TOPFEEL

# AIOT7-GML

## User's Manual

Version 1.0

## Disclaimer

The intellectual property of this manual belongs to our company. The ownership of all of the products, including accessories and software etc. belong to our company. No one is permitted to copy, change, or translate without our written permission.

We compiled this manual based on our careful attitude, but we can not guarantee the accuracy of the contents. This manual is purely technical documentation, without any hint or other meanings, and we won't commit users' misunderstanding of the typesetting error.

Our products are in continuous improvement and updating, Therefore, we retain the right that we won't give notice to the users in future.

## Copyright

All of the trademark in this manual belong to their own registered company. All of the products name is only for identification, its title belongs to its manufacturer or brand owner.

## **Table of Contents**

Chapter 1 Introduction	1
1.1 Package Checklist	1
1.2 Motherboard specifications	2
1.3 Motherboard structure	4
1.4 Motherboard IO interface structure	4
1.5 Motherboard Layout	5
1.6 I/O Interface	6
Chapter 2 Hardware Setup ·····	7
2.1 install I / O Rear Panel·····	7
2.2 Installing the Mainboard	7
2.3 Installing Memory Module	8
2.4 Connect with External Devices	9
2.4.1 Serial ATA Connector	9
2.4.2 MINI_PCIE、 PCIE_X1 ······	9
2.4.3 M.2-KEY Slots	10
Chapter 3 Jumpers & Headers Setup	11
3.1 Checking Jumper Settings ·····	11
3.2 Jumpers settings	11
3.3 COM1/2/3/4/5/6 interface	13
3.4 LPT interface	13
3.5 JGPIO interface	14
3.6 DRAWER interface	14
3.7 CPU_FUN interface	15
3.8 SMB interface ······	15
3.9 SYS_FUN interface	16
3.10 PS2 interface ······	16
3.11 DC_OUT main board power interface	17
3.12 ATX 12V and HDD PWR interface	17
3.13 VGA 1 interface	18
3.14 F AUDIOI interface	18
3.15 SPEAKERS interface	19
3.16 USB interface	19
3.17 INVT interface ·······	20
3.18 F PANEL interface	20

3.19 JLPC interface 21
3.20 EDP interface 22
3.21 LVDS interface 23
Chapter 4 BIOS Setup Utility 24
4.1 About BIOS Setup
4.2 BIOS Setup24
4.2.1 Enter BIOS25
4.2.2 Control Keys25
4.3 Main26
4.4 Advanced
4.5 Chipset 28
4.6 Security
4.7 Boot
4.8 Save & Exit
Chapter 5 Driver Installation
Chapter 6 WDT Programming Guide
Chapter 7 GPIO Programming Guidance 36
Ordering Information
Toxic and hazardous substances or elements logo40

## **Chapter 1 Introduction**

## 1.1 Package Checklist

Thank you for choosing our products.

Before using your product, please make sure your packaging is complete, if there have damage or you find any shortage, please contact your supplier as soon as possible.

- Motherboard x 1
- Drive disc x 1 (Industrial packaging: 2PCS / box)
- SATA hard drive adapter cable X 1

The above accessories and specifications are only for reference, we reserve the modify rights.

## **1.2 Motherboard specifications**

Processor	- Adopt Intel Gemini Lake series j4005 / j4025, j4105 / j4125, j5005, n4100 board stick CPU standard board stick j4005 dual core two Rui frequency to 2.7GHz Or j4105 quad core one point five Rui frequency to 2.5GHz		
Chipset	- Intel Gemini Lake SOC		
Memory	- 1 <b>x</b> DDR4 sodimm memory socket, up to support ddr4-2400 single up to 8GB		
Display controller	- Intel CPU integrated display controller UHD 600		
Display interface	- VGA + HDMI + LVDS or EDP (jumper selection) HDMI 2.0 max resolution up to 4096*2160@60Hz LVDS, Dual Channel 24bit, max resolution up to 1920*1200@60 或eDP, max resolution up to 4096*2160@60Hz		
Storage	- 1 $\textbf{x}$ SATA3.0 , - 1 $\textbf{x}$ M.2 Support 2242 / 2280 SSDs (not used M.2 When you choose, you can choose to choose the EMMC solid-state storage of the optional customized board: 64GB or 128GB		
Audio	- Rear IO supports mic in, line out (single layer), audic output on front panel of chassis, and digital audio output with power amplifier (3w, 4Pin wafer)		
Network	- 板载2个Realtek 8111 RJ45千兆网口(默认一个千兆口)		
USB	- 2 <b>x</b> USB3.0 (rear), two USB2.0 (rear) single interface boards, eight USB2.0 pins, six USB2.0 pins of single interface boards, among which usb7-usb10 are USB hub rear expansion		
I <sup>2</sup> C/SM bus	- 1 <b>x</b> 4Pin SM bus interface		
Serial port	- COM1 supports RS422 / 485, com2 & COM3 supports pin9 + 5 / 12V power supply, and COM3 can select TTL level		
PS/2	- PS/2 keyboard and mouse board pin		
Digital I/O	- 8-bit digital I / 0, supply power and ground, + 5V level		
LPC extension interface	- Provide one LPC local bus interface (expandable security encryption module, serial port module)		
Cash Drawer	- 1 <b>x</b> 4Pin CashDrawer control interface		
Parallel port/LPT	- 1 <b>x</b> parallel pin		
Power Supply	- DC_In single voltage + 12V, DC_jack with anti screw and 4Pin dual interface power supply DC_out interface supports + 12V and + 5V power supply 500mA for main board		
Expansion bus	- 1 <b>x</b> Mini PCIe extension (with SIM card holder supporting 4G module) - 1 <b>x</b> PCIe slot		
Working Environment	<ul> <li>Operating temperature / humidity requirements:0~60°C, 10%~90% non condensing;Scalable support -20~70°C, 10%~90% non condensing;</li> <li>Non operating temperature / humidity requirements:25 ~ 75 °C, 5% ~ 95% non condensing</li> </ul>		

Watch Dog	- 255 programmable seconds/min, support timeout interrupt or system reset		
BIOS	- AMI UEFI BIOS		
Operating System	- Win10/Linux , 32 bit / 64 bit		
PCB appearance color	arance color - PCB spray with industrial green paint		
PCB size (LxWxH)	- 170mm X 170mm		

## **1.3 Motherboard structure**



## **1.4 Motherboard IO interface structure**





(This picture is only for reference)

## 1.5 Motherboard Layout



(This picture is only for reference)

## 1.6 I/O Interface



(This picture is only for reference)

- **DCIN:** DC power input interface
- VGA: VGA display interface
- **COM:** Serial port
- **HDMI:** HDMI isplay interface
- USB3.0: USB3.0 interface
- LAN: RJ45 network connector
- USB2.0: USB2.0 interface
- LINE\_OUT: This jack is used to connect to the front left and right channel speakers of the audio system
- MIC\_IN: This jack is used to connect an external microphone

## **Chapter 2 Hardware Setup**

## 2.1 install I / O rear panel

The motherboard provides an I / O back panel. After the rear panel is installed in the chassis, it can block the transmission of radio frequency, protect the internal components from dust and foreign matters, and make the air flow in the chassis keep smooth. Before installing the motherboard into the chassis, install the I / O back panel. First, remove the rear panel attached to the chassis (shake it back and forth several times to remove it), then install the rear panel attached to the motherboard into the chassis, push and press the protective plate to make it firmly in place.

### 2.2 Installing the Mainboard

Most computer chassis have a base with many mounting holes to allow the mainboard to be securely attached, and at the same time, prevent the system from short circuits. There are two ways to attach the mainboard to the chassis base:

(1) with studs

(2) with spacers

Basically, the best way to attach the board is with studs. Only if you are unable to do this should you attach the board with spacers. Line up the holes onthe board with the mounting holes on the chassis. If the holes line up and there are screw holes, you can attach the board with studs. If the holes line up and there are only slots, you can only attach with spacers. Take the



tip of the spacers and insert theminto the slots. After doing this to all the slots, you can slide the board into position aligned with slots. After the board has been positioned, check to make sure everything is OK before putting the chassis back on.

Always power off the computer and unplug the AC power cord before adding or removing any peripheral or component. Failing to do so may cause severe damage to your mainboard and/or peripherals. Plug in the AC power cord only after you have carefully checked everything.

## 2.3 Installing Memory Module

Before starting the installation, please read the following warning messages:

- 1. Make sure your purchased memory specification is supported with the motherboard;
- 2. Before installing or removing memory, make sure that the computer is turned off;
- 3. The memory is designed with fool-proof marker, if you insert with wrong direction, it can not be inserted.

Installing memory:

- 1. Turn off the power and unplug the AC power cord before installing or removing memory.
- 2. Push the motherboard memory card open.
- 3. Check the gap of the memory module, and the location corresponds to the memory slot.

4. After the gap position is pressed, press firmly into the slot. If the installation is successful, there will be a noticeable card sound! (Note: the force of pressing the memory stick should not be too large to avoid damage to the memory)

5. To remove the memory module, push the tabs on each end of the DIMM socket outward and pull out the memory module.

Memory installation illustration (only for reference):





Note: Static will demage the electronic components of computer and memory, when doing above step, you should contact with a grounded metial object to remove the static from your body.

## 2.4 Connect with External Devices

#### 2.4.1 Serial ATA Connector



The Serial ATA connectors can connect to Serial ATA hardware or other corresponding devices when use Serial ATA cable.

Note: choose one of two interfaces between SATA2 and M.2. M.2 is the default.

#### 2.4.2 MINI\_PCIE、PCIE\_X1 slot



Mini\_PCIe is a long card and supports 4G function.

When installing the card, insert the card at an angle of 30 degrees, press it down to the stud, and then fix it with screws.

PCIe\_X1 slot;

#### 2.4.3 M.2-KEYM slot



M.2 slot,M.2 solidstate drive supporting the SATA bus.

When installing the card, insert the card at an angle of 30 degrees, press it down to the stud, and then fix it with screws.

## **Chapter 3 Jumpers & Headers Setup**

## **3.1 Checking Jumper Settings**

- 2-pin jumper: Plug the jumper cap onto both pins will make it CLOSE (SHORT). Remove the cap or plug it on another pins (keep for future use) will activate the jumper.
- 3-pin jumper: Plug the jumper cap onto pin 1~2 or pin 2~3 will make it CLOSE (SHORT). shorted by plugging the jumper cap in.





How to identify the PIN1?

Please check the Motherboard carefully, the PIN1 is marked by "1", white thick line or white triangle.

### 3.2 Jumpers settings

JBAT Jumpers settings

pin	define	
SHORT	CLEAR CMOS	
OPEN	NORMAL	

#### COM1 set up : EDP / LVDS settings JC11/12/13 set up

RS232	RS422	RS485
JC11 (1-2)	JC11 (5-6)	JC11 (3-4)
JC12(1-3)	JC12 (3-5)	JC12 (3-5)
JC12 (2-4)	JC12 (4-6)	JC12 (4-6)
JC13(1-3)	JC13 (3-5)	
JC13 (2-4)	JC13 (4-6)	

#### COM2 set up : JC2 set up

pin	define	
1-2	5V	
2-3	RI#	
5-6	12V	

## COM3 set up : JC32/33 set up

TTL	RS232
JC32 (3-4)	JC32 (2-4)
JC32 (7-8)	JC32 (1-3)
JC33(1-3)	JC32 (6-8)
JC33 (2-4)	JC32 (5-7)
	JC33 (3-5)
	JC33 (4-6)

## JC31 set up

pin	define	
1-2	5V	
2-3	RI#	
5-6	12V	

## JP3/4/5/6 Jumpers settings

LVDS	EDP
JP3 (3-5)	JP3 (1-3)
JP3 (4-6)	JP3(2-4)
JP4 (3-5)	JP4(1-3)
JP4 (4-6)	JP4(2-4)
JP5 (3-5)	JP5(1-3)
JP5 (4-6)	JP5(2-4)
JP6 (3-5)	JP6 (1-3)
JP6 (4-6)	JP6(2-4)

## JLVDS Jumpers (EDP or LVDS, lcdvcc setting)

pin	define	
1-2	3.3V	
2-3	5V	

## 3.3 COM1/2/3/4/5/6 interface



pin	define	pin	define
1	DCD	2	SIN
3	SOUT	4	DTR
5	GND	6	DSR
7	RTS	8	CTS
9	RI		

## 3.4 JPT interface



pin	define	pin	define
1	-STB	2	AFD
3	PRD0	4	ERRORJ
5	PRD1	6	-INIT
7	PRD2	8	-SLIN
9	PRD3	10	GND
11	PRD4	12	GND
13	PRD5	14	GND
15	PR6	16	GND
17	PR7	18	GND
19	ACKJ	20	GND
21	BUSY	22	GND
23	PE	24	GND
25	SLCTJ	26	GND

## 3.5 JGPIO interface



pin	define	pin	define
1	GPIO1	2	GPIO2
3	GPIO3	4	GPIO4
5	GPIO5	6	GPIO6
7	GPIO7	8	GPIO8
9	5V	10	GND

## 3.6 DRAWER interface



pin	define
1	+5V_3V_DRAWER
2	SIO_GP52
3	SIO_GP51
4	GND

## 3.7 CPU\_FAN interface



pin	define
1	GND
2	+5V
3	FUN_IN
4	FUN_OUT

## 3.8 SMB (SM Bus) interface



pin	define
1	+5V_3V_SMBUS
2	SMB_CLK
3	SMB_DATA
4	GND

## 3.9 SYS\_FAN interface



pin	define
1	GND
2	12V
3	FAN_IN
4	FAN_OUT

## 3.10 PS2 interface



pin	define	
1	KB_CLK	
2	KB_DATA	
3	MS_CLK	
4	GND	
5	5V	
6	MS DATA	

## 3.11 DC\_OUT main board power interface



pin	define
1	+12V
2	GND
3	GND
4	+5V

## 3.12 ATX\_12V and HDD\_PWR power interface



ATX-12V power pin definition

pin	define
1	GND
2	GND
3	+12V
4	+12V

HDD\_PWR power pin definition

pin	define
1	+12V
2	GND
3	GND
4	+5V

3.13 VGA\_1 interface



pin	define	pin	define
1	GND	2	V
3	Н	4	GND
5	R	6	GND
7	G	8	GND
9	В	10	GND
11	DAT	12	CK

## 3.14 F\_AUDIOI (Front panel audio interface)



pin	define
1	MIC2_IN_L
2	GND
3	MIC2_IN_R
4	NC
5	F_R
6	F_AUDIO_DET
7	GND
8	空
9	F_L
10	FRONT_IO_SENSE

3.15 SPEAKERS interface(Audio power amplifier, 3W speaker can be connected externally)



pin	define
1	INTSPR-
2	INTSPR+
3	INTSPL-
4	INTSPL+

define

USBPWR3

USBPWR3

USB N7 R

USB\_N0\_R

USB\_P7\_R

USB\_P0\_R

GND

GND

空

GND

## 3.16 USB interface



## 3.17 INVT interface



## 3.18 F\_PANEL interface







pin	define	pin	define
1	LPC_AD3	2	3.3V_TPM
3	LPC_AD3	4	SOC_PMU_PLTRST_N
5	LPC_AD3	6	LPC_FRAME
7	LPC_AD3	8	LPC_CLK
9	LPC_ILB_SERIRQ	10	GND
11	+3.3V_VCC	12	+5V_VCC

## 3.20 EDP interface



pin	define	pin	define
1	BLK_PWR	2	EDP1
3	EDP1	4	GND
5	GND	6	NC
7	NC	8	NC
9	LCD_VCC	10	NC
11	LCD_VCC	12	PWM
13	GND	14	GND
15	VDDEN	16	AUXP_EDP_CN
17	EDP_HDP	18	AUXN_EDP_CN
19	GND	20	GND
21	EDP_P3	22	EDP_N3
23	EDP_P2	24	EDP_N2
25	GND	26	GND
27	EDP_P1	28	EDP_N1
29	EDP_P0	30	EDP_N0

## 3.21 LVDS interface



pin	define	pin	define
1	VDD_PANEL	2	VDD_PANEL
3	VDD_PANEL	4	GND
5	GND	6	GND
7	LVDS_A_DATA0-	8	LVDS_A_DATA0+
9	LVDS_A_DATA1-	10	LVDS_A_DATA1+
11	LVDS_A_DATA2-	12	LVDS_A_DATA2+
13	GND	14	GND
15	LVDS_A_CLK-	16	LVDS_A_CLK+
17	LVDS_A_DATA3-	18	LVDS_A_DATA3+
19	LVDS_B_DATA0-	20	LVDS_B_DATA0-
21	LVDS_B_DATA1-	22	LVDS_B_DATA1+
23	LVDS_B_DATA2-	24	LVDS_B_DATA2+
25	GND	26	GND
27	LVDS_B_CLK-	28	LVDS_B_CLK+
29	LVDS_B_DATA3-	30	LVDS_B_DATA3-

## **Chapter 4 BIOS Setup Utility**

## 4.1 About BIOS Setup

BIOS stands for Basic Input and Output System. It was once called ROM BIOS when it was stored in a Read-Only Memory (ROM) chip. Now manufacturers would like to store BIOS in EEPROM which means Electrically Erasable Programmable Memory. BIOS used in this series of mainboard is stored in EEPROM, and is the first program to run when you turn on your computer. BIOS performs the following functions:

- a. Initializing and testing hardware in your computer (a process called "POST", for Power On Self Test).
- b. Loading and running your operating system.
- c. Helping your operating system and application programs manage your PC hardware by means of a set of routines called BIOS Run-Time Service.
- d. Manage your computer through the SETUP.

CMOS is the memory maintained by a battery. CMOS is used to store the BIOS settings you have selected in BIOS Setup. CMOS also maintains the internal clock. Every time you turn on your computer, the BIOS Looks into CMOS for the settings you have selected and configures your computer accordingly. If the battery runs out of power, the CMOS data will be lost and POST will issue a "CMOS invalid" or "CMOS checksum invalid" message. If this happens, you have to replace the battery and check and configure the BIOS Setup for the new start.

## 4.2 BIOS Setup

This chapter provides informations about the BIOS setup program. Users can configure themselves to optimize the system settings. Those items not be described in detail are rarely-used ones. Do not change the BIOS parameters unless you fully understand its function.

You need to run SETUP when:

- a. Error messages appear and require to enter SETUP during the system self-test
- b. You want to change the default settings according to the customer characteristics.

Note : In order to increase system stability and performance, our engineering staff is constantly improving the BIOS menu. The BIOS setup screens and descriptions illustrated in this manual are for your reference only, and may not completely match with what you see on your screen. This chapter were based mainly on the model, unless specifically stated.

#### 4.2.1 Enter BIOS

After powering up the system, the BIOS message appears on the screen, when the first time or when CMOS setting wrong, there is following message appears on the screen , but if the first BIOS be setuped(or loaded default values) and save, the <DEL> key will be pressed if you will enter BIOS setup menu.



press <F11> to enter fast boot menu.

## Press <F11> to enter boot popup menu

#### 4.2.2 Control Keys

Press <F1> to pop up a small help window that describes the appropriate keys to use and the available options for the highlighted item.

Please check the following table for the function description of each control key.

Control Key(s)	Function Description
← / →	Move cursor left or right to select screens
↑ / ↓	Move cursor up or down to select items
+/ -/PU/PD	To Change option for the selected items
<enter></enter>	To bring up the selected screen
<esc></esc>	Main Menu - Quit and not save changes into CMOS Status Page Setup Menu and Option Page Setup Menu - Exit current page and return to Main Menu
<f1></f1>	General help
<f7></f7>	Previous Values
<f9></f9>	Load Optimal Defaults
<f10></f10>	Save configuration changes and exit setup

### 4.3 Main

Aptio Setup Ut Main Advanced Chipset Se	ility – Copyright (C) 2020 Americ curity Boot Save & Exit	can Megatrends, Inc.
BIOS Information BIOS Vendor Core Version Compliancy BIOS Version BUOS Version Build Date and Time	American Megatrends 5.13 UEFI 2.7; PI 1.6 GLO0A003 05/16/2020 12:18:53	Set the Date. Use Tab to switch between Date elements. Default Ranges: Year: 2005–2040 Months: 1-12 Days: dependent on month
CPU Type Intel(R) Celeron(R) J4005 CP	U @ 2.00GHz	
Memory Information Total Memory Memory Speed	4096 MB	
Memory speed	2400 MHZ	<pre>++: Select Screen  ↑↓: Select Item</pre>
System Date System Time	[PP1 01/12/2018] [20:01:35]	Enter: Select +/−: Change Opt. E1: General Help
Access Level	Administrator	F7: Previous Values F9: Optimized Defaults F10: Save & Exit ESC: Exit
Version 2.19.	1268. Copyright (C) 2020 Americar	n Megatrends, Inc.

- BIOS Information(BIOS related information)
- System Date (system date setting)

Set the date of the computer in the format of "week, month / day / year".

• System Time(system time setting)

The time format is < hour > < minute > < second >.

### 4.4 Advanced



#### • Super IO Configuration

This is super IO setting information, including COM port interrupt number and address setting.

• Hardware Monitor

This item is to monitor the current hardware status, including CPU temperature, voltage and other system status.

- LPC to Serial Port Configuration This is the system advanced IO chip parameter.
- **Power Management Configuration** This is the power management configuration.
- Serial Port Console Redirection This is the serial port redirection setting.
- **Onboard Device Configuration** This is the main board device control parameter.
- **CPU Configuration** This is the processor setting.
- **CSM Configuration** This is a CSM configuration.
- EMMC Configuration This is EMMC configuration.
- USB Configuration This is USB information and control options.

#### • SATA Configuration

This is to set up the SATA device.

## 4.5 Chipset

Aptio Setup Utility – Main Advanced Chipset Security	Copyright (C) 2020 American Boot Save & Exit	Megatrends, Inc.
Aperture Size DVHT Pre-Allocated DVHT Total Gfx Mem Cd Clock Frequency Primary IGFX Boot Display Secondary IGFX Boot Display	(256M8) (64H) (256M) (316.8 HHz] (HOHI] (VGA)	Select DVMT5.0 Total Graphic Memory slze used by the Internal Graphics Device
		<pre>++: Select Screen 14: select Item Enter: Select +/-: Change Opt. F1: General Help F7: Previous Values F9: Optimized Defaults F10: Save &amp; Exit ESC: Exit</pre>
Version 2.19.1268. Co	pyright (C) 2020 American M	egatrends, Inc.

#### • Aperture Size

This is the selection of aperture size. Optional: 128M, 256M, 512M.

#### DVMT Pre-Allocated

Select DVMT5.0 based on the allocated (fixed) graphics memory size used by the internal graphics device.

Optional: 64M, 96M, 128M, 160M, 192M, 224M, 256M, 288M, 320M, 352M, 384M, 416M, 448M,480M, 512M.

#### • DVMT Total Gfx Men

Select the total graphics memory size of DVMT5.0 used by the internal graphics device.

Optional: 128M, 256M, MAX.

#### • Cd Clock Frequency

This is the highest Cd Clock Frequency supported by the selected platform. Optional: 158.4MHz, 316.8MHz, 79.2MHz.

#### • Primary IGFX Boot Display

Select the on-board video card main display device. Optional: VGA, LVDS, HDMI.

#### • Secondary IGFX Boot Disp

This is the selection of auxiliary IGFX boot display. Optional: Disabled, VGA, HDMI.

## 4.6 Security



#### • Administrator Password

Setting this option is used to set the system administrator password. There are the following steps:

1. Select the Administrator Password setting item and press the < Enter > key.

2. In the "Create New Password" dialog box, enter 3-20 characters or digital password to be set, press the < Enter > key after input, the "Confirm Password" dialog box will appear, and enter the password again to confirm the password is correct. If the prompt "Invalid Password!" indicates that the two passwords do not match, please re-enter. To clear the system administrator password, select administrator password. When the enter current password dialog box appears, enter the old password and the create new password < Enter > password will be cleared.

#### • User Password Boot Prompt

Verify the administrator or user password before booting to the operating system. Optional: Enabled, Disabled.

### 4.7 Boot



#### • Setup Prompt Timeout

Set the dwell time of power on interface.

#### Bootup Numlock State

Set the status of NumLock after system startup. When it is set to On, NumLock will be turned on after the system is started key is valid. When it is set to Off, NumLock is closed after the system is started, and the keypad direction key is valid.

Optional: On, Off.

#### • Full Loge Display

Enables or disables the icon.

Optional: Enabled, Disabled.

#### OS Selection

This is the selection of the target operating system. Options: Windows, Android, MSDDS.

#### Boot Option #1/2/3

This is to set the system startup order.

Options: PO: Hoodisk SSD, UEFI General UDisk 5.00, Partition 1, General UDisk 5.00, Disabled.

## 4.8 Save & Exit



• Save Changes and Reset

Save the changes and restart the system.

- **Discard Changes and Reset** Do not save changes and restart the system.
- Load optimized Default

Restore / load default values for all settings options.

• Launch EFI Shell from filesystem device Try to load and run shellx64.efi from the file system.

## **Chapter 5 Driver Installation**

Insert DVD into your DVD-ROM drive and the menu should appear as below. If the menu does not appear, double-click My Computer / double-click DVD-ROM drive or click Start / click Run / type X:\AUTORUN.EXE (assuming X is your DVD-ROM drive).

Mainboard	Driver Install
Chipset Driver Lan Driver Audio Driver	Management Engine Graphics Driver USB3.0 ( For Win7 only )
	Browse Disc

Only for reference Please click on the options ,and install relevantly corresponding driver

## Chapter 6 WDT Programming Guide

```
6.1 Programming guidance documentation
   This document contains the secondary development instructions and software sample
    code of watchdog of mainboard aiot0-h110.
    Note: this programming guide is applicable to WDT projects with corresponding
    signals from ite8786 super io.
6.1.1 IO function definitions involved
   Under Linux system, several header files need to be included:
   #include <stdio.h>
   #include <errno.h>
   #include <sys/io.h>
   //Read 1 Byte from Io address
       /*
          Name: IoRead8
          Input: GpioAddress
                                         - GPIO base address
          Output: GPIO port read value
          Description:
       */
   unsigned char IoRead8(unsigned short GpioAddress) {
       unsigned char ret;
       ret = iopl(3);
       if (ret)
       {
            printf("ret = %d n", ret);
            printf("errno = %d\n", errno);
            printf("error for iopln");
            return 1:
       return inb(GpioAddress);
```

#### //Write 1 byte to IO address

```
unsigned char IoWrite8(unsigned short GpioAddress, unsigned char Data) {
    unsigned char ret;
    ret = iopl(3);
    if (ret)
    {
        printf("ret = %d\n", ret);
        printf("erron = %d\n", errno);
        printf("error for iopl\n");
        return 1;
    }
    outb(Data, GpioAddress);
    return 0;
}
```

#### 6.2 System watchdog programming

6.2.1 WatchDog programming considerations

1. It is recommended to set the Watchdog timeout time to no less than 5 seconds to prevent the software system from resetting the Watchdog in time when the CPU load is high, causing unnecessary system restart.

2. When shutting down the operating system and application software, it is recommended to first shut down the Watchdog or set a long time Watchdog timeout to avoid the Watchdog reset when the operating system is not completely shut down.

#### 6.2.2 Programming examples

Note: this programming guide is applicable to WDT projects with corresponding signals from IT8786 Super IO

**WDT** programming examples

#define SuperIO\_Index\_Port 0x2E
#define SuperIO\_Data\_Port 0x2F
#define WDT\_LDN\_DEVICE

0x07

#### //a. Enter Super IO Configuration Mode

IoWrite8(SuperIO\_Index\_Portt, 0x87); IoWrite8(SuperIO\_Index\_Port, 0x01); IoWrite8(SuperIO\_Index\_Port, 0x55); IoWrite8(SuperIO\_Index\_Port, 0x55);

#### //b. Select logical device GPIO, LDN 7

IoWrite8(SuperIO\_Index\_Port, 0x07); IoWrite8(SuperIO\_Data\_Port , WDT\_LDN\_DEVICE);

#### //c. Clear WDT status register

IoWrite8(SuperI0\_Index\_Port, 0x71);
Data8=IoWrite8(SuperI0\_Data\_Port, ((IoRead8(SuperI0\_Data\_Port)& ~0x10)|
0x01));

#### //d. Set the working mode of WDT, time in seconds or minutes

IoWrite8(SuperIO\_Index\_Port, 0x72); Data8 = IoRead8(SuperIO\_Data\_Port); Data8 = (Data8&0xc0)|0xc0; //WDT计数模式以秒为单位

// Data8 = (Data8&0x40) |0x40; /\*WDT count mode in minutes \*/

IoWrite8(SuperIO\_Data\_Port, Data8);

//e.Start WDT, WDT range is 0-255, if 0, WDT is in stop mode, not 0 The value of is WDT counter value. After this time, the system will restart automatically.

UINT8 Timer; IoWrite8(SuperI0\_Index\_Port,0x73); IoWrite8(SuperI0\_Data\_Port, Timer); //Low byte timer is user-defined 1 to 255
Any value between.
IoWrite8(SuperI0\_Index\_Port, 0x74);
IoWrite8(SuperI0\_Data\_Port, 0x00); //High byte timer is generally set to 0x00
because Low byte timer configuration is enough.

#### //f. Disable WDT function

IoWrite8(SuperIO\_Index\_Port, 0x73); IoWrite8(SuperIO\_Data\_Por, 0x00);

#### //g. Reset (dog feed) WDT timer

 $\label{eq:lowrites} $$ IoWrite8(SuperIO_Index_Port, 0x73);$$ IoWrite8(SuperIO_Data_Port, Timer); //Timer for user-defined between 1 and 255 any value. $$$ 

#### //h. Exit Super IO Configuration Mode

IoWrite8(SuperIO\_Index\_Port, 0x02);

 $\label{eq:lowrites} IoWrite8(SuperI0_Index_Port, 0x02);//After exiting super IO configuration mode, //Each logical device of superio can no longer be accessed unless it enters configuration mode again.$ 

## Chapter 7 GPIO programming guidance

#### 7.1 Programming guidance documentation

This document contains the GPIO secondary development instructions and software sample code of the aiot7-gml motherboard. Note: this programming guide is applicable to GPIO projects with corresponding signals from Intel PCH, and specific GPIO used by different boards the signals may be different. In addition, note that the memory address to access GPIO is the physical memory address.

 $7.\,1.\,1$  IO function definition involved (take definition in BIOS as an example)

/\*\*

Reads a 32-bit MMIO register.

Reads the 32-bit MMIO register specified by Address. The 32-bit read value is  $% \left[ 1 + 1 \right] \left[ \left( 1 + 1 \right) \left[ \left( 1 + 1 \right) \right] \left[ \left( 1 + 1 \right) \right] \left[ \left( 1 + 1 \right) \left[ \left( 1 + 1 \right) \right] \left[ \left( 1 + 1 \right) \left[ \left( 1 + 1 \right) \right] \left[ \left( 1 + 1 \right) \left[ \left( 1 + 1 \right)$ 

returned. This function must guarantee that all MMIO read and write operations are serialized.

If Address is not aligned on a 32-bit boundary, then ASSERT().

If 32-bit MMIO register operations are not supported, then ASSERT().

@param Address The MMIO register to read.

@return The value read.

```
**/
UINT32
EFIAPI
MmioRead32 (
    IN UINTN Address
    )
    {
        // // Make sure Address is aligned on a 32-bit boundary.
```

```
11
  ASSERT ((Address & 3) == 0):
  return (UINT32)MmioReadWorker (Address, EfiCpuIoWidthUint32);
}
/**
 Writes a 32-bit MMIO register.
 Writes the 32-bit MMIO register specified by Address with the value
specified
  by Value and returns Value. This function must guarantee that all MMIO
read
  and write operations are serialized.
  If Address is not aligned on a 32-bit boundary, then ASSERT().
  If 32-bit MMIO register operations are not supported, then ASSERT().
  @param Address The MMIO register to write.
  @param Value The value to write to the MMIO register.
 **/
 UINT32
 EFIAPI
 MmioWrite32 (
  IN
          UINTN
                                     Address.
  IN
                                     Value
          UINT32
   )
  // Make sure Address is aligned on a 32-bit boundary.
  ASSERT ((Address & 3) == 0);
  return (UINT32) MmioWriteWorker (Address, EfiCpuIoWidthUint32, Value);
 }
  /**
  Reads memory-mapped registers in the EFI system memory space.
  Reads the MMIO registers specified by Address with registers width
 specified by Width.
   The read value is returned. If such operations are not supported, then
  ASSERT().
   This function must guarantee that all MMIO read and write operations
  are serialized.
```

@param Address The MMIO register to read.

```
The caller is responsible for aligning the Address
if required.
 @param Width
                        The width of the I/O operation.
 @return Data read from registers in the EFI system memory space.
**/
          UINT64
          EFIAPI
          MmioReadWorker (
          UINTN
IN
                                     Address.
IN
         EFI CPU IO PROTOCOL WIDTH Width
)
EFI STATUS Status;
UINT64
             Data:
Status = mCpuIo->Mem. Read (mCpuIo, Width, Address, 1, &Data);
ASSERT EFI ERROR (Status);
return Data;
/**
 Writes memory-mapped registers in the EFI system memory space.
 Writes the MMIO registers specified by Address with registers width
 and value specified by Width
 and Data respectively. Data is returned. If such operations are not
 supported, then ASSERT().
 This function must guarantee that all MMIO read and write operations
 are serialized.
 @param Address
                        The MMIO register to read.
                      The caller is responsible for aligning the
 Address
if required.
 @param Width
                        The width of the I/0 operation.
 @param Data
                        The value to write to the I/0 port.
 @return Data read from registers in the EFI system memory space.
 **/
UINT64
EFIAPI
MmioWriteWorker (
  ΤN
           UINTN
                                      Address,
  IN
           EFI CPU IO PROTOCOL WIDTH Width,
  IN
           UINT64
                                      Data
   )
```

```
EFI_STATUS Status;
Status = mCpuIo->Mem.Write (mCpuIo, Width, Address, 1, &Data);
ASSERT_EFI_ERROR (Status);
return Data;
}
```

## Ordering Information

Product number	chipset	RAM	display	storage	USB3	USB2	COM	LAN	SSD	PCIe
AIoT7-GML1	J4005	1	3	2SATA	2	8	6	1GbE	1	2
AIoT7-GML2	J4005	1	3	2SATA	2	8	6	2GbE	1	2
AIoT7-GML3	J4105	1	3	2SATA	2	8	6	2GbE	1	2

### Toxic and hazardous substances or elements logo:



Carried out under the Ministry of Information Industry of the People's Republic of China released the <Electronic Information Products Pollution Control Management Measures >> SJ/T11364-2006 standard requirements, pollution control identification of the products and toxic and hazardous substances or elements of identity are described below:

Toxic and hazardous substances or elements logo:

The names and contents of toxic and hazardous substances or elements in the product

Dart Name		Toxic and	d hazardous	substances o	r elements		
Part Name	(Pb)	(Hg)	(Cd)	(Cr(VI))	(PBB)	(PBDE)	
PCB Board	×	0	0	0	0	0	
Structure	0	0	0	0	0	0	
Chipset	0	0	0	0	0	0	
Connector	0	0	0	0	0	0	
Passive electronic components	0	0	0	0	0	0	
Weld metal	0	0	0	0	0	0	
Wire	×	0	0	0	0	0	
Help welding,thermal grease, labels and other supplies	0	0	0	0	0	0	
<ul> <li>O : Indicates that this toxic and hazardous substance content in all of the components of homogeneous material provisions of the SJ/T11363-2006 standard limited requirement.</li> <li>X : Indicates that this toxic or hazardous substances at least in the part of a homogeneous material content than SJ/T11363-2006 standards limited requirement.</li> </ul>							

Note : X means the location of the lead content exceeds the limit requirement of ST/T11363-2006's standard, but in line with the European Union RoHS Directive exemption clause.