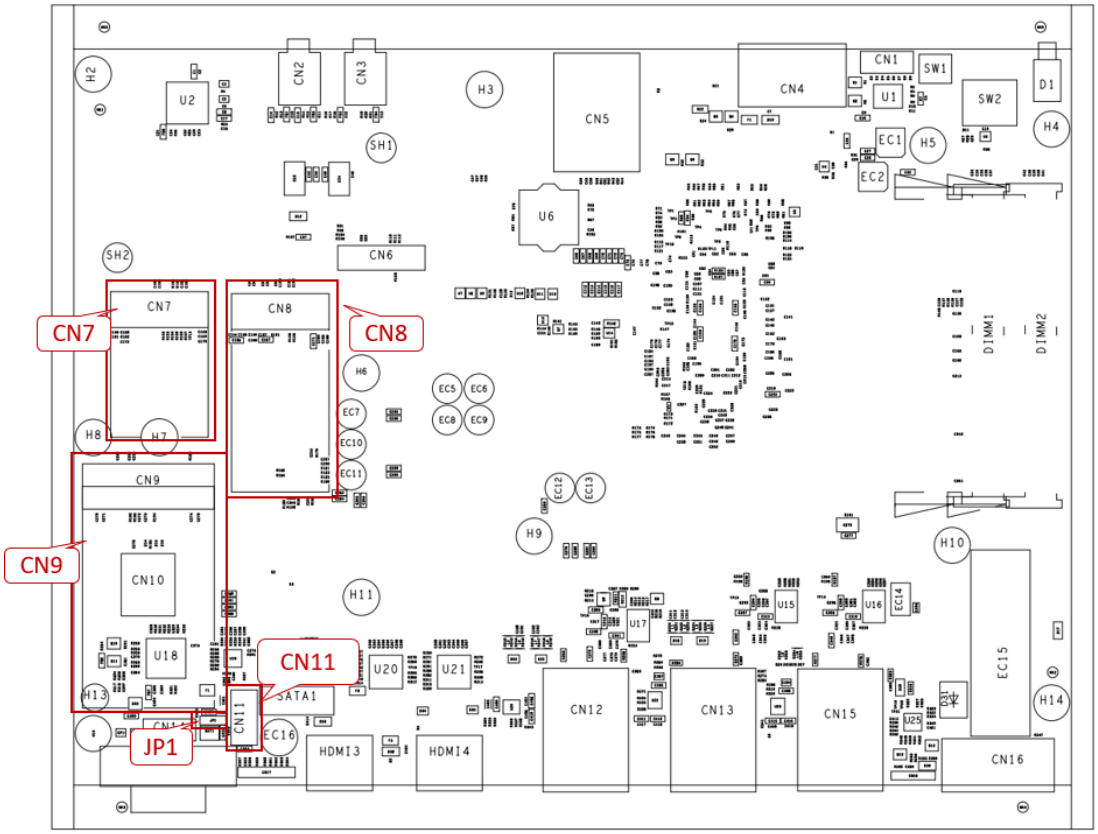
Step 2 Insert the Line out/ Mic in cables and thread the cable tie through the hoop and tighten it.

SECTION 3
CONNECTORS

Proper jumper settings configure the ROBOX300 to meet various application needs.-Hereby all jumpers settings along with their default settings are listed for devices onboard.

3.1 Locations of Connectors

PSB535 Top View

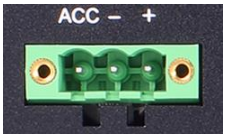


【Note】: ***It is strongly recommended that any unmentioned jumper settings should not be modified without instructions by Axiomtek FAEs. Any modifications without instructions might cause system failure.***

3.1.1 DC-in Phoenix Power Connector

The system supports 9-60V Phoenix DC-in connector for system power input. Typically, It's for 12V, 24V and 48V battery.

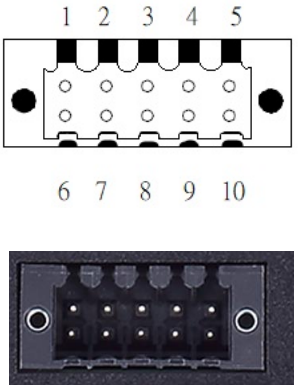
Pins	Signals
1	DC+
2	DC-
3	ACC



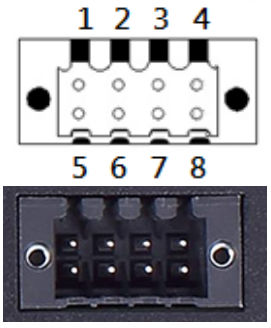
3.1.2 Isolated RS232/422/485 Serial Port and CANbus Connector

- 6 ports terminal block support RS-232/422/485 which can be selected by BIOS with isolation 2KV protection.
- Supports Auto Flow Control in RS485 mode.

CN1-CN3 (COM1 ~ COM6)			
Pins	**RS-232**	**RS-422**	**RS-485**
1	GND1, Ground	GND1, Ground	GND1, Ground
2	RTS1, Request To Send	RX1-	No use
3	TXD1, Transmit Data	RX1+	No use
4	CTS1, Clear To Send	TX1-	D1-
5	RXD1, Receive Data	TX1+	D1+
6	GND2, Ground	GND2, Ground	GND2, Ground
7	RTS2, Request To Send	RX2-	No use
8	TXD2, Transmit Data	RX2+	No use
9	CTS2, Clear To Send	TX2-	D2-
10	RXD2, Receive Data	TX2+	D2+

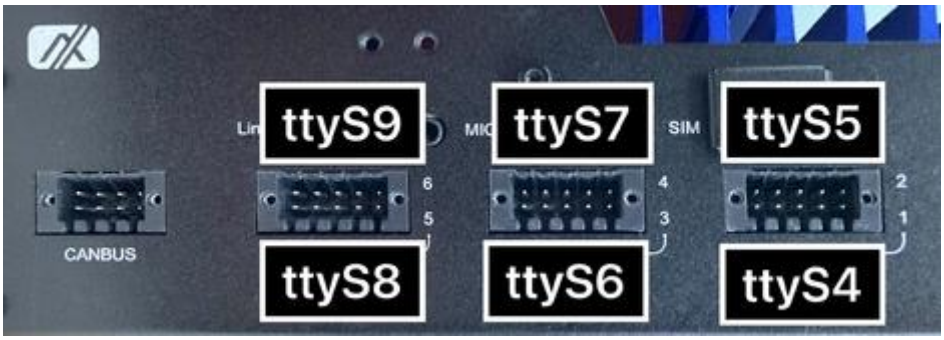


CAN1 & CAN2	
Pins	**CAN**
1	VCC1
2	CAN1-H
3	CAN1-L
4	GND1, Ground
5	VCC2
6	CAN2-H
7	CAN2-L



8	GND2, Ground

The serial port mapping in Linux system as below:

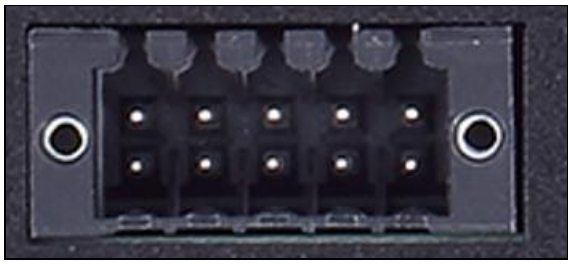


3.1.3 8-CH Digital IO (CN2)

ROBOX300 supports one 8-CH GPIO connector (DIO1~8) , default : 4IN & 4OUT. Each bit can be set to function as input or output by software programming, and users can set up via BIOS setting.

Pins	Description
1	DIO1
2	DIO2
3	DIO3
4	DIO4
5	DIO5
6	DIO6
7	DIO7
8	DIO8
9	5V
10	GND

1 2 3 4 5



6 7 8 9 10

3.1.4 Reset Button

The Reset button can allow users to reset the ROBOX300 system.

Functions	Descriptions
On	Reset system
Off	Keep system status

3.1.5 Audio Connector

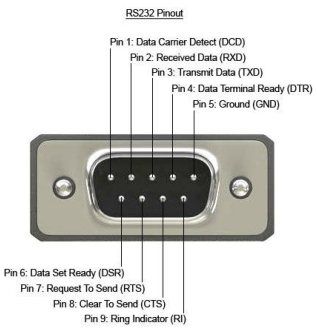
The Audio jacks ideal for Audio Mic-In and Audio Line-out.

Pins	Signals
1	Line Out
2	Microphone In

3.1.6 RS-232 Serial Port Connector

The ROBOX300 supports 1x RS-232 port. Please refer to Chapter 4 for the details of BIOS settings.

Pins	RS-232
1	DCD, Data Carrier Detect
2	RXD, Receive Data
3	TXD, Transmit Data
4	DTR, Data Terminal Ready
5	GND, Ground
6	DSR, Data Set Ready
7	RTS, Request To Send
8	CTS, Clear To Send
9	RI, Ring Indicator



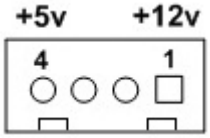
3.1.7 SATA Power Connector (CN11)

The SATA power for SATA 2.5" HDD/SSD.

*Connector specification: wafer 4P, P=2.5mm

*5V & 12V Max. power rating is 0.5A.

Pins	Signals
1	+12V level
2	GND
3	GND
4	+5V level

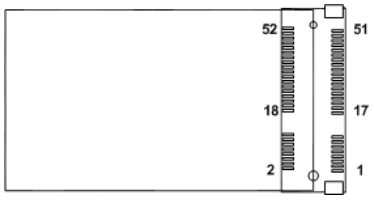


3.1.8 Full-Size PCI Express Mini Card Slot (CN9)

The ROBOX300 supports one full-size PCI-Express Mini Card slots.CN8 is applying to either PCI-Express or USB 2.0 signal and complies with PCI-Express Mini Card Spec. V1.2.

CN8

Pins	Signals	Pins	Signals
1	WAKE#	2	+3.3VSB
3	No use	4	GND
5	No use	6	+1.5V
7	CLKREQ#	8	SIM_PWR
9	GND	10	SIM_DATA
11	REFCLK-	12	SIM_CLK
13	REFCLK+	14	SIM_REST
15	GND	16	SIM_VPP
17	No use	18	GND
19	No use	20	W_DISABLE#
21	GND	22	PERST#
23	PE_RXN3/	24	+3.3VSB
25	PE_RXP3/	26	GND
27	GND	28	+1.5V
29	GND	30	SMB_CLK
31	PE_TXN3/	32	SMB_DATA
33	PE_TXP3/	34	GND
35	GND	36	USB_D8-
37	GND	38	USB_D8+
39	+3.3VSB	40	GND
41	+3.3VSB	42	No use
43	GND	44	No use
45	No use	46	No use
47	No use	48	+1.5V
49	No use	50	GND
51	No use	52	+3.3VSB



3.1.9 M.2 2280 Key M (CN8)

The M.2 2280 Key M for NVMe storage.

Pins	Signals	Pins	Signals	Pins	Signals	Pins	Signals
1	GND	2	+3.3V	3	GND	4	+3.3V
5	PEX3_RX-	6	NC	7	PEX3_RX+	8	NC
9	GND	10	LED_1#	11	PEX3_TX-	12	+3.3V
13	PEX3_TX+	14	+3.3V	15	GND	16	+3.3V
17	PEX2_RX-	18	+3.3V	19	PEX2_RX+	20	NC
21	GND	22	NC	23	PEX2_TX-	24	NC
25	PEX2_TX+	26	NC	27	GND	28	NC
29	PEX1_RX-	30	NC	31	PEX1_RX+	32	NC
33	GND	34	NC	35	PEX1_TX-	36	NC
37	PEX1_TX+	38	DEV SLP	39	GND	40	NC
41	PEX0_RX-	42	NC	43	PEX0_RX+	44	NC
45	GND	46	NC	47	PEX0_TX-	48	NC
49	PEX0_TX+	50	PERST#	51	GND	52	CLKREQ#
53	PEX0_REFCLkn	54	PEWAKE#	55	PEX0_REFCLkp	56	NC
57	GND	58	NC	59	CONNECTOR KEY M	60	CONNECTOR KEY M
61	CONNECTOR KEY M	62	CONNECTOR KEY M	63	CONNECTOR KEY M	64	CONNECTOR KEY M
65	CONNECTOR KEY M	66	CONNECTOR KEY M	67	NC	68	NC
69	NC	70	+3.3V	71	GND	72	+3.3V
73	GND	74	+3.3V	75	GND		

3.1.10 M.2 2230 Key E (CN7)

The M.2 2230 Key E for Wi-Fi.

1	GND	2	+3.3V	3	USB_D+	4	+3.3V
5	USB_D-	6	NC	7	GND	8	NC
9	NC	10	NC	11	NC	12	NC
13	GND	14	NC	15	NC	16	NC
17	NC	18	GND	19	GND	20	NC
21	NC	22	NC	23	NC	24	CONNECTOR Key E
25	CONNECTOR Key E	26	CONNECTOR Key E	27	CONNECTOR Key E	28	CONNECTOR Key E
29	CONNECTOR Key E	30	CONNECTOR Key E	31	CONNECTOR Key E	32	NC
33	GND	34	NC	35	PETp0	36	NC
37	PETn0	38	NC	39	GND	40	NC
41	PERp0	42	NC	43	PERn0	44	NC
45	GND	46	NC	47	REFCLKp0	48	NC
49	REFCLKn0	50	SUSCLK	51	GND	52	PERST0#
53	CLKREQ#	54	MCP_BT_DISABLE	55	PEWAKE0#	56	MCP_WIFI_DISABLE
57	GND	58	M.2_E_SMB_DATA	59	PETp1	60	M.2_E_SMB_CLK
61	PETn1	62	NC	63	GND	64	M.2_E_REFCLK
65	PERp1	66	NC	67	PERn1	68	NC
69	GND	70	NC	71	REFCLKp1	72	+3.3V
73	REFCLKn1	74	+3.3V	75	GND		

SECTION 4
BIOS SETUP UTILITY

This section provides users with detailed descriptions in terms of how to set up basic system configurations through the BIOS setup utility.

4.1 Starting

To enter the setup screens, follow the steps below:

1. Turn on the computer and press the <Del> key immediately.
2. After pressing the <Del> key, the main BIOS setup menu displays. Users can access to other setup screens, such as the Advanced and Chipset menus, from the main BIOS setup menu.

It is strongly recommended that users should avoid changing the chipset's defaults. Both AMI and system manufacturer have carefully set up these defaults that provide the best performance and reliability.

4.2 Navigation Keys

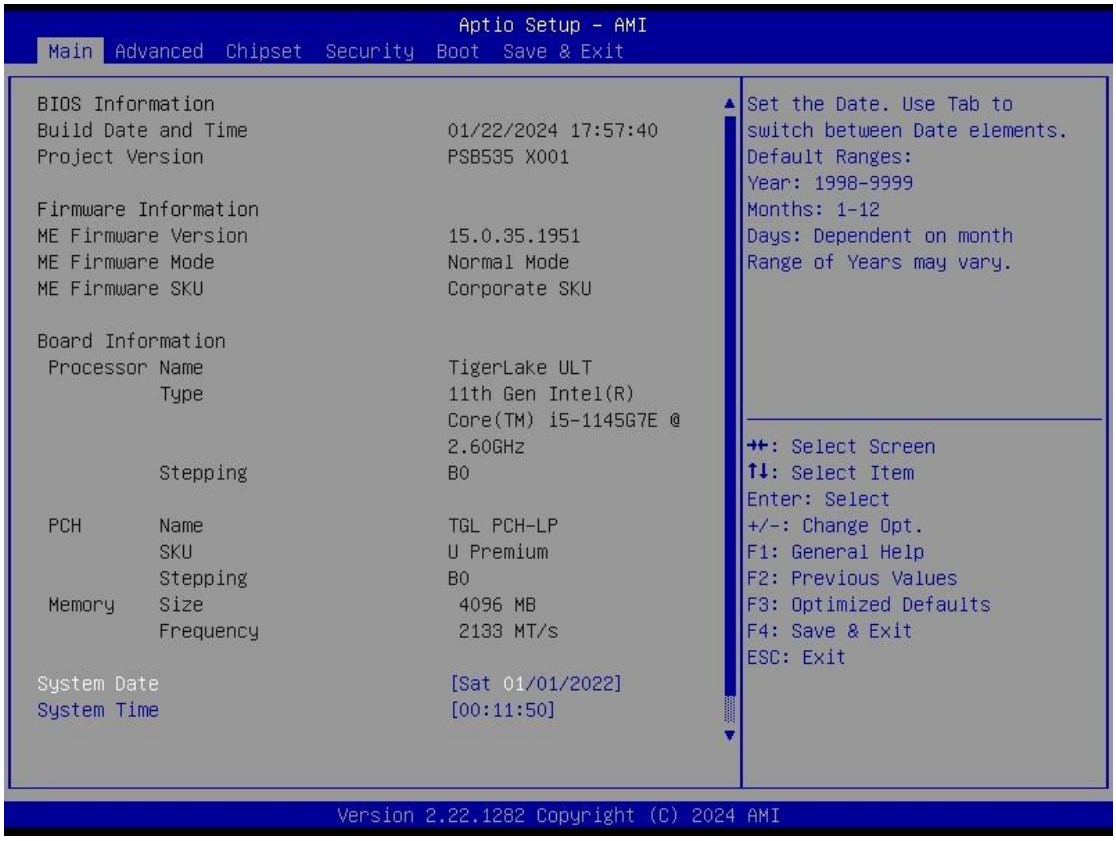
The BIOS setup/utility uses a key-based navigation system called hot keys. Most of the BIOS setup utility hot keys can be used at any time during the setup navigation process. These keys include <F1>, <F2>, <Enter>, <ESC>, <Arrow> keys, and so on.

【Note】: ***Some of the navigation keys differ from one screen to another.***

Hot Keys	Descriptions
→← **Left/Right**	The Left and Right <Arrow> keys allow users to select a setup screen.
↑↓ **Up/Down**	The Up and Down <Arrow> keys allow users to select a setup screen or sub-screen.
+– Plus/Minus	The Plus and Minus <Arrow> keys allow users to change the field value of a particular setup item.
Tab	The <Tab> key allows users to select setup fields.
F1	The <F1> key allows users to display the General Help screen.
F2	The <F2> key allows users to Load Previous Values.
F3	The <F3> key allows users to Load Optimized Defaults.
F4	The <F4> key allows users to save any changes they made and exit the Setup. Press the <F4> key to save any changes.
Esc	The <Esc> key allows users to discard any changes they made and exit the Setup. Press the <Esc> key to exit the setup without saving any changes.
Enter	The <Enter> key allows users to display or change the setup option listed for a particular setup item. The <Enter> key can also allow users to display the setup sub- screens.

4.3 Main Menu

The Main Menu screen is the first screen users see when entering the setup utility. Users can always return to the Main setup screen by selecting the Main tab. System Time/Date can be set up as described below. The Main BIOS setup screen is also shown below.



BIOS Information
Display the auto-detected BIOS information.

System Date/Time
Use this option to change the system time and date. Highlight System Time or System Date using the <Arrow> keys. Enter new values through the keyboard. Press the <Tab> key or the <Arrow> keys to move between fields. The date must be entered in MM/DD/YY format. The time is entered in HH:MM:SS format.

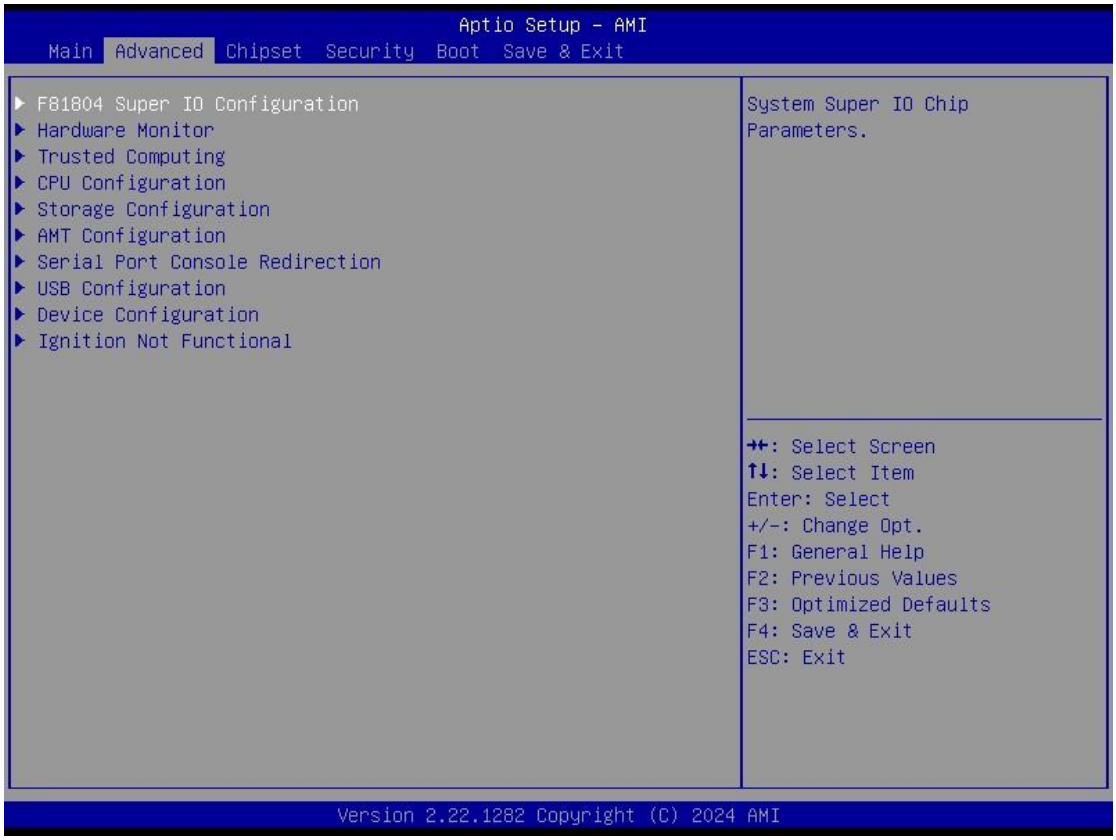
Access Level
Display the access level of current user.

4.4 Advanced Menu

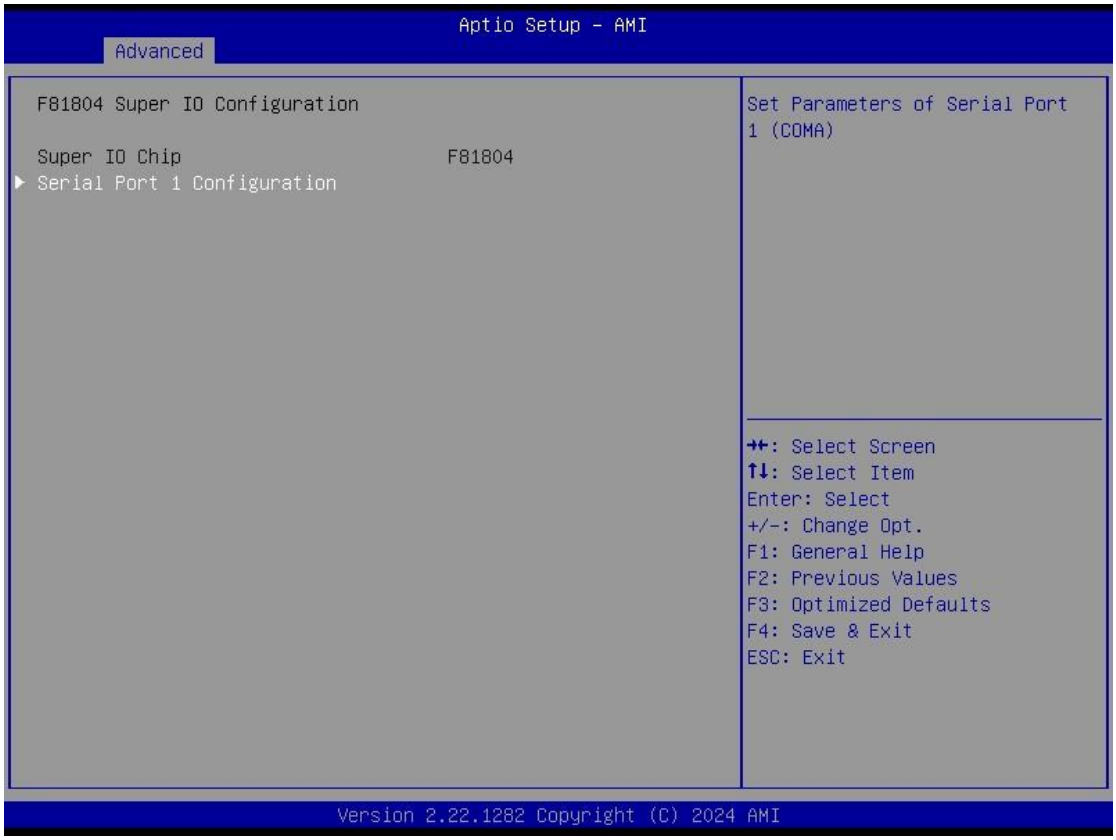
The Advanced menu also allows users to set configuration of the CPU and other system devices. Users can select any items in the left frame of the screen to go to sub menus:

- ► F81804 Super IO Configuration
- ► Hardware Monitor
- ► Trusted Computing
- ► CPU Configurations
- ► Storage Configuration
- ► AMT Configuration
- ► Serial Port Console Redirection
- ► USB Configuration
- ► Device Configuration
- ► Ignition

For items marked with "►", please press <Enter> for more options.



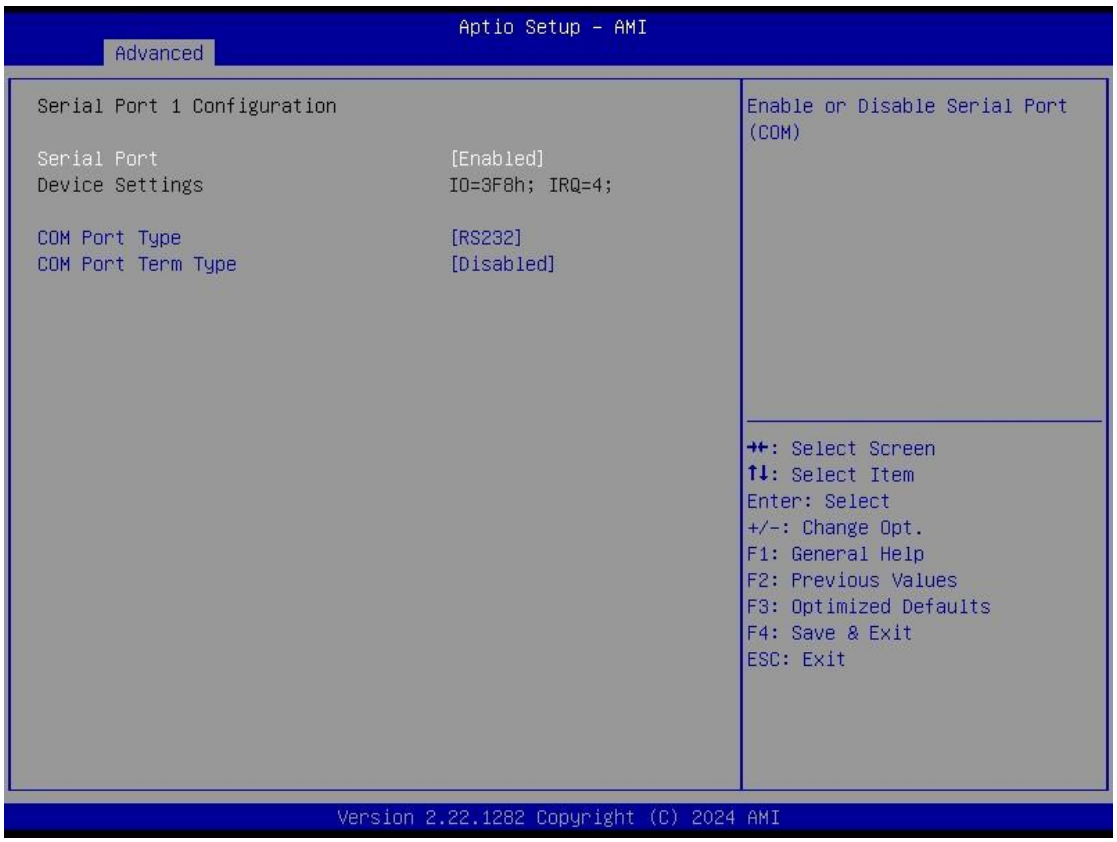
F81804 Super IO Configuration
Use this screen to select options for the F81966 Super IO Configurations and change the value of the selected option. A description of the selected item appears on the right side of the screen. For items marked with "►", please press <Enter> for more options.



Serial Port 1 Configurations
Use these items to set parameters related to serial ports COM1 (RS232 only)

Serial Port 1

Use this to set parameters of COM 1.

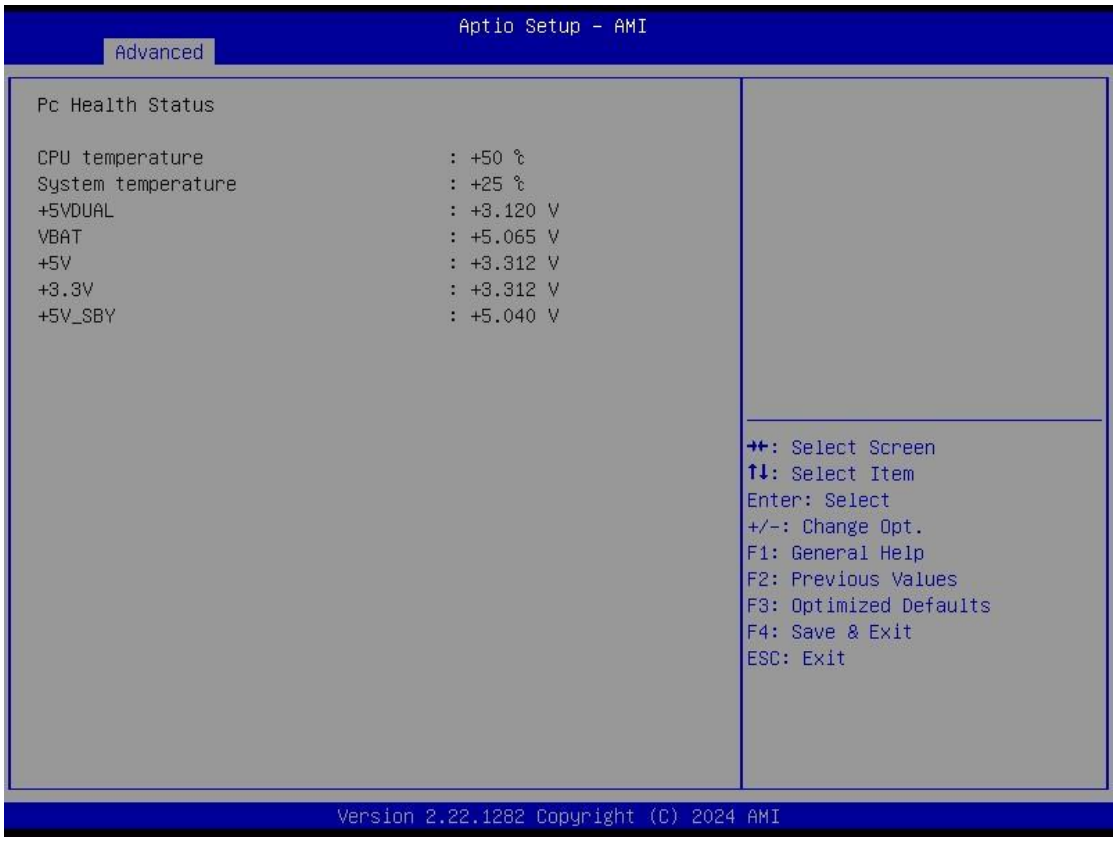


COM Port type

Use this item to set parameters related to serial ports COM 1 (RS232 only)

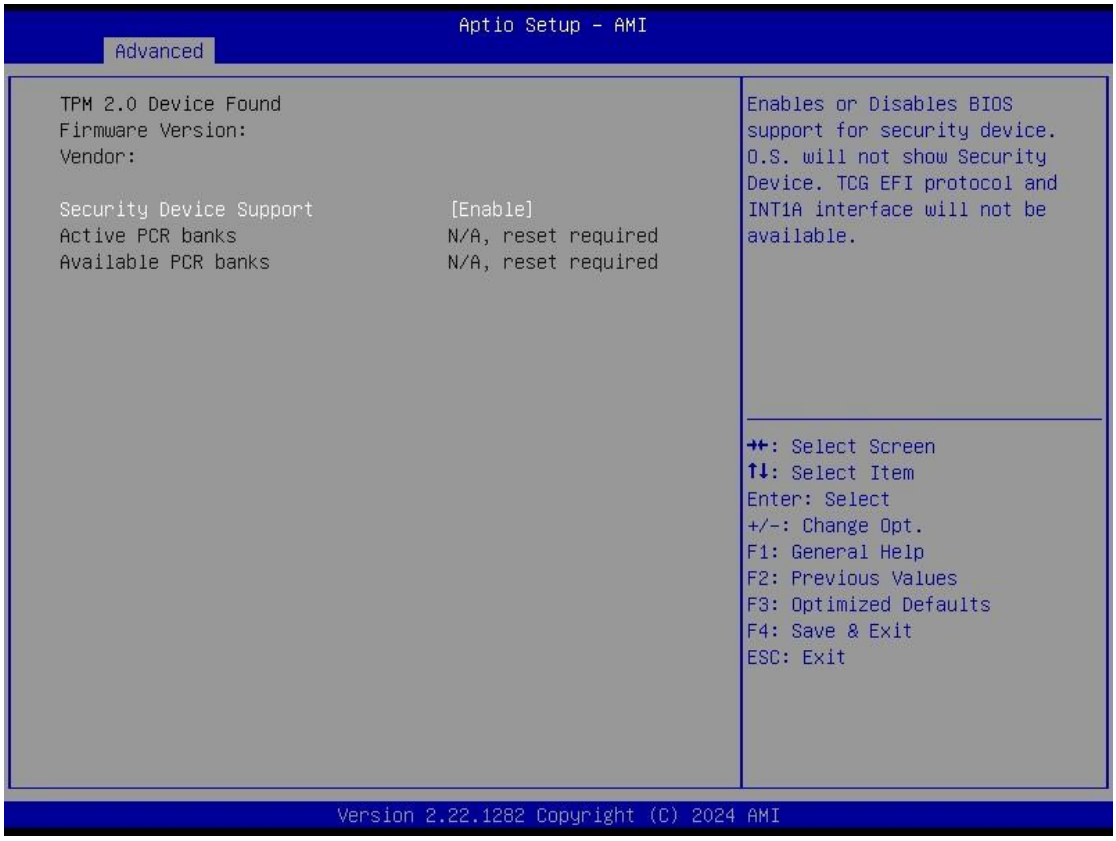
Hardware Monitor

This screen displays the temperature of system and CPU and system voltages (VCORE, +3.3V, +12V and +5V).



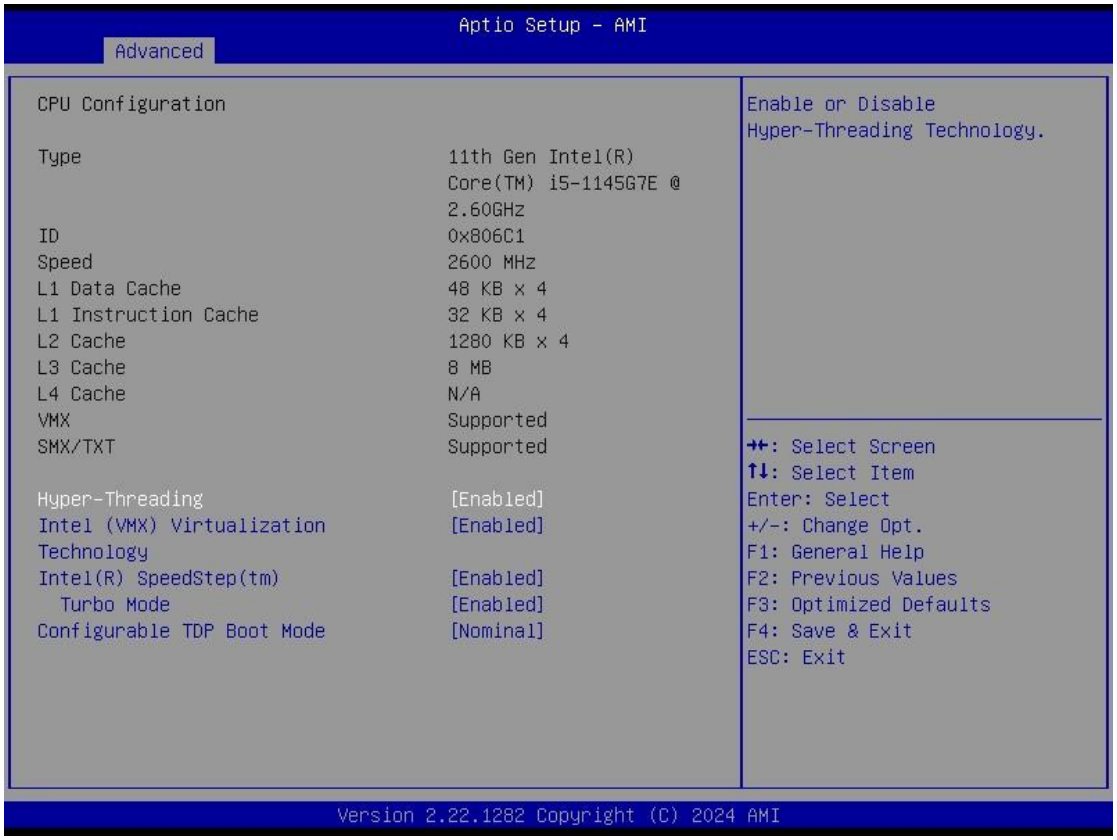
Trusted Computing

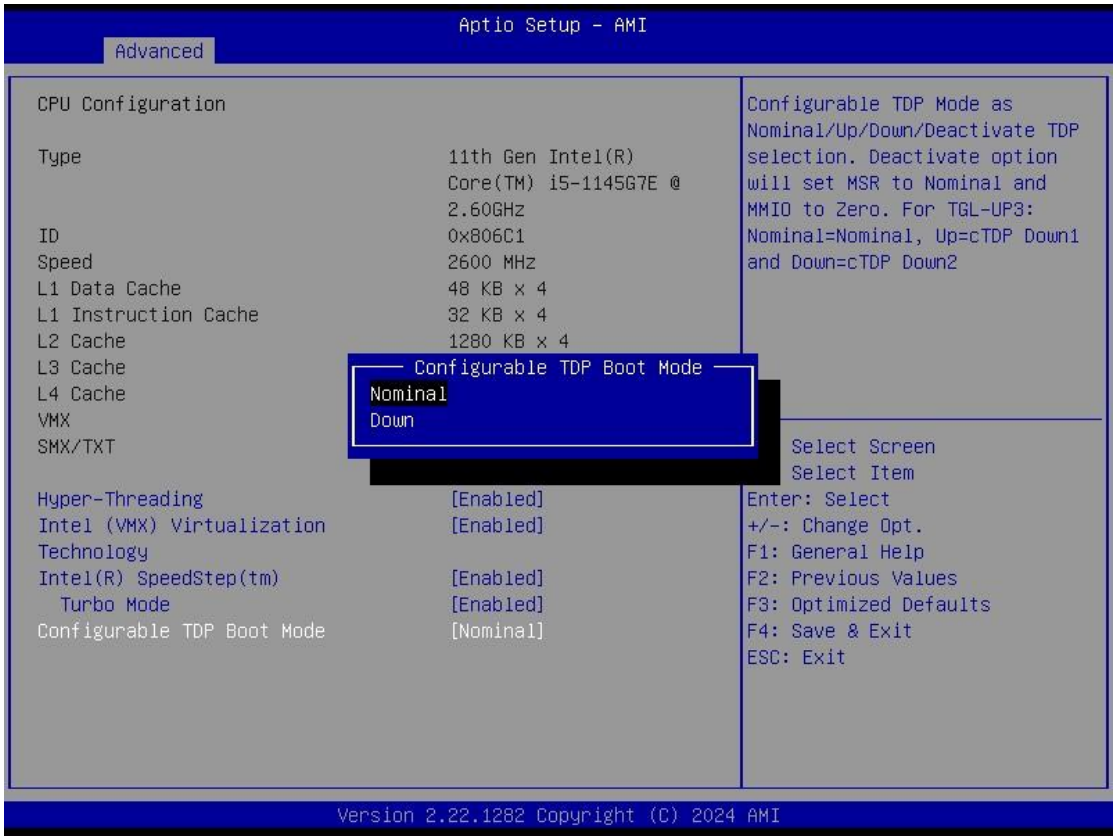
If users install a security device, such as TPM, users will see the following information for the TPM device and status.



CPU Configuration

This screen shows the CPU version and its detailed information.





Hyper-Threading
Use this item to enable or disable Hyper-Threading Technology, which makes a single physical processor perform multi-tasking functions as two logical ones.

Intel® I Virtualization Technology
It allows a hardware platform to run multiple operating systems separately and simultaneously, enabling one system to virtually function as several systems.

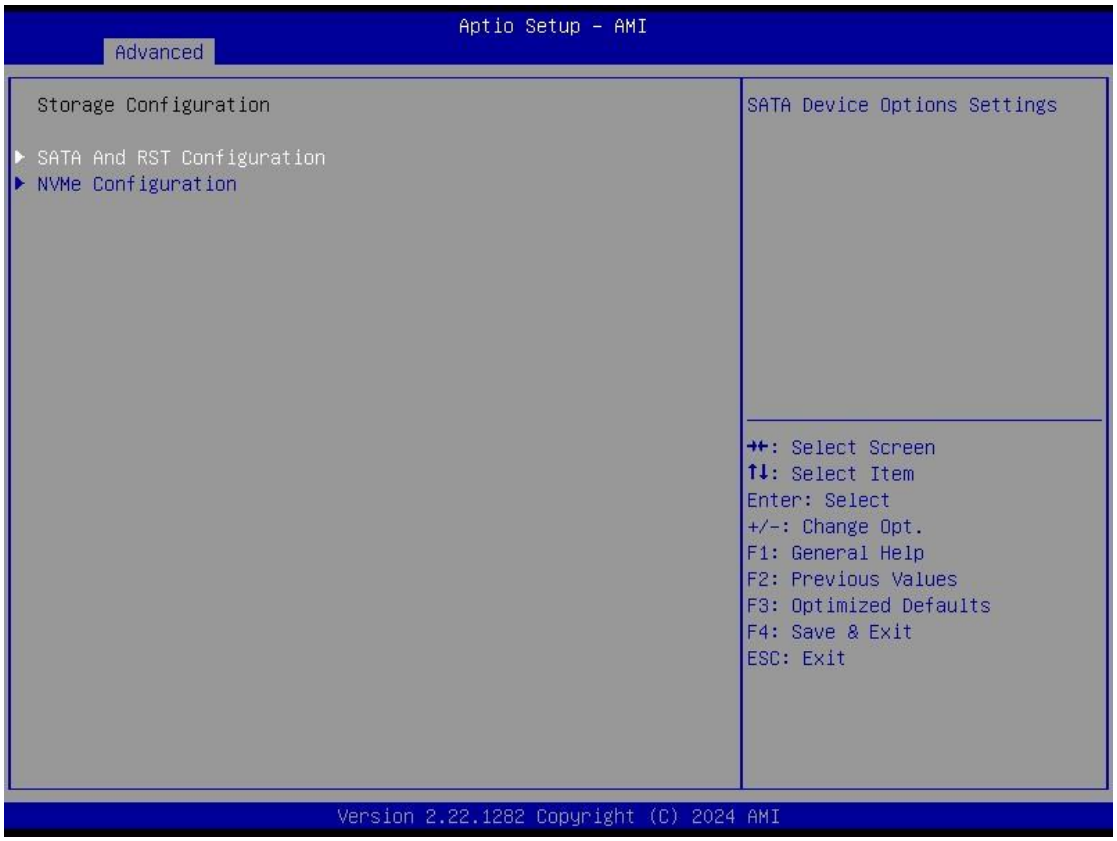
Intel® Speedstep™
Allows more than two frequency ranges to be supported.

Turbo Mode
This item is for enabling or disabling turbo mode. When enabled, it allows processor cores to run faster than marked frequency under certain conditions. The default is Disable Mode.

Configurable TDP Boot Mode
Config TDP Configurations.

Storage Configuration

Users can read the current installed hardware configurations from those SATA ports in the SATA and RST Configuration menu. During system boot up, BIOS will detect the present SATA devices automatically.



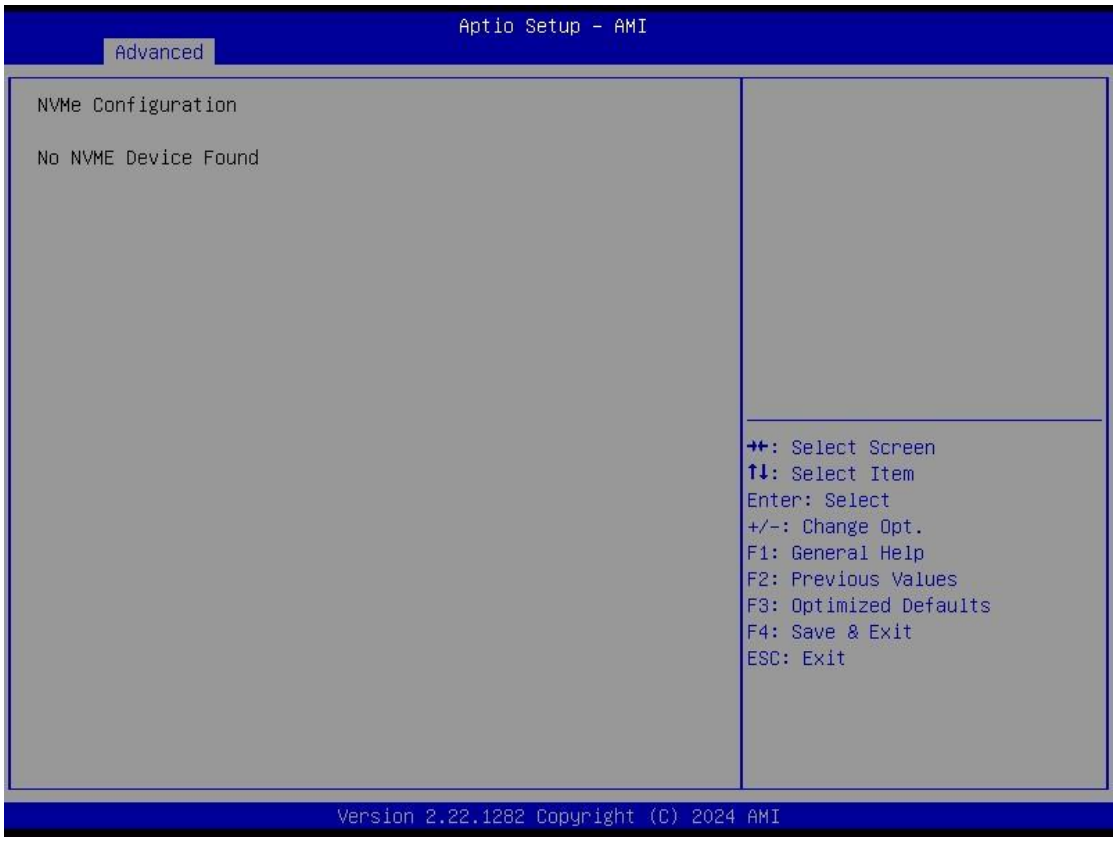
SATA Controller
Enable or disable the SATA Controller feature. The default is Enabled.

VMD Setup menu

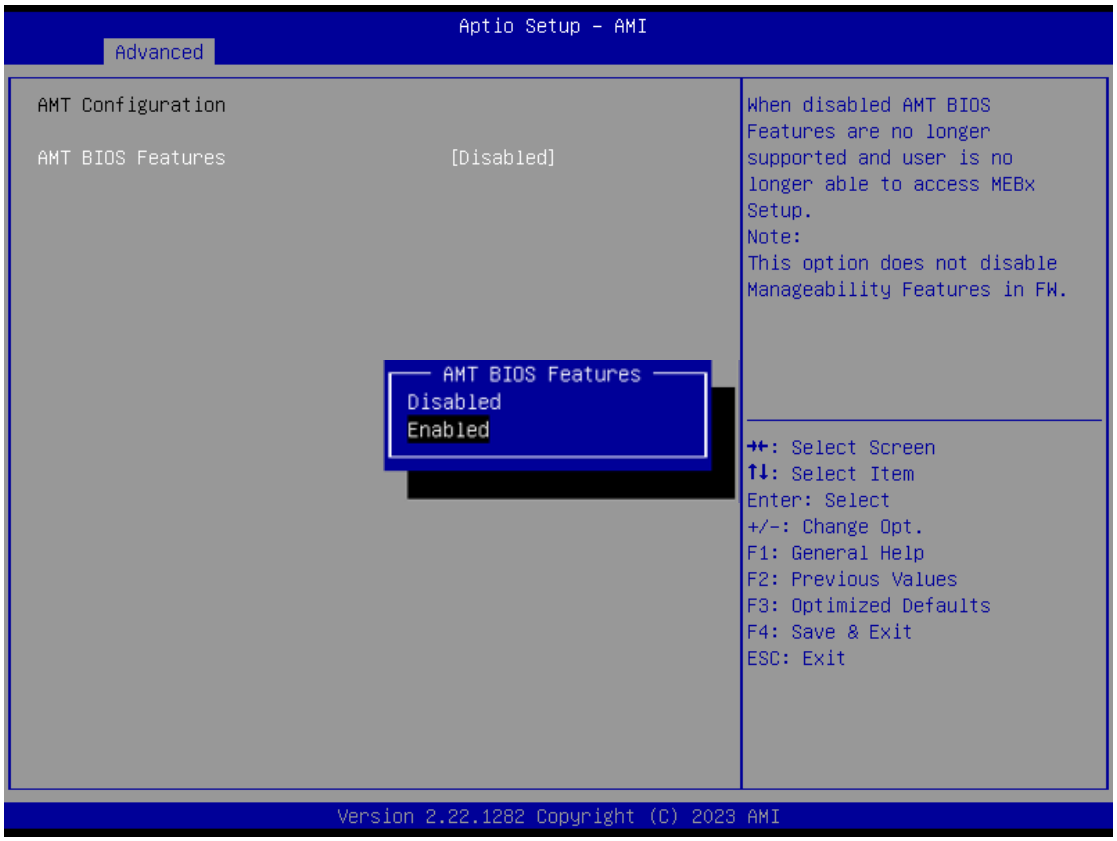
Intel® VMD (Intel® Volume Management Device) is the new way to configure 11th Generation and greater Intel® Core™ Processor-based platforms for Intel® RST management of RAID and Intel® Optane™ memory volumes. please refer to Appendix B for how to set up and configure RAID function through VMD.

NVMe Configuration

Shows NVMe storage here.



AMT Configurations
Users can use this screen to configure AMT parameters.

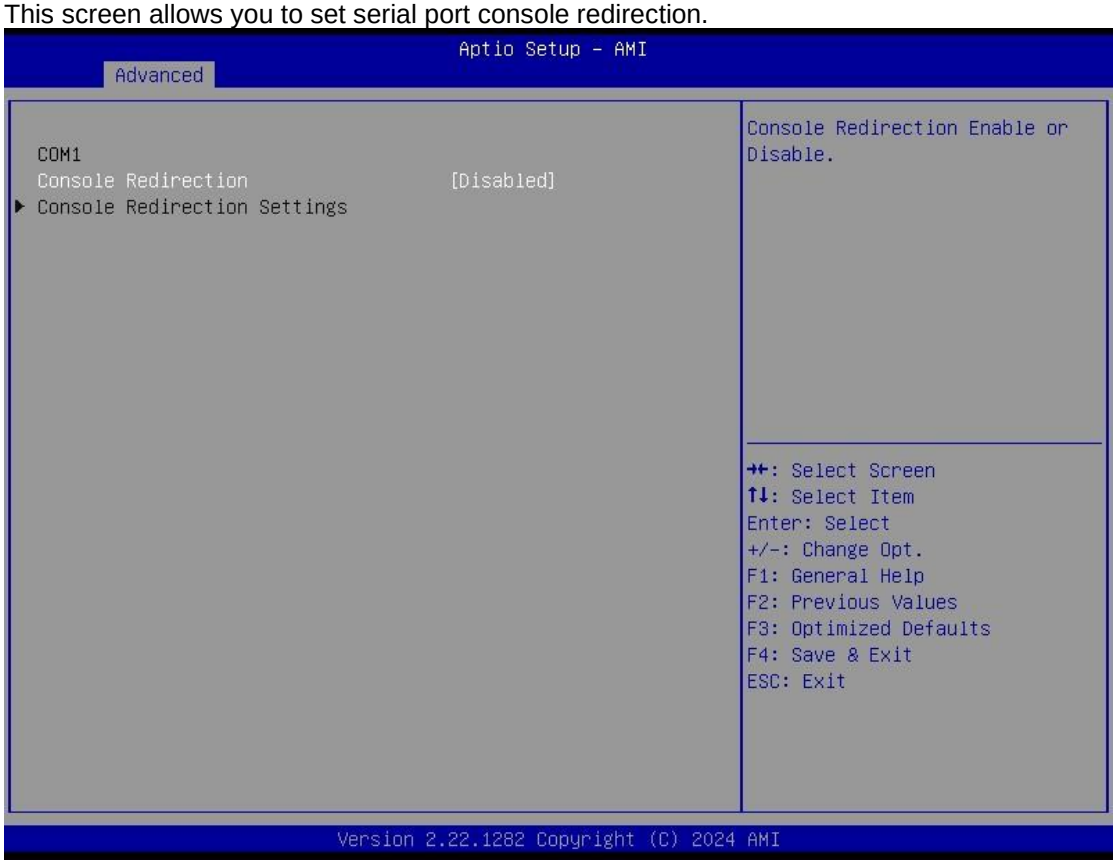


Intel® AMT
Enable or disable Intel® Active Management Technology BIOS Extension.

The default is enabled.

Serial Port Console Redirection

This screen allows you to set serial port console redirection.

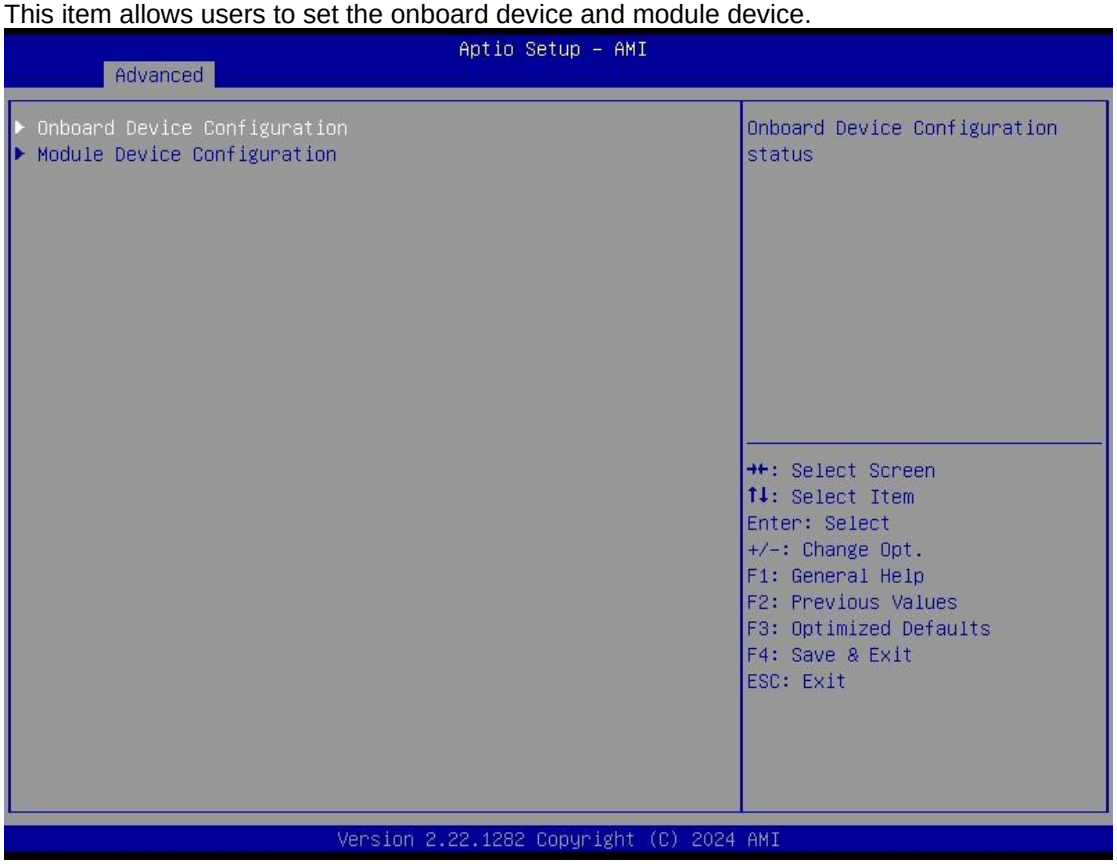


Console Redirection
Console Redirection Enable or Disable.

Console Redirection Settings
The settings specify how the host computer and the remote computer (which the user is using) will exchange data. Both computers should have the same or compatible settings.

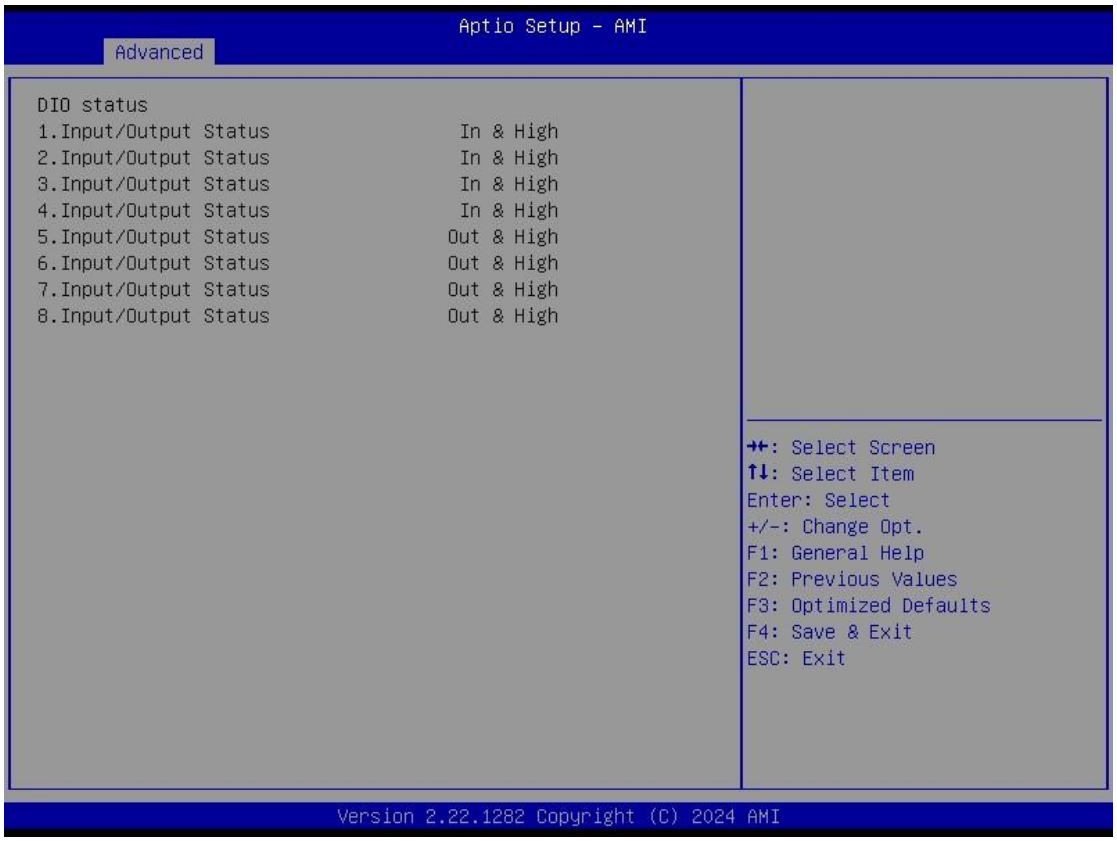
Device Configurations

This item allows users to set the onboard device and module device.



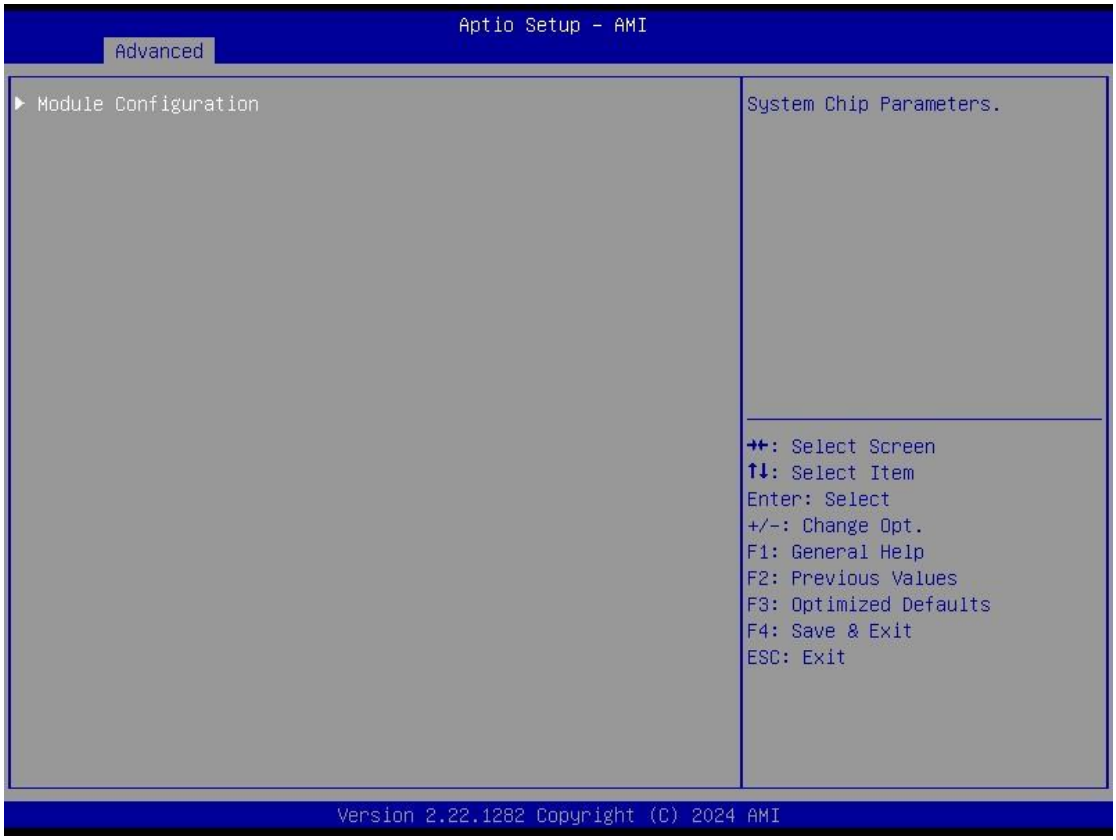
Onboard Device Configuration

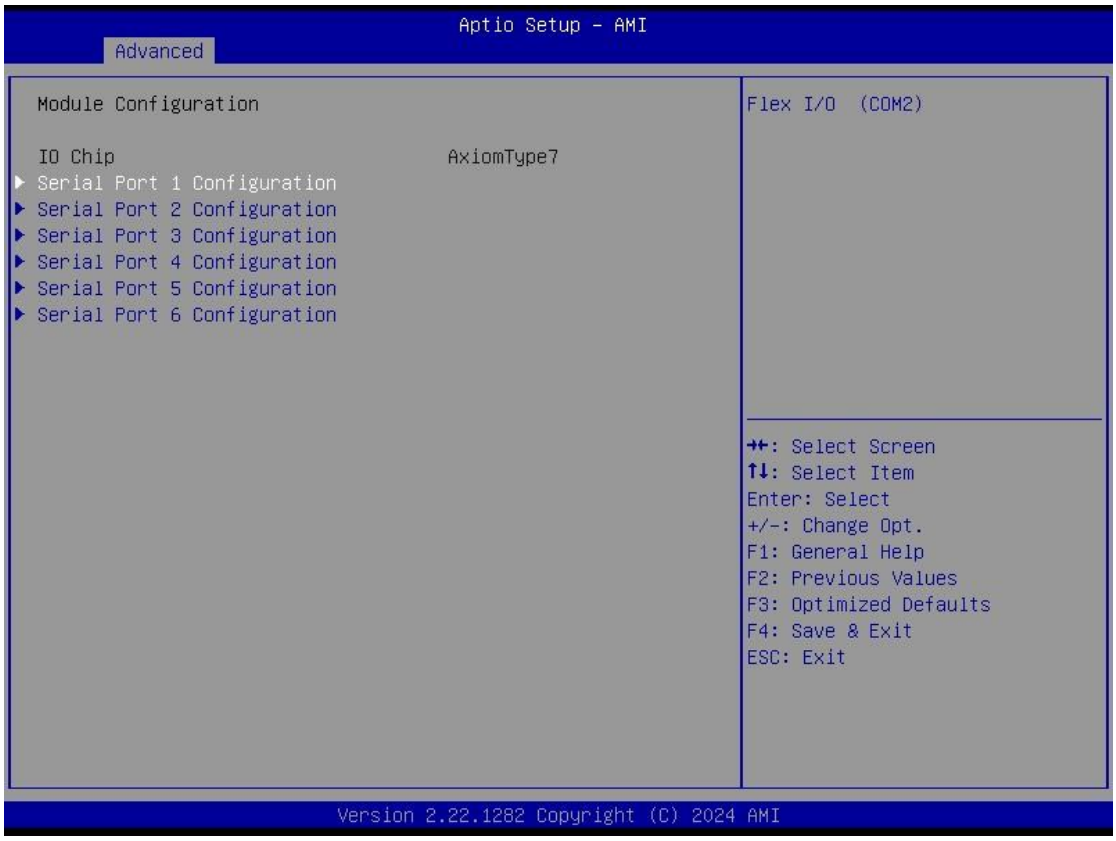
Digital I/O to Input or Output. Default is 4 IN & 4 OUT.

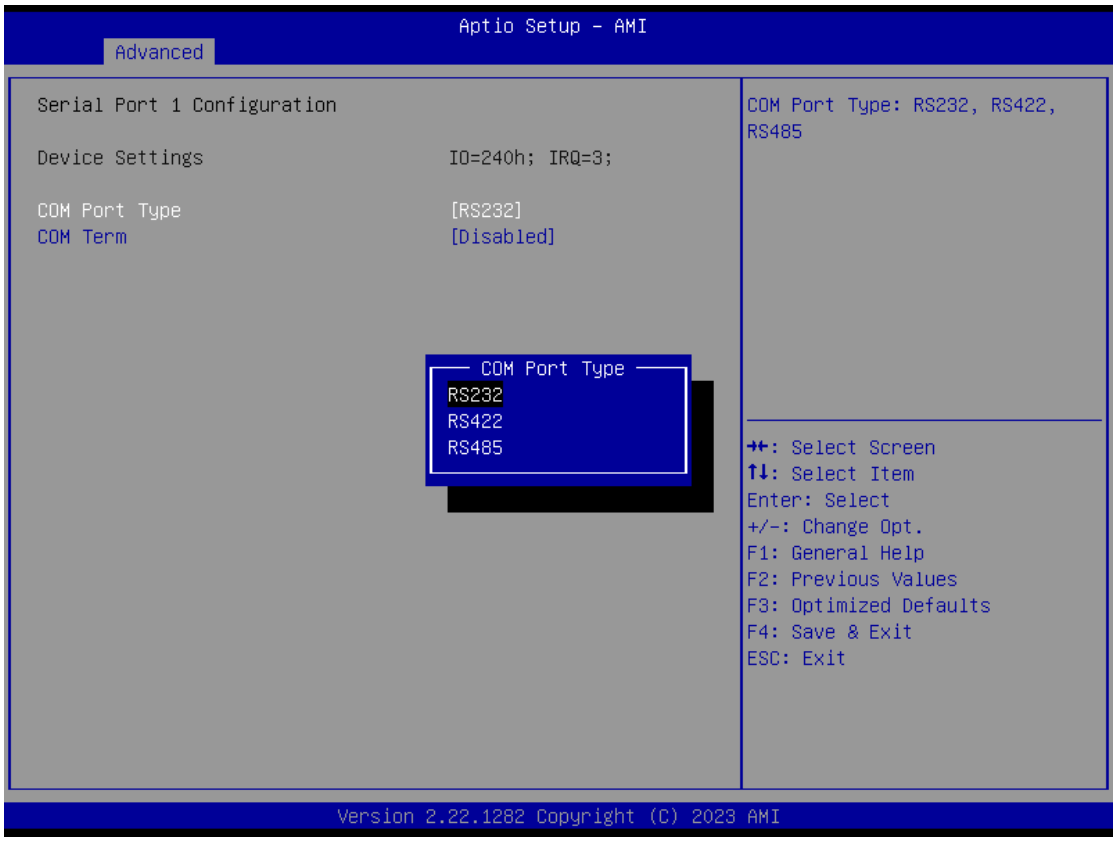


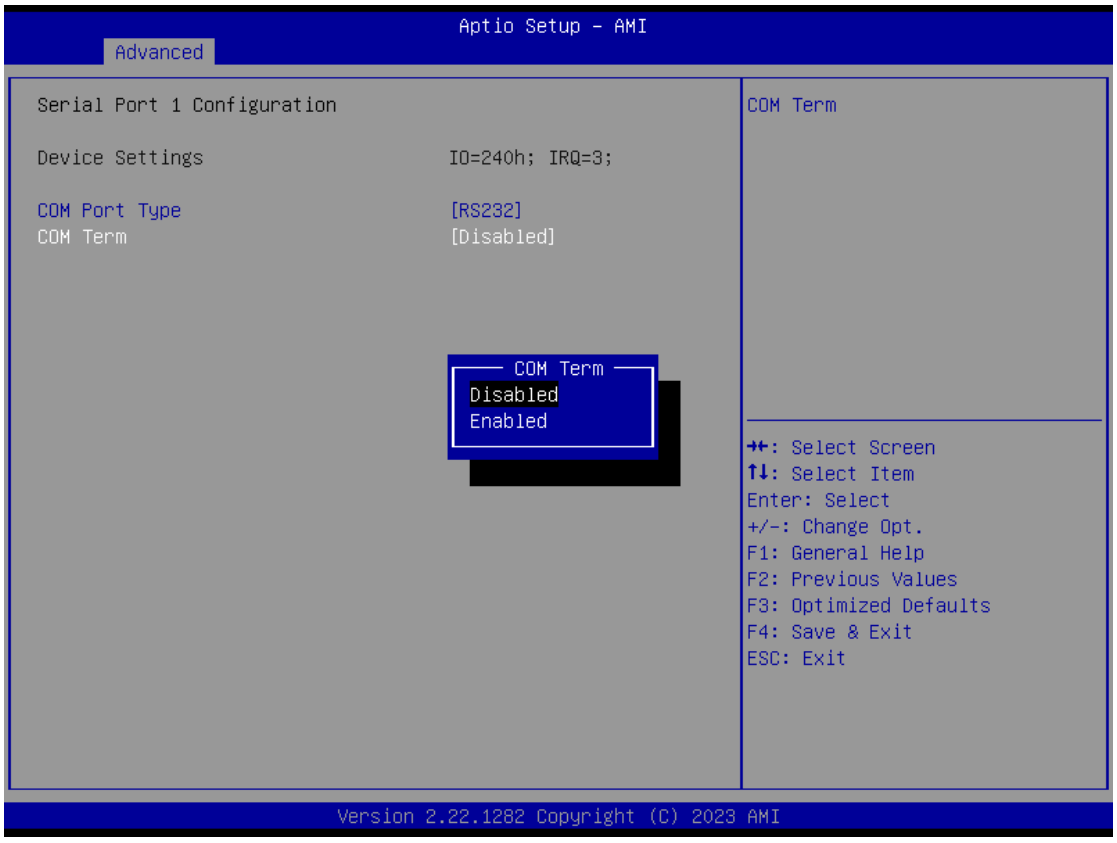
Module Configuration

This item shows the configuration of serial port 1 to 6


Aptio Setup – AMI
Advanced
Module Configuration
System Chip Parameters.
→←: Select Screen
↑↓: Select Item
Enter: Select
+/-: Change Opt.
F1: General Help
F2: Previous Values
F3: Optimized Defaults
F4: Save & Exit
ESC: Exit
Version 2.22.1282 Copyright (C) 2024 AMI


Aptio Setup – AMI
Advanced
Module Configuration
IO Chip AxiomType7
Serial Port 1 Configuration
Serial Port 2 Configuration
Serial Port 3 Configuration
Serial Port 4 Configuration
Serial Port 5 Configuration
Serial Port 6 Configuration
Flex I/O (COM2)
→←: Select Screen
↑↓: Select Item
Enter: Select
+/-: Change Opt.
F1: General Help
F2: Previous Values
F3: Optimized Defaults
F4: Save & Exit
ESC: Exit
Version 2.22.1282 Copyright (C) 2024 AMI


Aptio Setup – AMI
Advanced
Serial Port 1 Configuration
Device Settings
IO=240h; IRQ=3;
COM Port Type
[RS232]
COM Term
[Disabled]
COM Port Type
RS232
RS422
RS485
COM Port Type: RS232, RS422, RS485
→←: Select Screen
↑↓: Select Item
Enter: Select
+/-: Change Opt.
F1: General Help
F2: Previous Values
F3: Optimized Defaults
F4: Save & Exit
ESC: Exit
Version 2.22.1282 Copyright (C) 2023 AMI

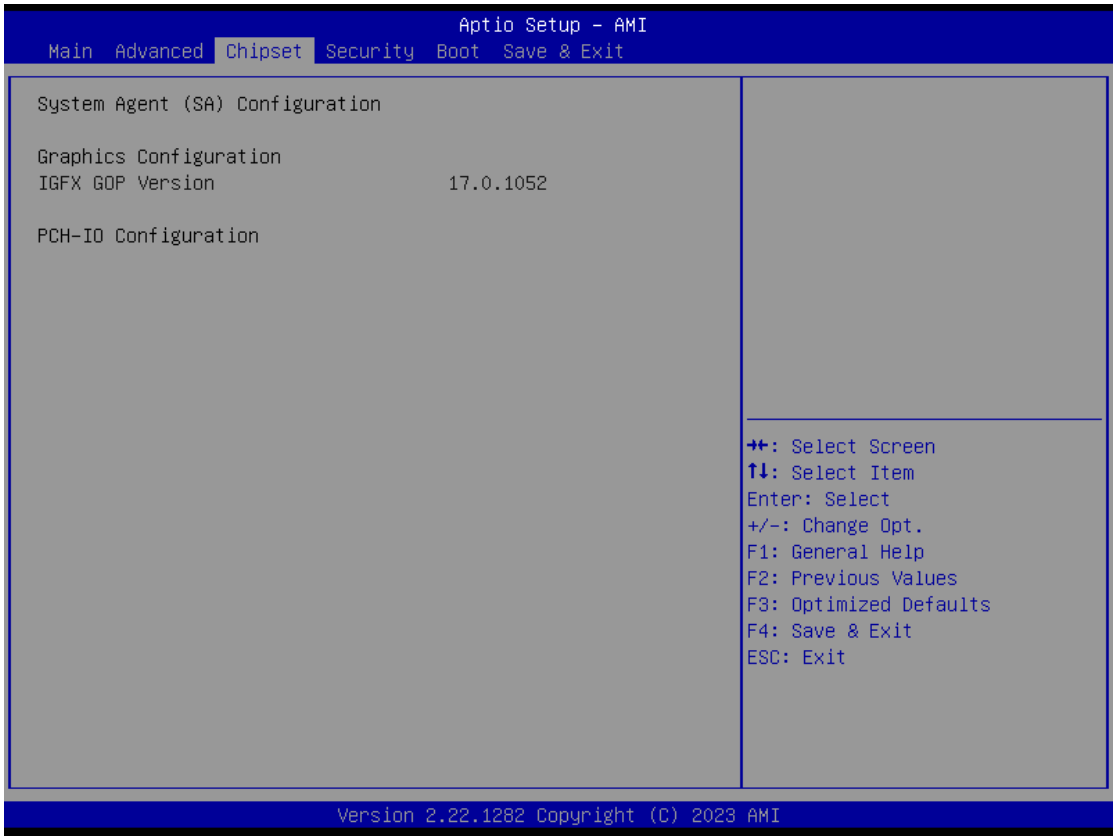

Aptio Setup – AMI
Advanced
Serial Port 1 Configuration
Device Settings
IO=240h; IRQ=3;
COM Port Type
[RS232]
COM Term
[Disabled]
COM Term
Disabled
Enabled
COM Term
→←: Select Screen
↑↓: Select Item
Enter: Select
+/-: Change Opt.
F1: General Help
F2: Previous Values
F3: Optimized Defaults
F4: Save & Exit
ESC: Exit
Version 2.22.1282 Copyright (C) 2023 AMI

4.5 Chipset Menu

The Chipset menu allows users to change the advanced chipset settings. Users can select any of the items in the left frame of the screen to go to the sub menus:

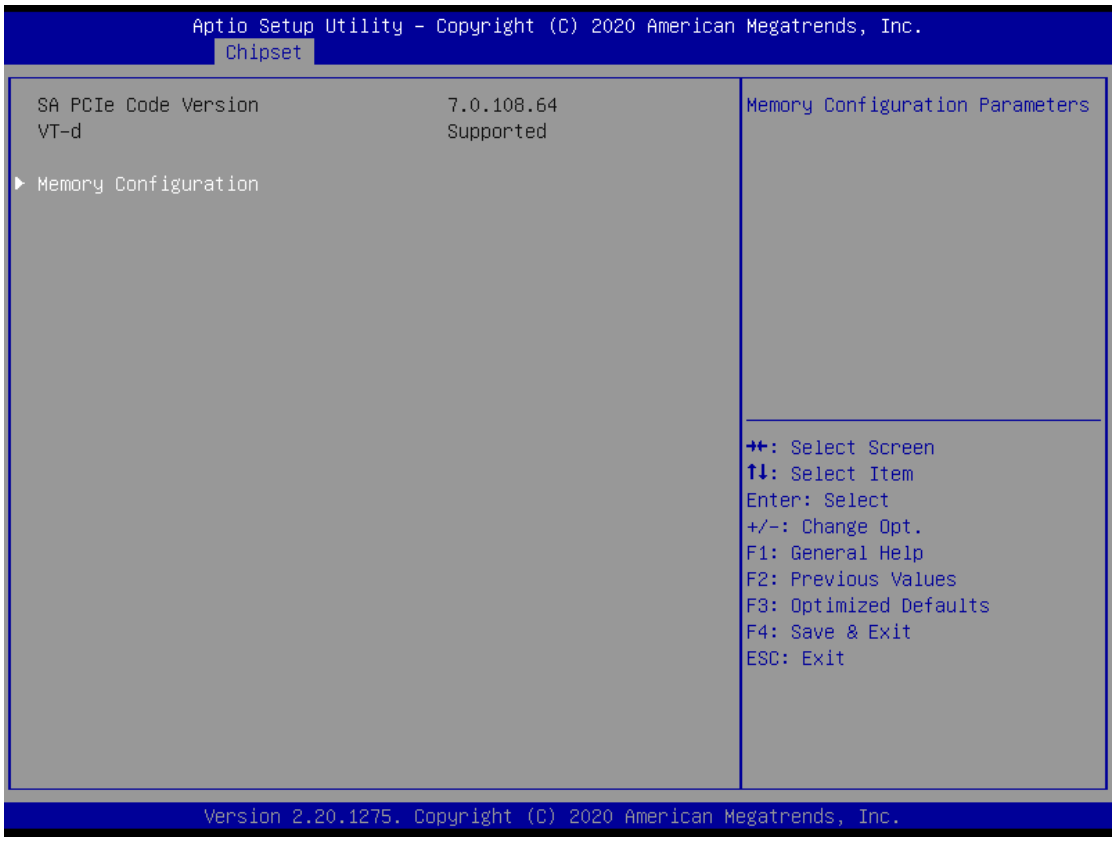
- ► System Agent (SA) Configurations
- ► PCH-IO Configurations

For items marked with "►", please press <Enter> for more options.



System Agent (SA) Configurations

This screen allows users to configure System Agent (SA) parameters. For items marked with "▶" , please press <Enter> for more options.

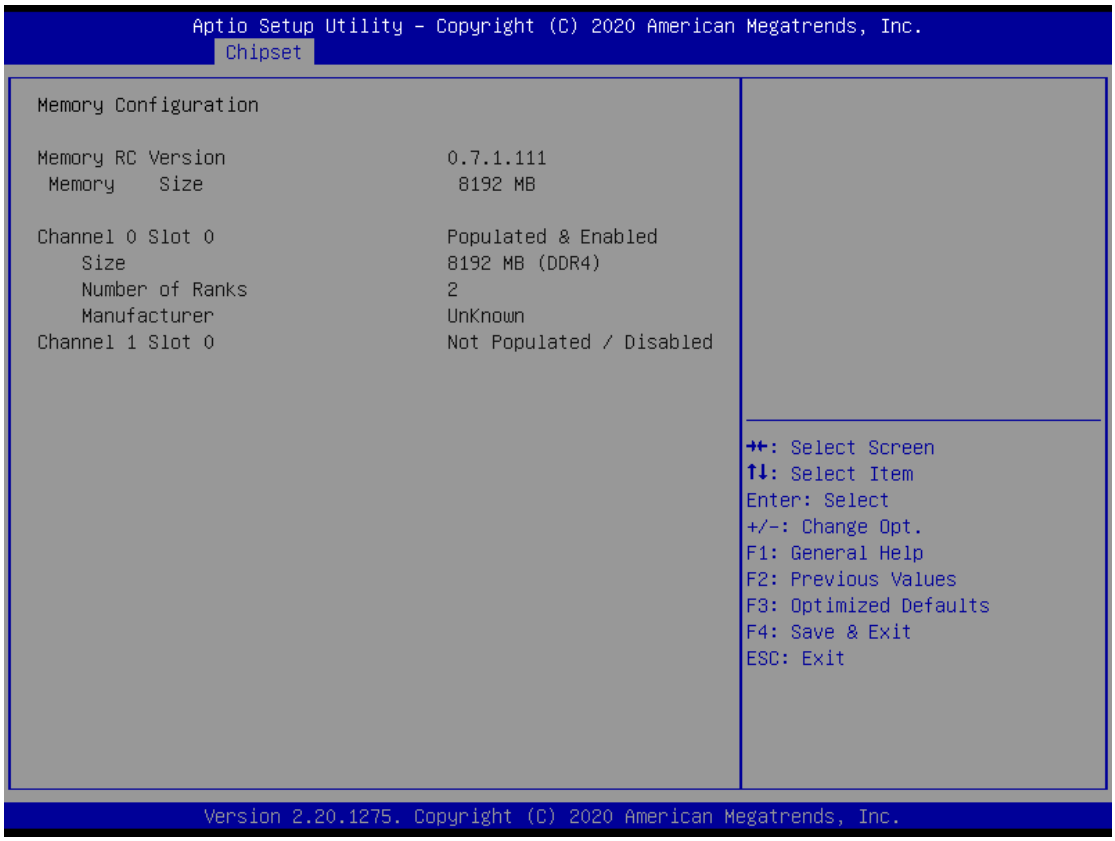


Memory Configuration

Use this item to refer to the information related to system memory.

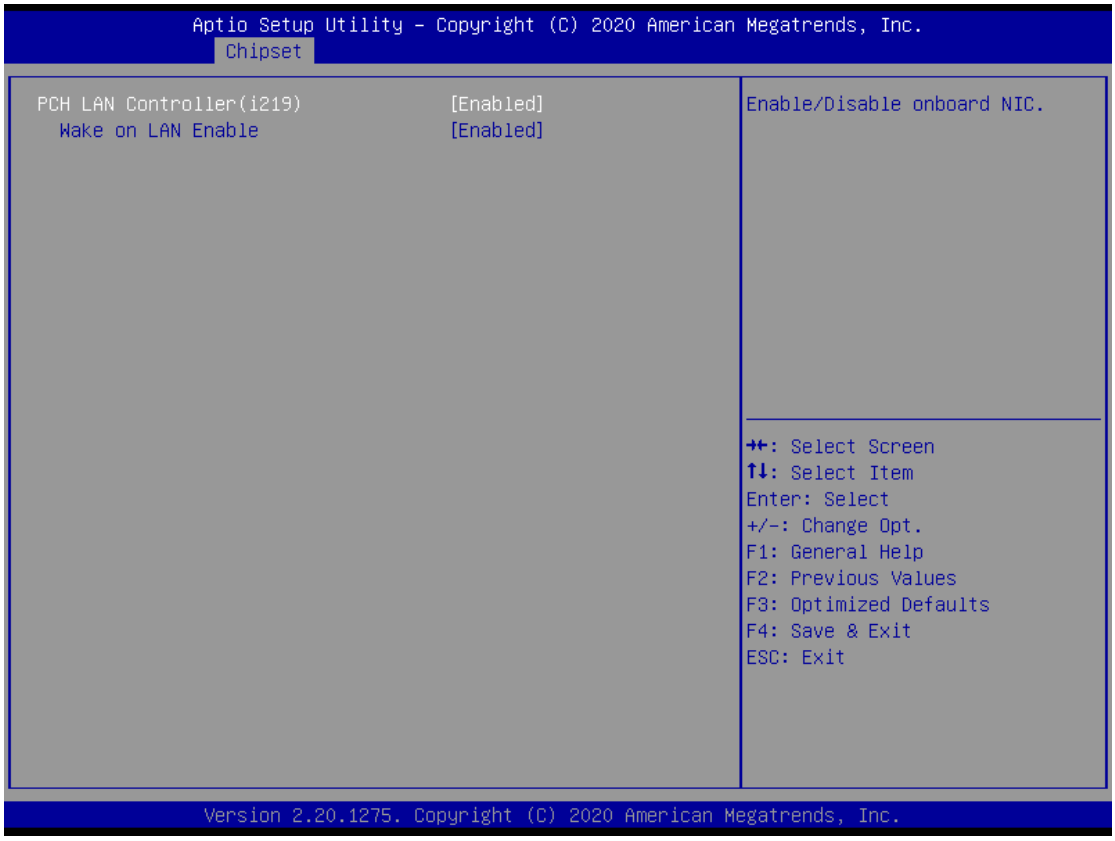
Memory Configurations

This screen shows the system memory information.

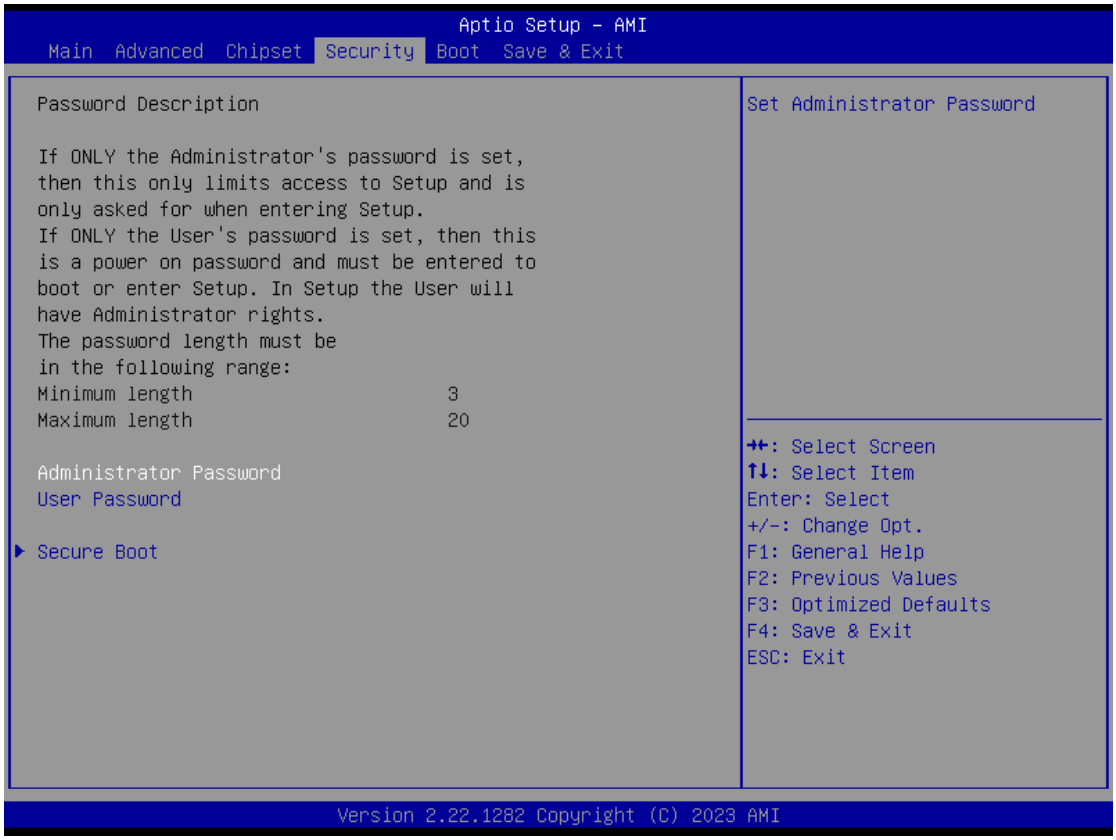


PCH-IO Configurations

This screen allows users to set PCH parameters.



Security Menu

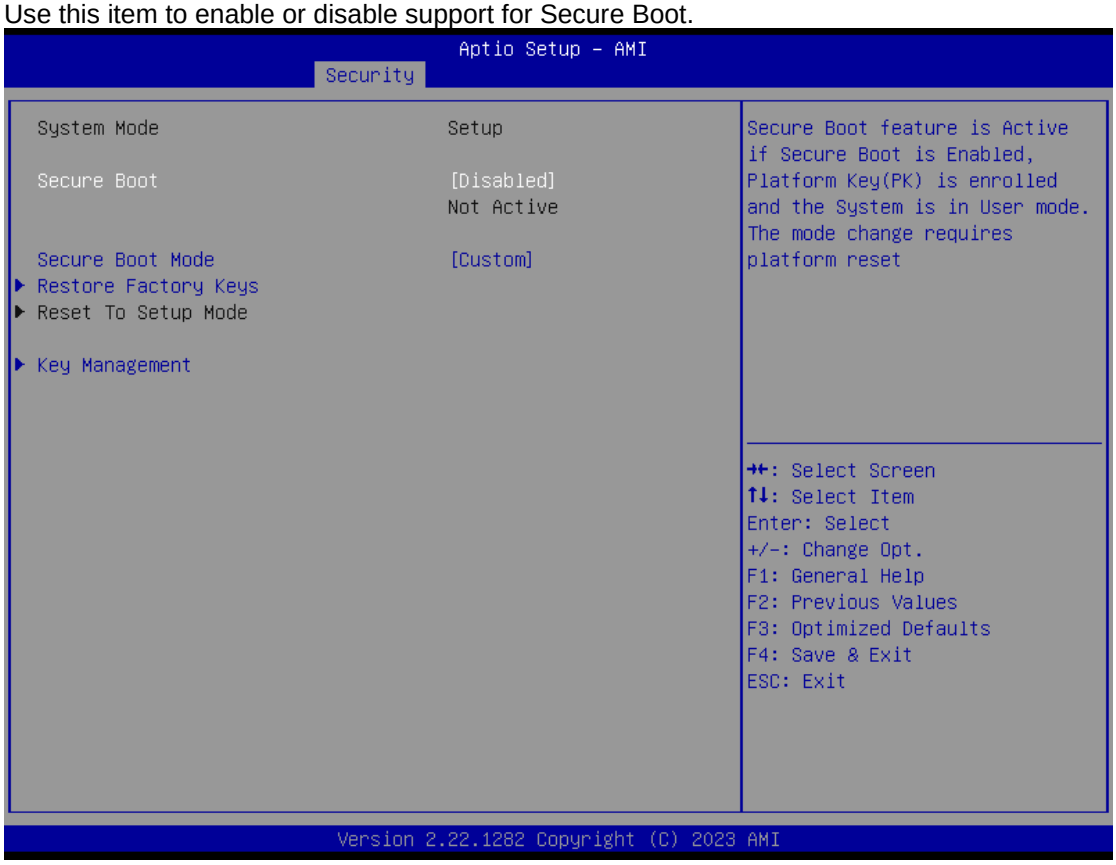


Administrator Password
This item indicates whether an administrator password has been set (installed or uninstalled).

User Password
This item indicates whether a user password has been set (installed or uninstalled).

Secure Boot
Secure Boot feature is Active if Secure Boot is Enabled, Platform Key (PK) is enrolled and the System is in User mode. The mode change requires platform reset.

Secure Boot
Use this item to enable or disable support for Secure Boot.



Secure Boot Mode
Secure Boot mode options: Standard or Custom.
In Custom mode, Secure Boot Policy variables can be configured by a physically present user without full authentication.

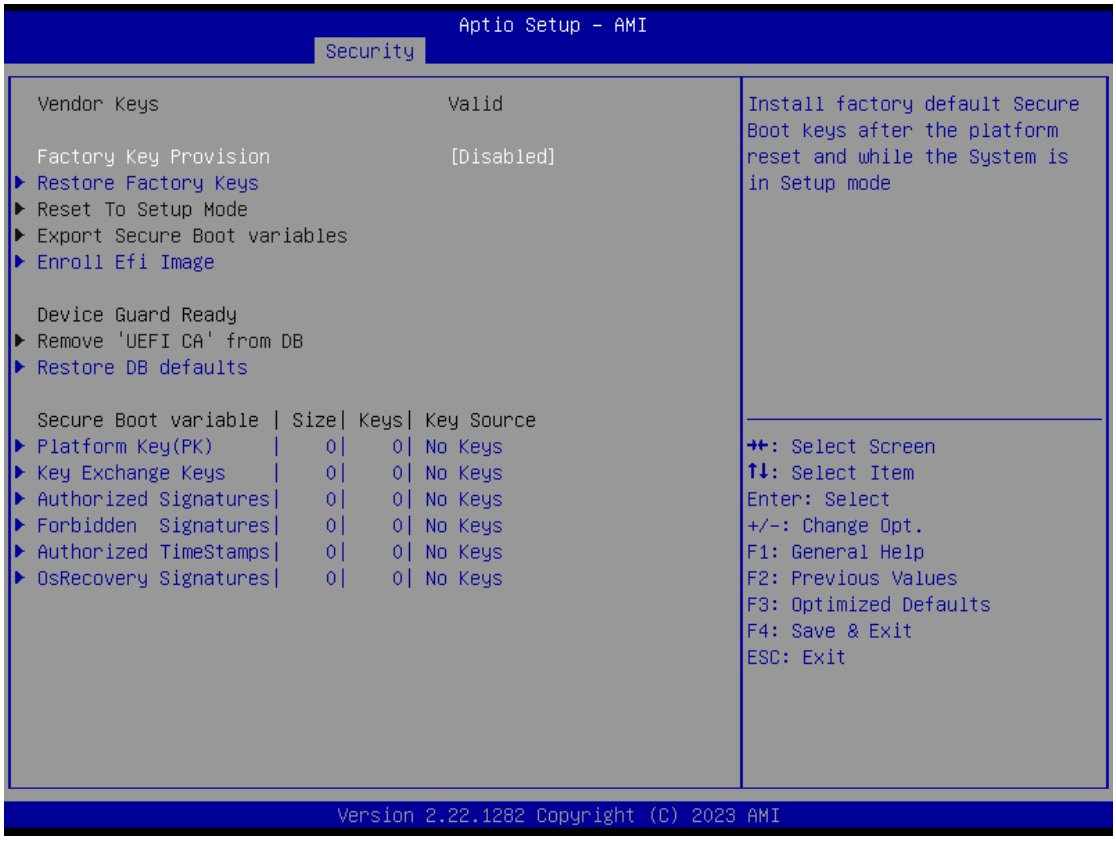
Restore Factory Keys
Use this item to force System to User Mode, to install factory default SecureBoot key databases.

Reset To Setup Mode
Select Yes and press <Enter> to restore the manufacturer default Secure Boot keys. This will also reset the system to User mode. The options are Yes and No.

Key Management
Enables expert users to modify Secure Boot Policy variables without full authentication.

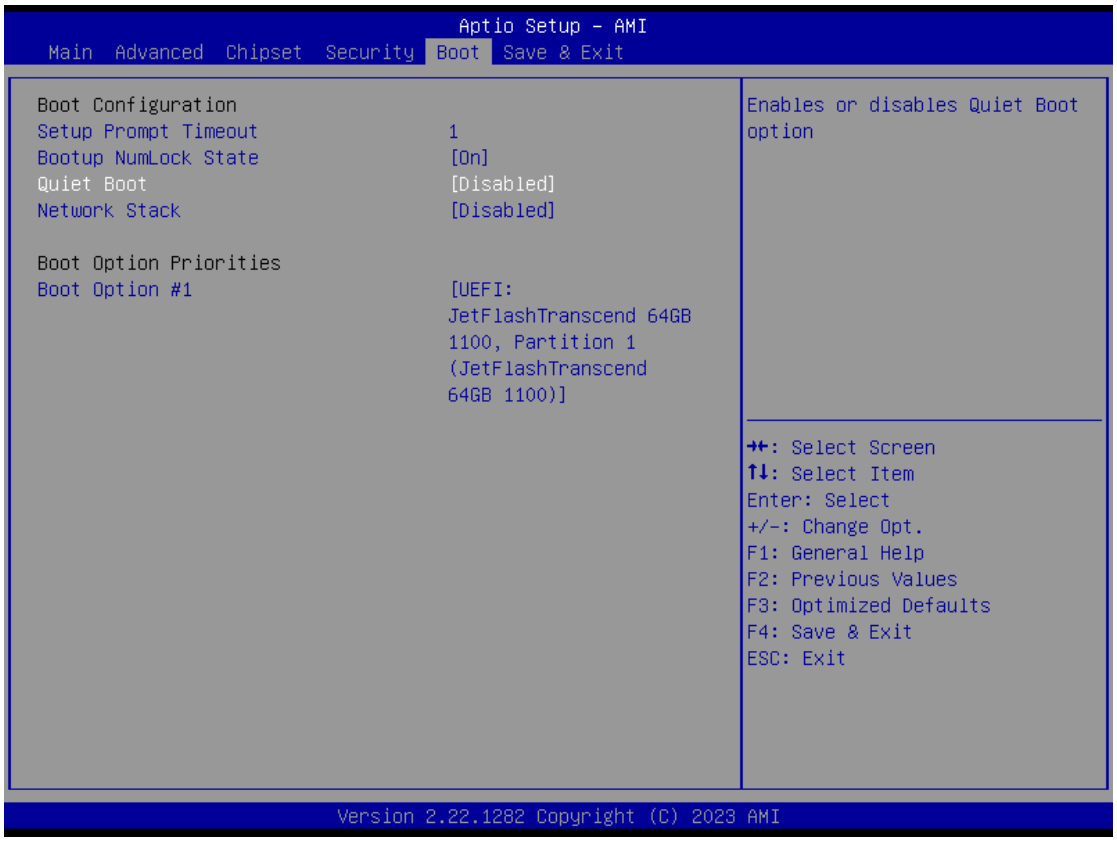
Key Management

Install factory default Secure Boot key the platform rest and while the System is in Setup mode.



4.6 Boot Menu

The Boot menu allows users to change boot options of the system.



Setup Prompt Timeout
Use this item to set up number of seconds to wait for setup activation key where 65535(0xFFFF) means indefinite waiting.

Bootup NumLock State
Use this item to select the power-on state for the keyboard NumLock.

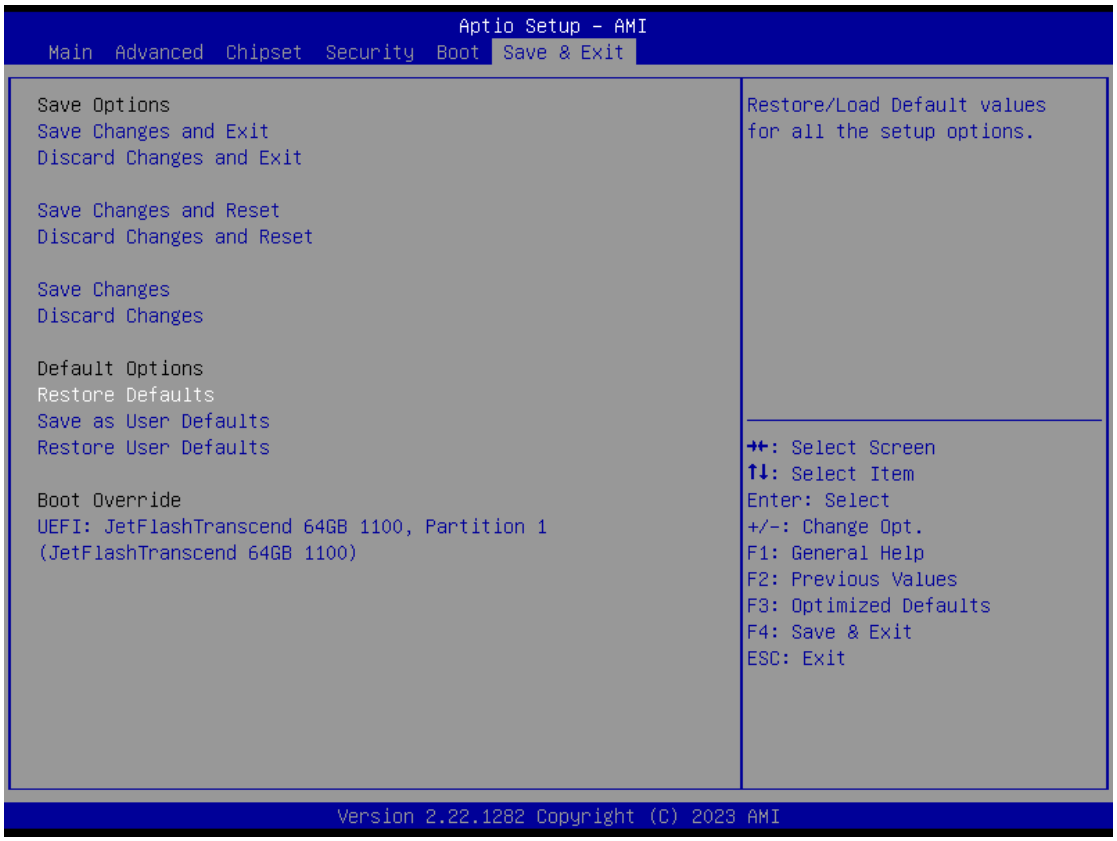
Quiet Boot
Select to display either POST output messages or a splash screen during boot-up.

Network Stack
Enable/Disable UEFI Network Stack. Default setting is Enabled.

Boot Option Priorities
These are settings for boot priority. Specify the boot device priority sequence from the available devices.

4.7 Save & Exit Menu

The Save & Exit menu allows users to load system configurations with optimal or fail-safe default values.



Save Changes and Exit
When users have completed the system configuration changes, select this option to leave Setup and return to Main Menu. Select Save Changes and Exit from the Save & Exit menu and press <Enter>. Select Yes to save changes and exit.

Discard Changes and Exit
Select this option to quit Setup without making any permanent changes to the system configurations and return to Main Menu. Select Discard Changes and Exit from the Save & Exit menu and press <Enter>. Select Yes to discard changes and exit.

Save Changes and Reset
Having completed the system configuration changes, select this option to leave Setup and reboot the computer so the new system configurations take effect. Select Save Changes and Reset from the Save & Exit menu and press <Enter>. Select Yes to save changes and reset.

Discard Changes and Reset
Select this option to quit Setup without making any permanent changes to the system configuration and reboot the computer. Select Discard Changes and Reset from the Save & Exit menu and press <Enter>. Select Yes to discard changes and reset.

Save Changes
Having completed the system configuration changes, select this option to save changes. Select Save Changes from the Save & Exit menu and press <Enter>. Select Yes to save changes.

Discard Changes
Select this option to quit Setup without making any permanent changes to the system configurations. Select Discard Changes from the Save & Exit menu and press <Enter>. Select Yes to discard changes.

Restore Defaults
It automatically sets all Setup options to a complete set of default settings when users select this option. Select Restore Defaults from the Save & Exit menu and press <Enter>.

Save as User Defaults
Select this option to save system configuration changes done so far as User Defaults. Select Save as User Defaults from the Save & Exit menu and press <Enter>.

Restore User Defaults
It automatically sets all Setup options to a complete set of User Defaults when users select this option. Select Restore User Defaults from the Save & Exit menu and press <Enter>.

Boot Override
Select a drive to immediately boot that device regardless of the current boot order.

APPENDIX A
WATCHDOG TIMER

About Watchdog Timer

Software stability is major issue in most applications. Some embedded systems are not watched by human for 24 hours. It is usually too slow to wait for someone to reboot when a computer hangs. The system needs to be able to reset automatically when things go wrong. The watchdog timer gives us solutions in this regard.

The watchdog timer is a counter that triggers a system to reset when it counts down to zero from a preset value. The software starts the counter with an initial value and must reset it periodically. If the counter ever reaches zero which means the software has crashed, the system will reboot.

How to Use the Watchdog Timer

The user can configure the watchdog timer using the watchdog function included in the **AxEAPI** developed by Axiomtek. The sample program show how to enable and disable watchdog timer.

Sample Program

```
#include <stdio.h>
#include <stdlib.h>

#include "EApi.h"

#if defined(WIN32) || defined(_WIN32) || defined(__WIN32__) || defined(__NT__)
    //#define _WIN32_WINNT    0x0501
    #include <windows.h>
    #define _msleep(t) Sleep(t)
#else
    #include <unistd.h>
    #define _msleep(t) usleep(t * 1000)
#endif
 static const char* tag = "eapi_runner";

int main (
    int argc,
    char *argv[]
) {
    EApiStatus_t status;
    uint32_t buffer_size = 64;
    uint32_t value;
    char buffer[64];
    //char exit_request = 0;
    uint32_t wdt_runmode = 0;
    uint32_t wdt_timermode = 0;
    uint32_t wdt_event = 0;    // run wdt
    uint32_t wdt_delay = 0;    // run wdt
    uint32_t wdt_reset = 0;    // reload wdt
    bool wdt_check = false;

    //
    // Initialize
    //
    status = EApiLibInitialize ();
    if (status != EAPI_STATUS_SUCCESS) {
        fprintf (stderr, "fail to initialize, status=%d\n", status);
        goto fail;
    }
```

```
    fprintf (stderr, "---- intialize: ok\n");

    //
    // Watch dog
    //
    while (wdt_runmode != 4) {
        fprintf(stderr, "1. Run WDT mode\n");
        fprintf(stderr, "2. reload WDT mode\n");
        fprintf(stderr, "3. Stop WDT mode\n");
        fprintf(stderr, "4. Exit\n");
        fprintf(stderr, "Select watch dog run mode: ");
        scanf("%d", &wdt_runmode);

        switch (wdt_runmode)
        {
        case 1:
            fprintf(stderr, "Set watch dog timer mode (0 means second, 1 means minute)\n");
            scanf("%d", &wdt_timermode);

            if (wdt_timermode == 0)
            {
                fprintf(stderr, "WDT timer mode is second.\n");
            }
            else if (wdt_timermode == 1)
            {
                fprintf(stderr, "WDT timer mode is minute.\n");
            }
            else
            {
                wdt_timermode = 0;
                fprintf(stderr, "WDT timer mode is second.\n");
            }

            fprintf(stderr, "Set watch dog count (Count range is 5-255): ");
            scanf("%d", &wdt_event);

            if (wdt_event < 5 || wdt_event > 255) {
                fprintf(stderr, "Watch dog count is invalid!!!\n\n");
                //goto invalid;
                break;
            }

            wdt_reset = wdt_event;
            if (wdt_timermode == 0)
                status = EApiWDogStart(wdt_delay * 1000, wdt_event * 1000, wdt_reset * 1000);
            else if (wdt_timermode == 1)
                status = EApiWDogStart(wdt_delay * 60000, wdt_event * 60000, wdt_reset * 60000);

            if (status != EAPI_STATUS_SUCCESS) {
                fprintf(stderr, "WDog run Error!!!\n\n");
                goto fail;
            }
            else {
                wdt_check = true;
            }
            break;
        case 2:
            if (wdt_check == true)
            {
                status = EApiWDogTrigger();
                if (status != EAPI_STATUS_SUCCESS) {
                    fprintf(stderr, "WDog Reload Error!!!\n\n");
                    goto fail;
                }
            }
            else
            {
                fprintf(stderr, "Need to run WDT first!!!\n\n");
            }
            break;
        case 3:
            status = EApiWDogStop();
            if (status != EAPI_STATUS_SUCCESS) {
                fprintf(stderr, "WDog Stop Error!!!\n\n");
                goto fail;
            }
            else {
                wdt_check = false;
            }
            break;
        case 4:
            fprintf(stderr, "Exit program!!!\n\n");
```

```
            break;
        default:
            fprintf(stderr, "WDog run mode error!!!\n\n");
            break;
        }
    }

fail :

    return 0;
}
```

This page is intentionally left blank.

APPENDIX B
USB power on/off

About USB power on/off

The USB on/off function is an important feature in AMR controllers. It allows users to easily and quickly turn on or off the USB port, which is essential for connecting devices such as cameras.

Copy the sample code and make it as a bash script, then run the script with arguments to switch on/off the power of USB devices.

Usage:

Test_USB_ROBOX300.sh

- # Test_USB_ROBOX300.sh *x y*
- *x*: USB port, 1 ... 6
- *y*: ON/OFF, 0=OFF 1=ON

Sample Code

```
#!/bin/bash
# Program:
#ROBOX300/PSB555 USB Power ON/OFF Test Program
# History:
# 2024/01/12First release
PATH=/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin:~/bin
export PATH
clear

USB1_ADDR=0xFD6D0900
USB2_ADDR=0xFD6D0910
USB3_ADDR=0xFD6D0920
USB4_ADDR=0xFD6D0930
USB5_ADDR=0xFD6D0A10
USB6_ADDR=0xFD6D0A20

USB_CLEAR=0xfffffff0
if ! [[ "$1" =~ ^[0-9]+$ ]] || ! [[ "$2" =~ ^[0-9]+$ ]]
then
  echo "Test_USB_ROBOX300.sh x y"
  echo "x: USB port, 1 ... 6"
  echo "y: ON/OFF, 0=OFF 1=ON"
else
  if [ $# -eq 0 ] || [ $# -eq 1 ] || [ $# -gt 2 ] || [ $1 -gt 6 ] || [ $1 -lt 1 ] ||
[ $2 -gt 1 ] || [ $2 -lt 0 ]
  then
    echo "Test_USB_ROBOX300.sh x y"
    echo "x: USB port, 1 ... 6"
    echo "y: ON/OFF, 0=OFF 1=ON"
  else

    case $1 in
      1) USB_PORT=$USB1_ADDR
    ;;
      2) USB_PORT=$USB2_ADDR
    ;;
      3) USB_PORT=$USB3_ADDR
    ;;
      4) USB_PORT=$USB4_ADDR
    ;;
      5) USB_PORT=$USB5_ADDR
    ;;
      6) USB_PORT=$USB6_ADDR
    ;;
    esac

    USB_STATUS=$2
    READDATA=$(sudo busybox devmem $USB_PORT)
    sudo busybox devmem $USB_PORT w $((($READDATA&$USB_CLEAR)+$USB_STATUS))
    READDATA=$(sudo busybox devmem $USB_PORT)
    echo "========== set USB port $1 $2 =========="
  fi
fi
```

APPENDIX C
Digital I/O

Sample Program

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#include "EApi.h"

#if defined(WIN32) || defined(_WIN32) || defined(__WIN32__) || defined(__NT__)
    //#define _WIN32_WINNT    0x0501
    #include <windows.h>
    #define _msleep(t) Sleep(t)
#else
    #include <unistd.h>
    #define _msleep(t) usleep(t * 1000)
#endif

#define MAX_PIN_COUNT (64)
#define STDIN_INPUT_MAX_LENGTH (4)

static const char* tag = "eapi_runner";

int main (
    int argc,
    char *argv[]
) {
    EApiStatus_t status;
    uint32_t value; // Number of DIO pin count
    char input_buf[STDIN_INPUT_MAX_LENGTH];

    //
    // Initialize
    //
    status = EApiLibInitialize ();
    if (status != EAPI_STATUS_SUCCESS) {
        fprintf (stderr, "fail to initialize, status=%d\n", status);
        goto fail;
    }
    fprintf (stderr, "---- intialize: ok\n");

    //
    // Internal DIO
    //
    status = EApiHWMGetCaps (
        EAPI_ID_DIO_COUNTER_INTERNAL,
        &value
    );
    if (status != EAPI_STATUS_SUCCESS) {
        fprintf (stderr, "fail to get dio internal cap, status=%d\n", status);
        goto fail;
    }
    fprintf (stderr, "---- dio internal cap: %u\n", value);

    if (value) {
        uint32_t pin_dir[MAX_PIN_COUNT];
        uint32_t pin_value[MAX_PIN_COUNT];
        char key = 0;
        uint32_t i, j;

        // Hardcode to change DIO pin 4 to output direction.
        i = 4;
        pin_dir[4] = EAPI_OUTPUT;
        status = EApiBoardSetValueWithIndex(
```

```
            EAPI_ID_BOARD_SENSOR_DIO_DIRECTION_INTERNAL,
            4,
            pin_dir[4]
        );

        if (status != EAPI_STATUS_SUCCESS) {
            fprintf (stderr,
                "fail to set internal dio dir, index=%d status=%d\n",
                i,
                status
            );
            goto fail;
        }

        // Dump DIOs direction of all pins
        for (i = 0; i < value; i++) {
            status = EApiBoardGetValueWithIndex(
                EAPI_ID_BOARD_SENSOR_DIO_DIRECTION_INTERNAL,
                i,
                &pin_dir[i]
            );
            if (status != EAPI_STATUS_SUCCESS) {
                fprintf (stderr,
                    "fail to get internal dio dir, index=%d status=%d\n",
                    i,
                    status
                );
                goto fail;
            }
            fprintf (stderr, "---- dio internal dir[%d]: %u\n", i, pin_dir[i]);
        }

        while (1) {
            // Dump DIOs level of all pins
            for (i = 0; i < value; i++) {
                status = EApiBoardGetValueWithIndex(
                    EAPI_ID_BOARD_SENSOR_DIO_INTERNAL,
                    i,
                    &pin_value[i]
                );
                if (status != EAPI_STATUS_SUCCESS) {
                    fprintf (stderr,
                        "fail to get internal dio value, index=%d status=%d\n",
                        i,
                        status
                    );
                    goto fail;
                }
                fprintf (stderr, "---- dio internal value[%d]: %u\n", i,
pin_value[i]);
            }


            fprintf (stderr, "press `q` to exit or 0-%d to invert value: ", value -
1);

            // Read char until 0x0a or buffer full from stdin
            memset(input_buf, 0, STDIN_INPUT_MAX_LENGTH);
            j = 0;
            do {
                key = getchar();
                input_buf[j++] = key;
            } while ((j < (STDIN_INPUT_MAX_LENGTH - 1)) && key != 0x0A);


            if (input_buf[0] == 'q' || input_buf[0] == 'Q')
                break;

            i = atoi(input_buf);
            // Toggle the voltage level of the selected pin
            if (i >= 0 && i < value) {
                status = EApiBoardSetValueWithIndex(
```

```
                    EAPI_ID_BOARD_SENSOR_DIO_INTERNAL,
                    i,
                    pin_value[i] ? 0 : 1
                );

                if (status != EAPI_STATUS_SUCCESS) {
                    fprintf (stderr,
                        "fail to set internal dio value, status=%d\n",
                        status
                    );
                    goto fail;
                }
            }
        }
    }

    //
    // External DIO
    //
    status = EApiHWMGetCaps (
        EAPI_ID_DIO_COUNTER_EXTERNAL,
        &value
    );
    if (status != EAPI_STATUS_SUCCESS) {
        fprintf (stderr, "fail to get dio external cap, status=%d\n", status);
        goto fail;
    }
    fprintf (stderr, "---- dio external cap: %u\n", value);

    if (value) {
        ;
    }


fail :

    return 0;
}
```

This page is intentionally left blank.

APPENDIX D
CANbus

Install the CANbus driver first, then user can initial the CAN devices with below shell command:

```
#!/bin/bash

ifconfig $1 down
ip link set $1 type can restart-ms 100
ip link set $1 type can bitrate $2 sample-point 0.875
ip link set $1 type can berr-reporting on
ifconfig $1 txqueuelen 1000
tc qdisc add dev $1 root handle 1: pfifo
ifconfig $1 up
```

The 1st parameter $1 can be CAN0 or CAN1

The 2nd parameter $2 can be the baud rate of the can device.

Run below command to install CAN tool

```
apt-get install can-utils
```

Use ***candump*** to revice data, **cangen** and ***cansend*** tosend specific data & ID, user can also use ***cansend*** to send random data & ID.

This page is intentionally left blank.