2I640HL

Intel Elkhart Lake ATOM® x6413E SoC CPU,
DDR4 SODIMM,
3 x LAN / HDMI / USB / COM / M.2

All-In-One SBC

Intel Elkhart Lake ATOM® x6413E SoC CPU

Onboard Hailo-8™ edge AI processor

2 x HDMI, eDP,

2 x M.2, 3 x LAN, 1 x Nano SIM

USB, COM, Wide Range DC-IN 9~24V

CAUTION

RISK OF EXPLOSION IF BATTERY IS REPLACED BY AN INCORRECT TYPE.
DISPOSE OF USED BATTERIES ACCORDING TO THE INSTRUCTIONS

NO. 2I640HL
Release date: SEP. 15. 2023

Contents

User Manual edition 0.1, SEP. 15. 2023

Warning !

1. Battery
   Batteries on board are consumables.
   The life time of them are not guaranteed.

2. Fanless solution with HDD
   The specification & limitation of HDD should be considered carefully when the fanless solution is implemented.

3. We will not give further notification in case of changes of product information and manual.

4. SATA interface does not support Hot SWAP function.

5. There might be a 20% inaccuracy of WDT at room temperature.

6. Please make sure the voltage specification meets the requirement of equipment before plugging in.

7. There are two types of SSD, commercial grade and industrial grade, which provide different read/write speed performance, operation temperature and life cycle. Please contact sales for further information before making orders.

8. Caution! Please notice that the heat dissipation problem could cause the MB system unstable. Please deal with heat dissipation properly when buying single MB set.

9. Please avoid approaching the heat sink area to prevent users from being scalded with fanless products.

10. If users repair, modify or destroy any component of product unauthorizedly, We will not take responsibility or provide warranty anymore.

11. DO NOT apply any other material which may reduce cooling performance onto the thermal pad.

12. It is important to install a system fan toward the CPU to decrease the possibility of overheating / system hanging up issues, or customer is suggested to have a fine cooling system to dissipate heat from CPU.

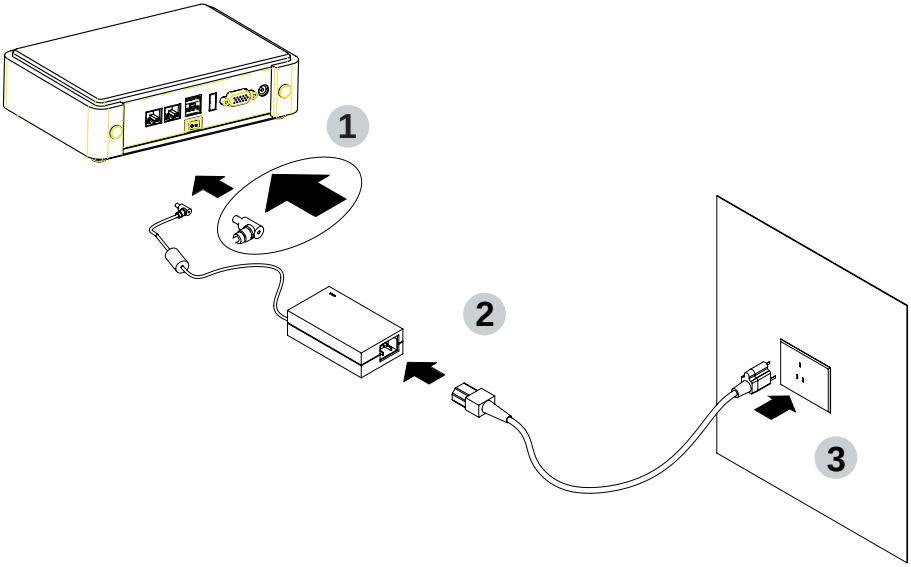
✱ Hardware Notice Guide

1. Before linking power supply with the motherboard, please attach DC-in adapter to the motherboard first. Then plug the adapter power to AC outlet.
   Always shut down the computer normally before you move the system unit or remove the power supply from the motherboard. Please unplug the DC-in adapter first and then unplug the adapter from the AC outlet.
   Please refer photo 1 as standard procedures.
2. In case of using DIRECT DC-in (without adapter), please check the allowed range for voltage & current of cables. And make sure you have the safety protection for outer issues such as short/broken circuit, overvoltage, surge, lightning strike.
3. In case of using DC-out to an external device, please make sure its voltage and current comply with the motherboard specification.
4. The total power consumption is determined by various conditions (CPU/motherboard type, device, application, etc.). Be cautious to the power cable you use for the system, one with UL standard will be highly recommended.
5. It's highly possible to burn out the CPU if you change / modify any parts of the CPU cooler.
6. Please wear wrist strap and attach it to a metal part of the system unit before handling a component. You can also touch an object which is ground connected or attached with metal surface if you don't have wrist strap.
7. Please be careful to handle & don't touch the sharp-pointed components on the bottom of PCBA.
8. Remove or change any components form the motherboard will VOID the warranty of the motherboard.
9. Before you install / remove any components or even make any jumper setting on the motherboard, please make sure to disconnect the power supply first. (follow the aforementioned instruction guide)
10. "POWERON after PWR-Fail" function must be used carefully as below:
    When the DC power adaptor runs out of power, unplug it from the DC current; Once power returns, plug it back after 5 seconds.
    If there is a power outage, unplug it from the AC current, once power returns, plug it back after 30 seconds. Otherwise it will cause system locked or made a severe damage.

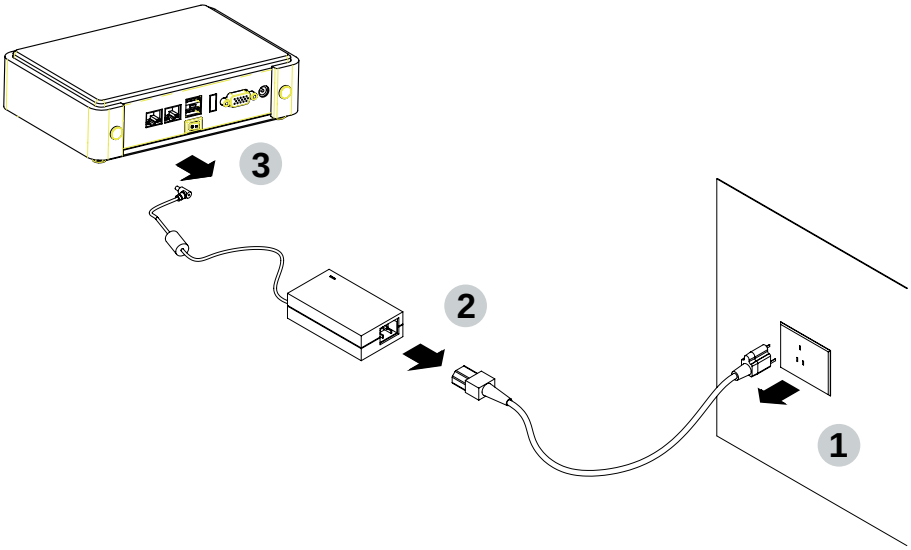
Remark 1:
Always insert / unplug the DC-in horizontally & directly to / from the motherboard. DO NOT twist, it is designed to fit snugly.
Moreover, erratic pull / push action might cause an unpredictable damage to the component & system unit.

Photo 1

Insert



Unplug



Chapter-1

General Information

The 2I640HL is a 2.5″ (122 x 100 mm) SBC: 2I640HL is with Intel Atom® x6000E Series and Celeron® (formerly Elkhart Lake) Series processors & onboard Hailo-8 AI Accelerator. 2I640HL integrated 3 x 2.5 GbE LAN, 6 x USB, 2 x COM Port, 2 x HDMI (or DP) display interface and 2 x M.2 expansion slot offers the ideal platform for graphics performance with integrated IoT features, real-time performance, manageability, and security.
In addition, via wide range 9-24V DC input and onboard Hailo-8™ AI acceleration, 2I640HL can easily be adopted to space limited AI applications. 2I640HL with onboard Hailo-8™ AI accelerator provides maximum power of Edge AI and helping customers speed up time to market and run complicated deep learning and machine vision applications with lower power consumption.

The 2I640HL supports high-speed data transfer interfaces such as PCIe gen3, USB 3.0, and SATA 6 Gb/s (SATA III) for SATA port and supports 2 serial port RS232 / RS485 / RS422 jumper free auto switch by BIOS. It supports 1 port of USB 3.0, 5 port of USB 2.0. The expandable interfaces include 1 M.2 B-Key for PCIe x 2 and USB 2.0 interface, and 1 M.2 B-Key for SATA / PCIe auto switch and USB 2.0 interface with nano SIM.

The embedded motherboard 2I640HL is specially designed with Wide-Range Voltage DC in (9~24V) for widely varying input voltage requirement. 2I640HL incorporating with AI accelerator to support up to 26 Tera-Operations Per Second (TOPS), 2I640HL provides high-performance computing capability to fulfill the needs of AI applications. 2I640HL is a trusted solution for Machine Vision, In-vehicle Computing, ARM / AGV and Intelligent Control applications, or any Industry 4.0 / AIoT applications It supports with three 2.5Gbps Ethernet for seamless broadband connectivity. With Wake-On LAN function and the PXE function in BIOS, these are perfect control boards for networking devices.

1-1 Major Feature

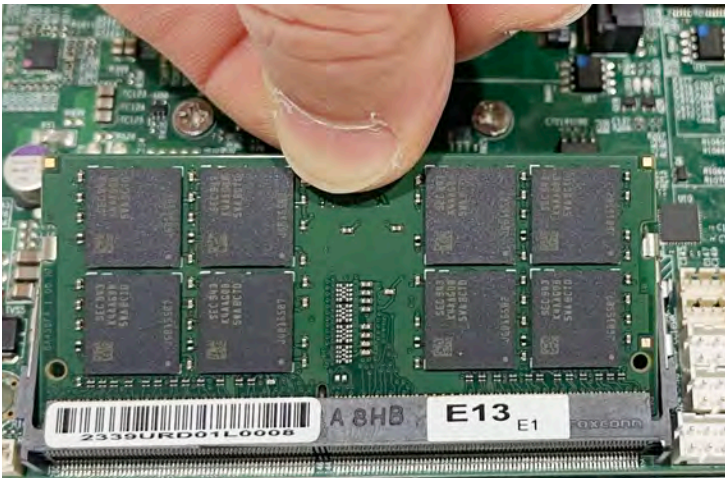
1. Intel® Atom x6413E Processor 1.5GHz / 2.7GHz (Quad Core),
   Intel® Celeron Processor J6412 2.0GHz / 2.6GHz (Quad Core)
2. Intel® UHD Graphics for 10th Gen Intel® Atom x6413E 500MHz / 750MHz,
   Intel® Celeron J6412 400MHz / 800MHz
3. Support HDMI 1.4b up to 3840 x 2160at 60Hz and eDP 1.3 2 Lanes up
   to 1920 x 1080 at 60Hz.
4. Hailo-8™ AI Processor, up to 26 TOPS
5. DDR4 SODIMM slot x 1, up to 32GB
6. Support 3 x 2.5G Intel LAN port.
7. Support 2 x RS232 selectable to RS485 / RS422 by BIOS
8. 1 x USB 3.0 and 5 x USB 2.0
9. Support extended 1 x M.2 B-Key for PCIe x 2 and USB 2.0 and 1 x M.2 B-Key
   for PCIe / SATA auto switch and USB 2.0 interface with Nano SIM.
10. Hardware digital Input & Output, 4 x DI / 4 x DO, Hardware Watch Dog Timer,
    0~255 sec programmable
11. Wide Range DC IN +9V~24V

1-2 Specification

1. **SOC:** Intel® Atom x6413E Processor 1.5GHz / 2.7GHz (Quad Core),
   Intel® Celeron Processor J6412 2.0GHz / 2.6GHz (Quad Core)
2. **Memory:** DDR4 SODIMM slot x 1, up to 32GB
3. **Graphics:** Intel® UHD Graphics for 10th Gen Intel® Atom x6413E 500MHz / 750MHz,
   Intel® Celeron J6412 400MHz/800MHz, HDMI 1.4b up to 3840 x 2160at 60Hz
   and eDP 1.3 2 Lanes up to 1920 x 1080 at 60Hz.
4. **AI Accelerator:** Hailo-8™ AI Processor, up to 26 TOPS
5. **LAN:** 3 Intel I226IT LAN chipset with 2.5 Gbps
6. **I/O Chip:** Switch chipset for 2 port RS232 / RS422 / RS485 selected by BIOS.
7. **USB:** 1 x USB 3.0, 5 x USB 2.0
8. **WDT/DIO:** Hardware digital Input & Output, 4 x DI / 4 x DO (Option) / Hardware
   Watch Dog Timer, 0~255 sec programmable
9. **Expansion interface:** one M.2 B-Key for PCIe x 2 and USB 2.0 interface,
   one M.2 B-key for PCIe / SATA auto switch and USB 2.0 interface with Nano SIM
10. **BIOS:** Insyde UEFI BIOS
11. **Dimension:** 122 x 100 mm
12. **Power:** On board DC +9~24V

1-3 Installing the SO-DIMM

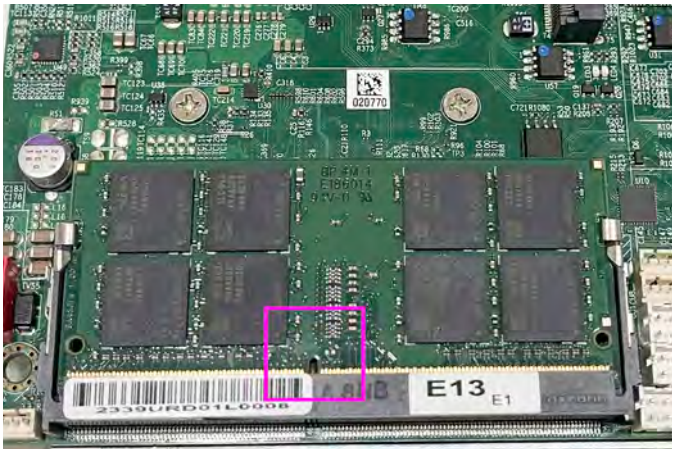
1. Align the SO-DIMM with the connector at a 45 degree angle.



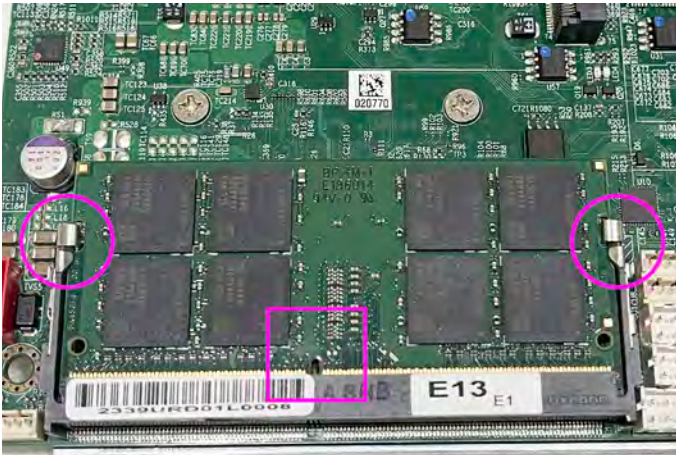
2. Press the SO-DIMM into the connector until you hear a click.

Notices:

1.The connectors are designed to ensure the correct insertion. If you feel resistance, check t h e connectors & golden finger direction, and realign the card.



2. Make sure the retaining clips (on two sides of the slot) lock onto the notches of the card firmly.



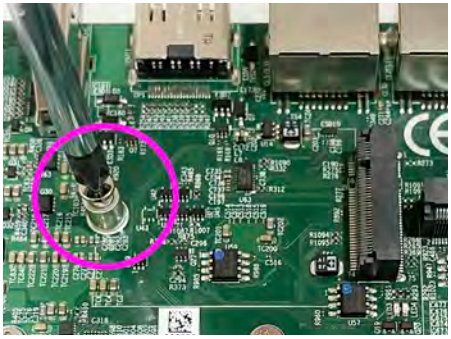
1-3-1 Removing the SO-DIMM

1. Release the SO-DIMM by pulling outward the two retaining clips and the SO-DIMM pops up slightly.

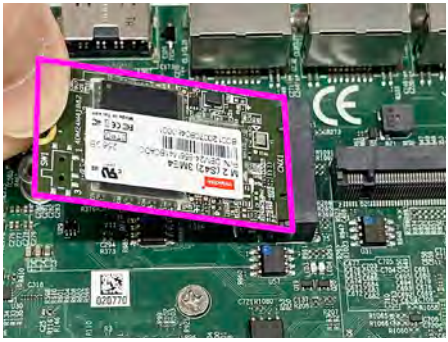
2. Lift the SO-DIMM out of its connector carefully.

1-4 Directions for installing the M.2 B Key Mini Card

1. Unscrew the screw on the board



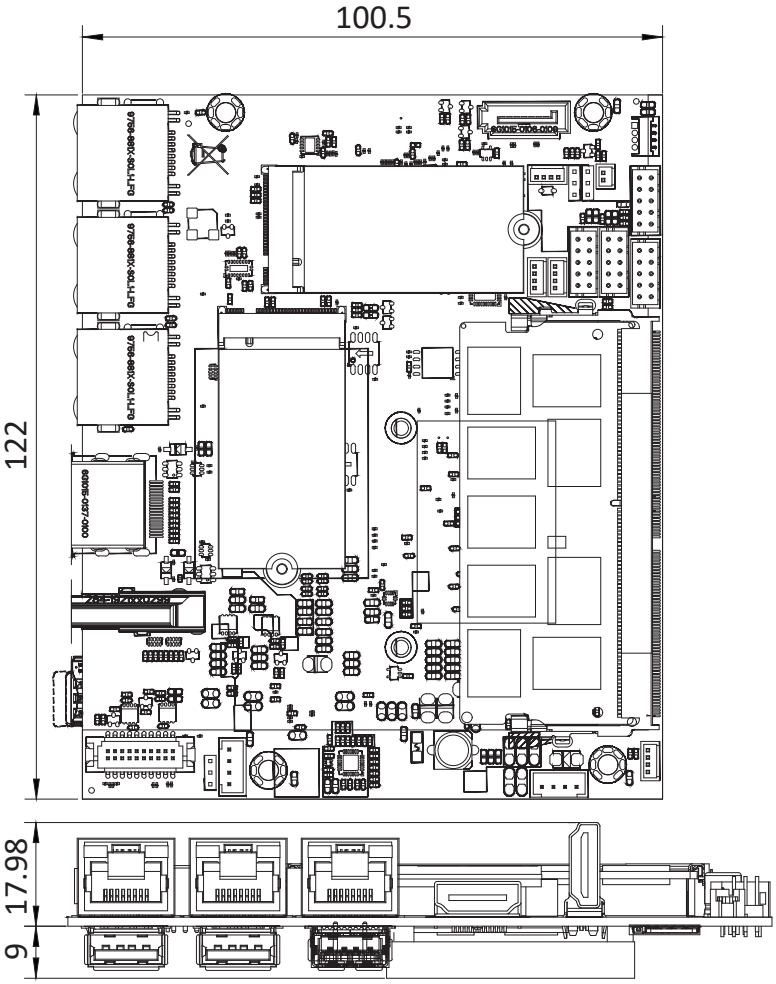
2. Plug in the Mini Card in a 45 angle



3. Gently push down the Mini Card and screw the screw back.

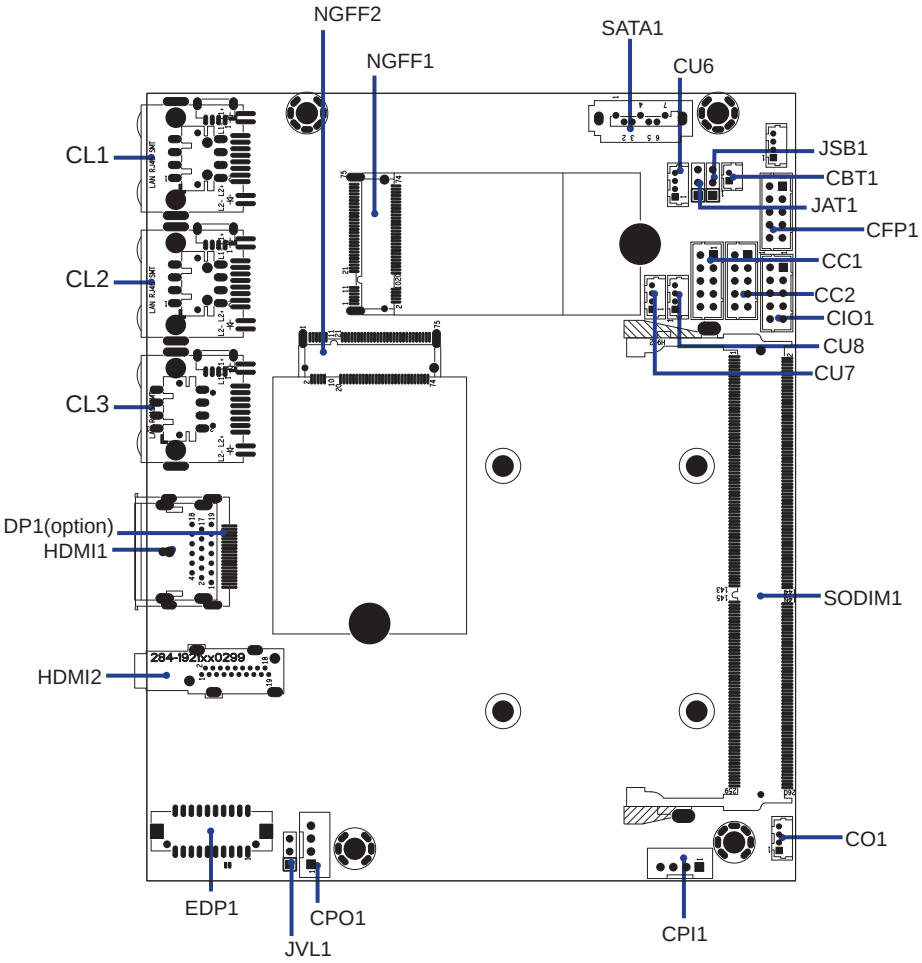
Chapter-2

2-1 Dimension-2I640HL



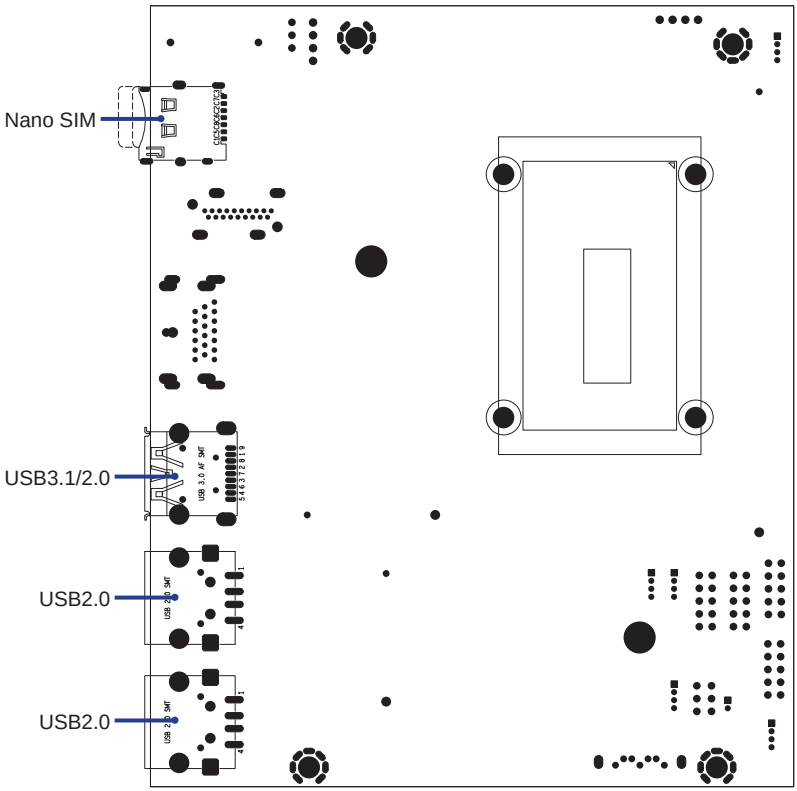
2-2 Layout-2I640HL-Connector and Jumper

TOP



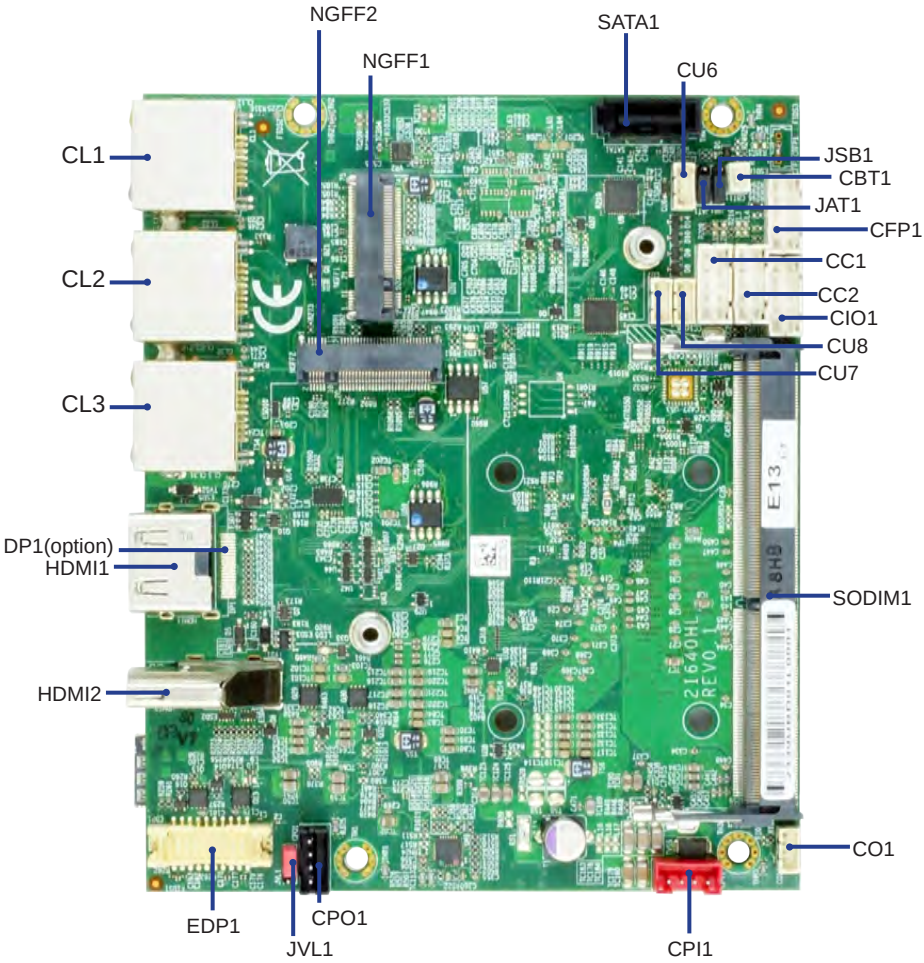
2-2-1 Layout-2I640HL-Connector and Jumper

BOT



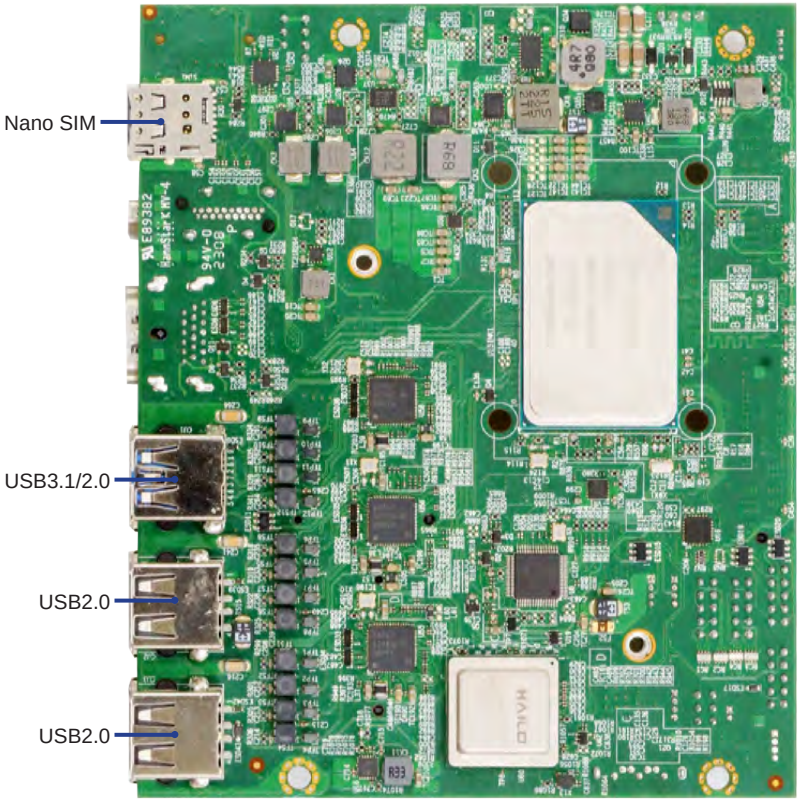
2-3 Diagram- 2I640HL

TOP

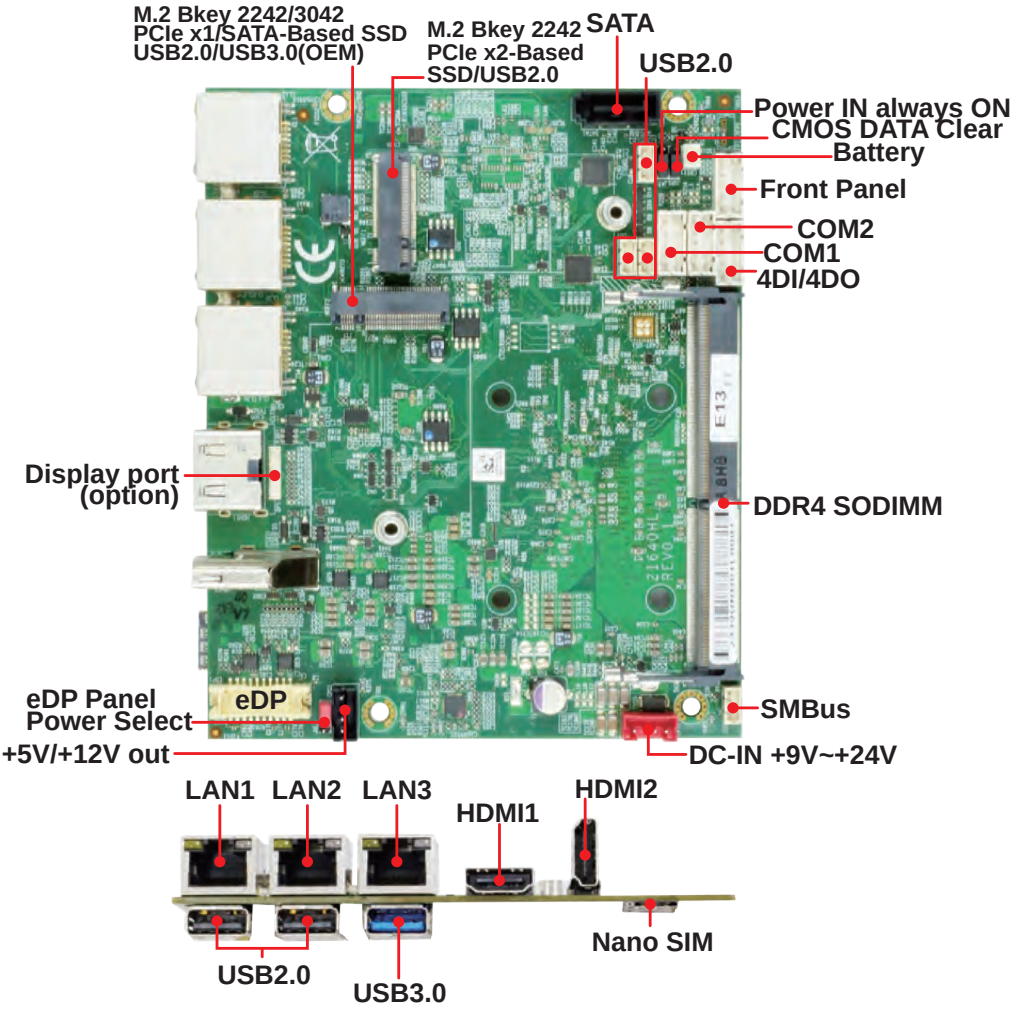


2-3-1 Diagram- 2I640HL

BOT



2-3-2 Function MAP- 2I614HL



2-4 List of Jumpers

JSB1: CMOS DATA Clear
JAT1: Power in always ON function
JVL1: eDP panel power select

2-5 Jumper Setting Description

A jumper is ON as a closed circuit with a plastic cap covering two pins. A jumper is OFF as an open circuit without the plastic cap. Some jumpers have three pins, labeled 1, 2, and 3. You could connect either pin 1 and 2 or 2 and 3.
The below figure 2.2 shows the examples of different jumper settings in this manual.

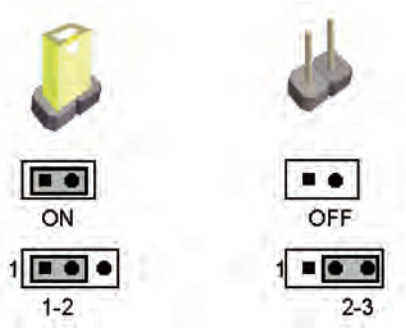


Figure 2.2

All jumpers already have its default setting with the plastic cap inserted as ON, or without the plastic cap as OFF. The default setting may be referred in this manual with a " * " symbol .

2-6 JSB1: CMOS DATA Clear

A battery must be used to retain the motherboard configuration in CMOS RAM.

Close Pin 1 and Pin 2 of JSB1 to store the CMOS data.

To clear the CMOS data, please follow the steps as below:

1. Turn off the system and unplug the AC power.
2. Make sure there is no AC & DC power connect to the system or MB.
3. Close pin 2-3 of JSB1 for a few seconds.
4. Return to default setting by close pin 1-2
5. Connect DC IN power cable back to DC IN Power connector

Note: The Panel resolution & Power failed state will not be restored after CMOS data clear. If your system MUST restore the Panel resolution & Power failed state to default settings, please update the BIOS.

The default settings of Panel resolution is 1024 x 768, power failed state is Keep last state (Last state).

JSB1	DESCRIPTION
*1-2	Normal set
2-3	CMOS data clear

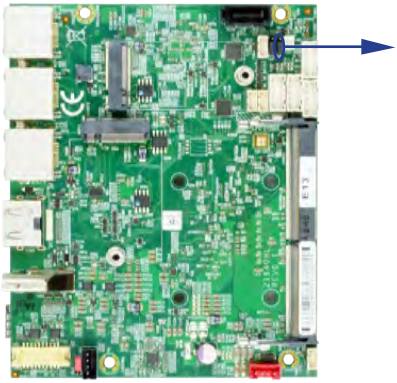
Note: Normal work is open jumper

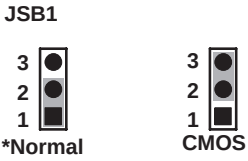
Note: Do not clear CMOS unless

1. Troubleshooting

2. Forget password

3. You fail over-clocking system

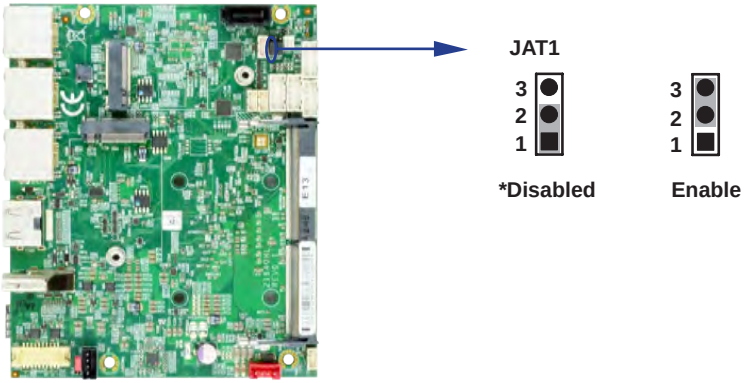




2-7 JAT1: Power in always ON function

JAT1	DESCRIPION
*1-2	Disabled
2-3	Enable

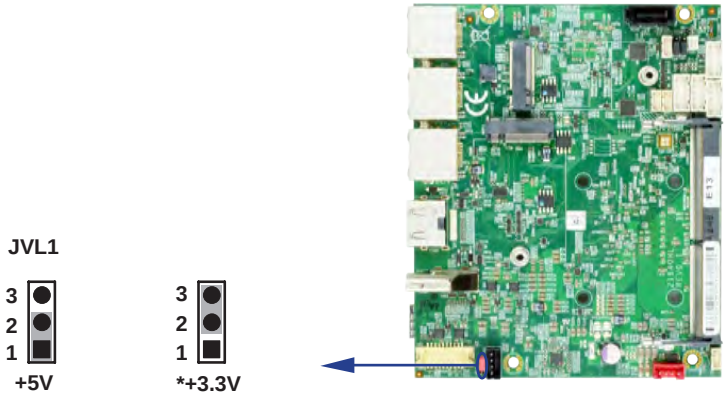
NOTE: Power always on function default is disabled.



2-8 JVL1: eDP panel power select

JVL1	DESCRIPION
1-2	+5V
*2-3	+3.3V

NOTE: Attention! Check Device Power in spec.



Chapter-3

Connection

This chapter provides all necessary information of the peripheral's connections, switches and indicators. Always power off the board before you install the peripherals.

3-1 List of Connectors

CBT1:	CMOS Battery in 1x2 pin (1.25mm) wafer
CU1:	USB 3.1 type A connector
CU2:	USB 2.0 type A connector
CU3:	USB 2.0 type A connector
CU6:	USB 2.0 port 1x4 pin (1.25mm) wafer
CU7:	USB 2.0 port 1x4 pin (1.25mm) wafer
CU8:	USB 2.0 port 1x4 pin (1.25mm) wafer
CL1:	RJ45 LAN connector
CL2:	RJ45 LAN connector
CL3:	RJ45 LAN connector
CL11:	LAN port 2x4 pin (2.0mm) wafer (option)
CL21:	LAN port 2x4 pin (2.0mm) wafer (option)
CL31:	LAN port 2x4 pin (2.0mm) wafer (option)
CL12:	LAN LED 1x4 pin (1.0mm) wafer (option)
CL22:	LAN LED 1x4 pin (1.0mm) wafer (option)
CL32:	LAN LED 1x4 pin (1.0mm) wafer (option)
CC1:	COM1 2x5 pin (2.0mm) wafer
CC2:	COM2 2x5 pin (2.0mm) wafer
CFP1:	Front Panel connector 2x5 pin (2.0mm) wafer
CIO1:	4DI / 4DO 2x5 pin (2.0mm) wafer
CO1:	SMBus 1x4 pin (1.25mm) wafer
CPI1:	DC-IN 1x4 pin (2.0mm) Red wafer
CPO1:	+12V / +5V output 1x4 pin (2.0mm) Black wafer
EDP1:	eDP 2x10 pin (1.25mm) wafer
SIM1:	Nano SIM card socket
SODIM1:	DDR4 SODIMM H: 9.2mm

NGFF1: M.2 B key 2242 H=8.5 sockets 75pin

NGFF2: M.2 B key 2242 / 3042 H=8.5 sockets 75pin

HDMI1: HDMI typeA connector

HDMI2: HDMI typeA connector 90º

DP1: DisplayPort connector (option)

SATA1: SATA connector 7 pin

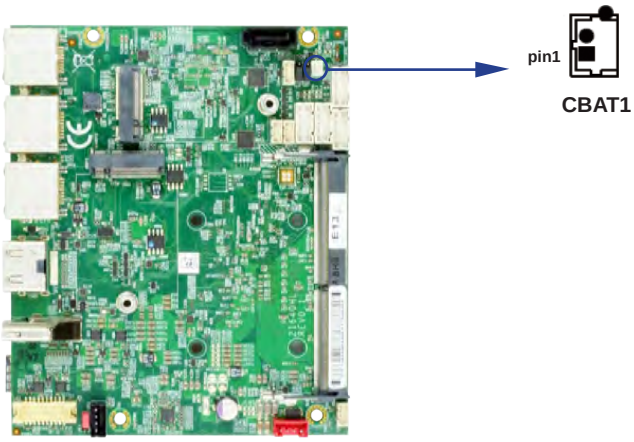
CRFP1: Antenna control 1x4 pin (1.25mm) wafer (OEM)

3-2 CMOS battery connector

- **CBT1: CMOS Battery in 1x2 pin (1.25mm) wafer**

PIN NO.	DESCRIPION
1	Battery in (GND)
2	Battery in (+3V)

NOTE: CBT1 for external connector can extend battery capacity.



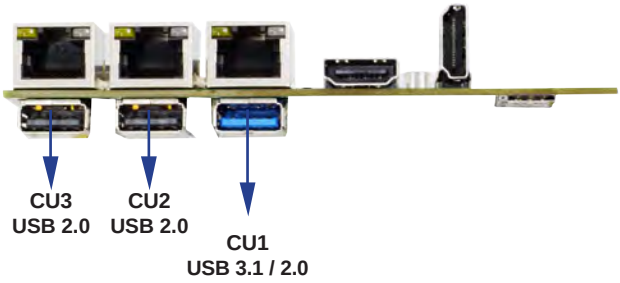
3-3 USB Interface

• CU1: USB 3.1 / 2.0 Type A connector

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
		1	USB3.0 TX+
1	+5V		
2	USB 2.0 D-	2	USB3.0 TX-
		3	GND
3	USB 2.0 D+	4	USB3.0 RX+
4	GND		
		5	USB3.0 RX-

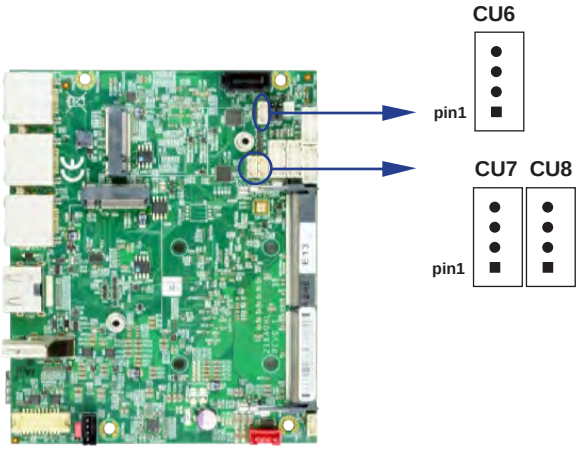
• CU2.CU3: USB 2.0 Type A connector

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	+5V	2	USB 2.0 D-
3	USB 2.0 D+	4	GND



- **CU6.CU7.CU8: USB 2.0 1x4 pin (1.25mm) wafer**

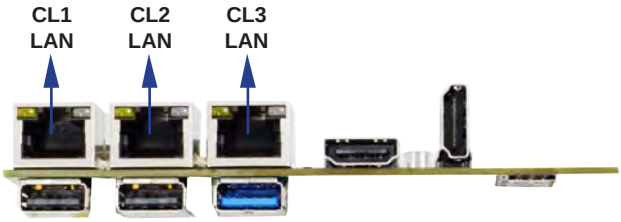
PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	+5V	2	DATA-
3	DATA+	4	GND



3-4 LAN Interface

- **CL1.CL2.CL3: RJ45 LAN Connector**

PIN NO.	Description	PIN NO.	Description
1	TD0+/TX+	2	TD0-/TX-
3	TD1+/RX+	4	TD2+/NC
5	TD2-/NC	6	TD1-/RX-
7	TD3+/NC	8	TD3-/NC



- **RJ45 LAN Connector-LED define 2.5 Giga / 1000 / 100Mb Connector**

Speed	100 Mbps		1000 Mbps		2.5 Gbps	
Indicate	Link LED	Active LED	Link LED	Active LED	Link LED	Active LED
Light						

- **CL11.CL21.CL31: LAN signal out 2x4 pin (2.0mm) wafer (option)**

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	TR0-	2	TR0+
3	TR2+	4	TR1+
5	TR1-	6	TR2+
7	TR3-	8	TR3+

3-5 COM interface

CC1.CC2: COM1 / COM2 2x5 pin (2.0mm) wafer

- **RS232 Mode**

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	DCD	2	RXD
3	TXD	4	DTR
5	GND	6	DSR
7	RTS	8	CTS
9	RI	10	+5V

Note:

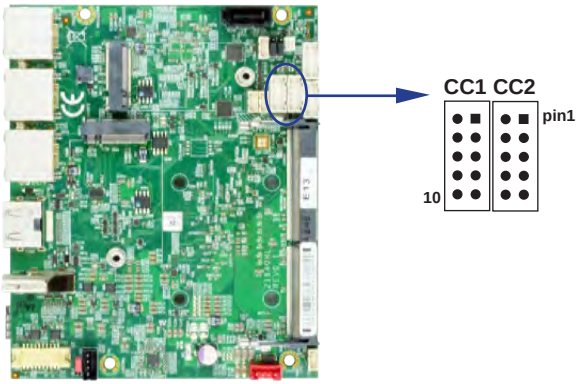
1. COM 1/2 Default RS232, RS485 / RS422 by BIOS control.
2. The pin9 RI can be modify to Power to supply device. The power voltage can be set +12V or +5V. The RI change Voltage function set by BOM control. Default is RI signal.
3. Pin 10 provides +5V for external device.

- **RS485 Mode**

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	Data-	2	Data+
3	NC	4	NC
5	GND	6	NC
7	NC	8	NC
9	NC	10	+5V

- **RS422 Mode**

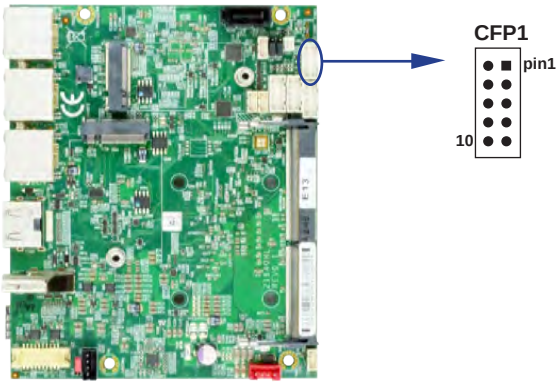
PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	TX-	2	TX+
3	RX+	4	RX-
5	GND	6	NC
7	NC	8	NC
9	NC	10	+5V



3-6 Front Panel Pin Header

- **CFP1**: **Front Panel 2x5 pin (2.0mm) wafer**

PIN NO.	Description	PIN NO.	Description
1	Power button pin	2	Power button GND
3	Reset pin	4	Reset GND
5	Power LED-	6	Power LED+
7	HDD LED-	8	HDD LED+
9	LAN LED-	10	LAN LED+

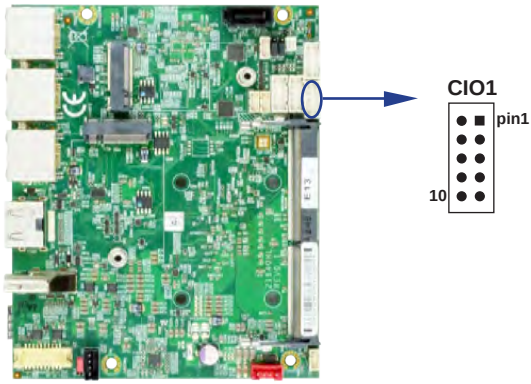


3-7 Digital Input / Output / Watch Dog Time

- CIO1: DIO 0-3 2x5 pin (2.0mm) wafer

PIN NO.	Description	PIN NO.	Description
1	DI-0	2	DO-3
3	DI-1	4	DO-2
5	DI-2	6	DO-1
7	DI-3	8	DO-0
9	GND	10	+5V

Note: 1. DI pin default pull up 10KΩ to +5V
2. If use need isolate circuit to control external device
3. F75111N-1 SMBus address 0x9c



- **WDT For F75111N SMBus watch dog timer device:**

DC spec :
Input low Voltage (VIL): +0.8 Max,
Input High Voltage(VIH) : +2V Min
Output low Current (IOL): 10mA (Min) VOL=0.4V
Output High Current (IOH): -10mA (Min) VOH=2.4V
Watch Dog Time value 0~255 sec
The system will be issued reset. When WDT is enable the hardware start down counter to zero.
The reset timer have 10~20% tolerance upon the Temperature.
Note: If want to SDK support. Please contact to sales window.

3-7-1 IO Device: F75111 CIO Utility

The Sample code source you can download from

http://tprd.info/lexwiki/index.php/IO_Device:F75111_CIO_Utility

<Google Drive>
Source file: CIO_Utility_v3.0.7.2W_Src
Binary file: CIO_Utility_v3.0.7.2W_Bin_x86 CIO_Utility_v3.0.7.2W_Bin_x64
F75113 DLL: F75113.zip

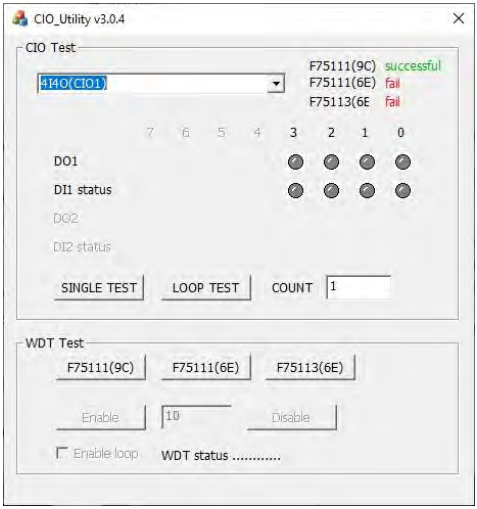
<FTP>
Source file: CIO_Utility_v3.0.7.2W_Src
Binary file: CIO_Utility_v3.0.7.2W_Bin_x86 CIO_Utility_v3.0.7.2W_Bin_x64
F75113 DLL: F75113.zip

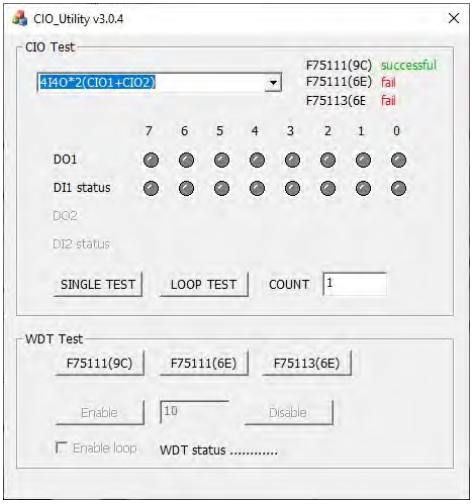
MB Support List

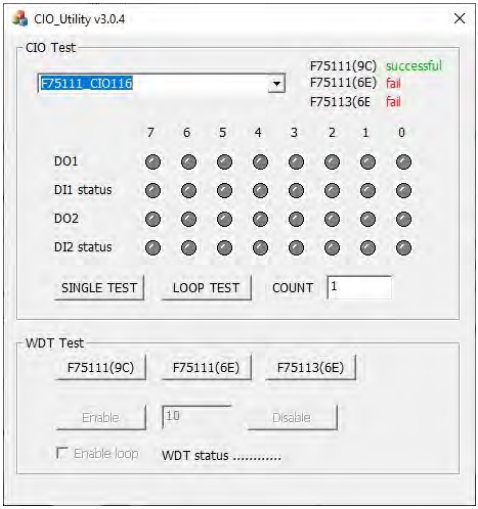
Ivybridge	Bay Trail	Apollo Lake	Skylake / Kabylake
2I847H 3I847A/D/CW/ NX/NM/HW 3I770A/CW CI847A/C CI770A/C	1I385A/H 1I386H 2I380A 2I385A/CW 2I380NX 2I385BW/EW/HW/PW 2I386EW 2I382A 2I382DW 3I380A/CW/NX ST385W/AW/CW	2I390CW 2I392CW 3I390AW 3I390NX 3I393NX PM390C	2I610DW/HW 2I612CW 3I610DW 3I612DW 3I170DW/HW/NX ST610W CI170A/C PM610DW PM170DW

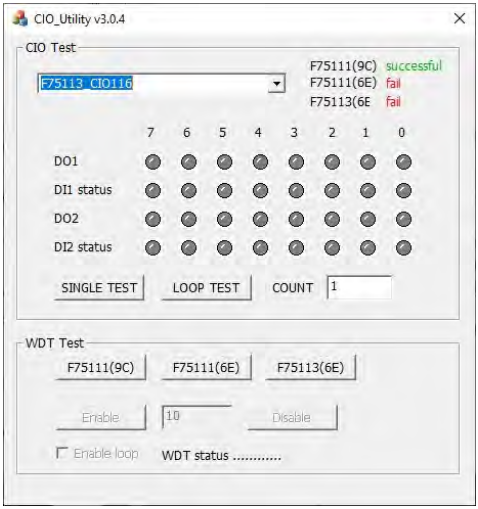
Coffee Lake	Whiskey Lake	AMD	Card	Elkhart Lake
CI370DW	2I810D 3I810DW	3A100DW	CIO116-G E691A	2I640DW

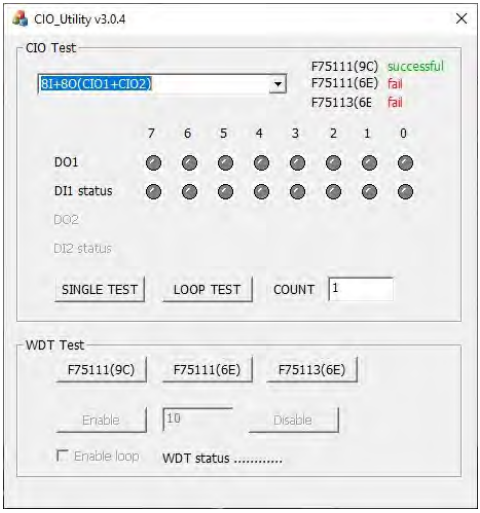
How to use this Demo Application









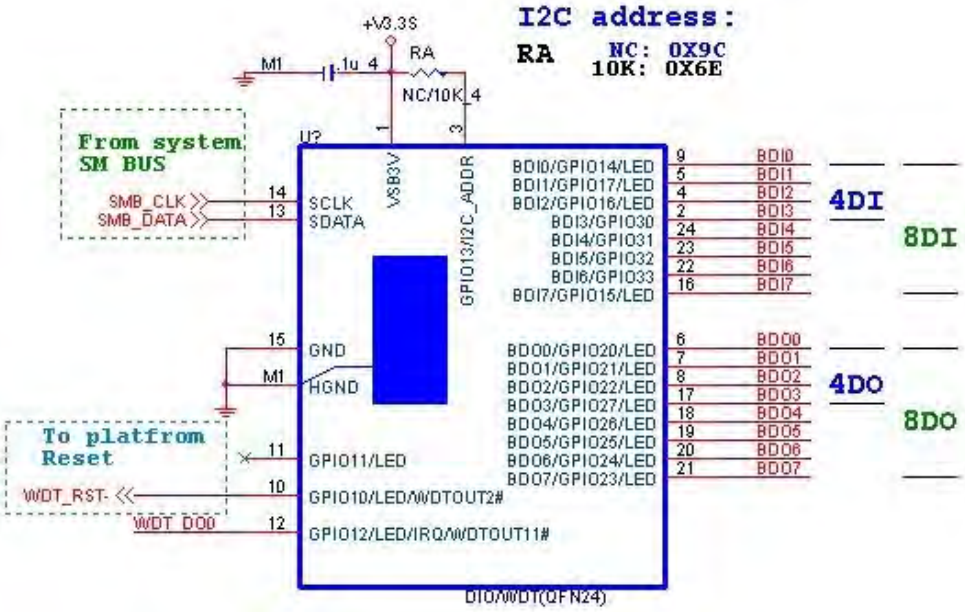


Attention Please:You must be install vcredist_x86.exe when first time you run the F75111_DIO.exe DEMO AP, The vcredist_x86.exe include all required DLL file.

1. Press the select your test "2i2o", "4i4o", "4i4o*2", "F75111CIO116", "F75113CIO116", "8i+8o", "1i1o"

2. start test, select single mode or looptest

F75111 Layout Picture



Introduction F75111

Initial Internal F75111 port address (0x9c)

define GPIO1X, GPIO2X, GPIO3X to input or output
and Enable WDT function pin

Set F75111 DI/DO (sample code as below Get Input value/Set output value)

DO: InterDigitalOutput(BYTE byteValue))
DI: InterDigitalInput()

PULSE mode

Sample to setting GP33, 32, 31, 30 output 1mS low pulse signal.

```
{
this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_PULSE_CONTROL,                0x00);
// This is setting low,Level output
this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_PULSE_WIDTH_CONTROL,     0x01);
// This selects the pulse width to 1mS
this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_CONTROL_MODE,                0x0F);
// This is setting the GP33, 32, 31, 30 to output function.
this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_Output_Data ,                0x0F);
// This is setting the GP33, 32, 31, 30 output data.
}
```

Initial internal F75111

```
void F75111::InitInternalF75111()
{
this->Write_Byte(F75111_INTERNAL_ADDR,GPIO1X_CONTROL_MODE,                0x00)
; //set GPIO1X to Input  function
this->Write_Byte(F75111_INTERNAL_ADDR,GPIO3X_CONTROL_MODE,                0x00)
; //set GPIO3X to Input  function
this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_CONTROL_MODE,                0xFF)
; //set GPIO2X to Output function
this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_OUTPUT_DRIVING,              0xFF)
; //set GPIO2X to Output Drving

this->Write_Byte(F75111_INTERNAL_ADDR,F75111_CONFIGURATION, 0x03);
//Enable WDT OUT function
}
```

Set output value

```
void F75111::InterDigitalOutput(BYTE byteValue)
{
 BYTE byteData = 0;
 byteData = (byteData & 0x01 )? byteValue + 0x01 : byteValue;
 byteData = (byteData & 0x02 )? byteValue + 0x02 : byteValue;
 byteData = (byteData & 0x04 )? byteValue + 0x04 : byteValue;
 byteData = (byteData & 0x80 )? byteValue + 0x08 : byteValue;
 byteData = (byteData & 0x40 )? byteValue + 0x10 : byteValue;
 byteData = (byteData & 0x20 )? byteValue + 0x20 : byteValue;
 byteData = (byteData & 0x10 )? byteValue + 0x40 : byteValue;
 byteData = (byteData & 0x08 )? byteValue + 0x80 : byteValue;          // get value bit by bit

this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_OUTPUT_DATA,byteData);
// write byteData value via GPIO2X output pin
}
```

Get Input value

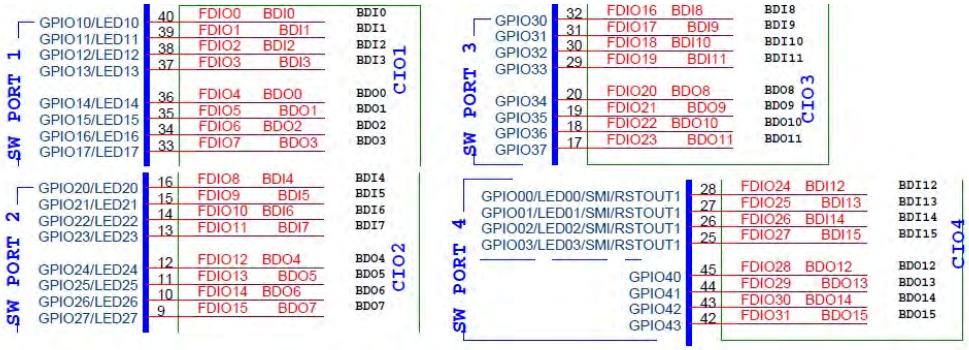
```
BYTE F75111::InterDigitalInput()
{
BYTE byteGPIO1X = 0;
BYTE byteGPIO3X = 0;
BYTE byteData   = 0;

this->Read_Byte(F75111_INTERNAL_ADDR,GPIO1X_INPUT_DATA,&byteGPIO1X) ;
// Get value from GPIO1X
this->Read_Byte(F75111_INTERNAL_ADDR,GPIO3X_INPUT_DATA,&byteGPIO3X) ;
// Get value from GPIO3X

byteGPIO1X = byteGPIO1X  & 0xF0;                    // Mask unuseful value
byteGPIO3X = byteGPIO3X  & 0x0F;                    // Mask unuseful value

byteData = ( byteGPIO1X & 0x10 )? byteData + 0x01 : byteData;
byteData = ( byteGPIO1X & 0x80 )? byteData + 0x02 : byteData;
byteData = ( byteGPIO1X & 0x40 )? byteData + 0x04 : byteData;
byteData = ( byteGPIO3X & 0x01 )? byteData + 0x08 : byteData;

byteData = ( byteGPIO3X & 0x02 )? byteData + 0x10 : byteData;
byteData = ( byteGPIO3X & 0x04 )? byteData + 0x20 : byteData;
byteData = ( byteGPIO3X & 0x08 )? byteData + 0x40 : byteData;
byteData = ( byteGPIO1X & 0x20 )? byteData + 0x80 : byteData;
// Get correct DI value from GPIO1X & GPIO3X

return byteData;
}
```

define F75111 pin in F75111.h

```
//-------------------------------------------------------------------------------------------------------------
#define   F75111_INTERNAL_ADDR                  0x9C //  OnBoard  F75111 Chipset
#define   F75111_EXTERNAL_ADDR                  0x6E //  External F75111 Chipset
//-------------------------------------------------------------------------------------------------------------
#define   F75111_CONFIGURATION                  0x03 //  Configure GPIO13 to WDT2 Function
//-------------------------------------------------------------------------------------------------------------
#define   GPIO1X_CONTROL_MODE                   0x10 //  Select Output Mode or Input Mode
#define   GPIO2X_CONTROL_MODE                   0x20 //  Select GPIO2X Output Mode or Input Mode
#define   GPIO3X_CONTROL_MODE                   0x40 //  Select GPIO3X Output Mode or Input Mode
//-------------------------------------------------------------------------------------------------------------
#define   GPIO1X_INPUT_DATA                     0x12 //  GPIO1X Input Data Register
#define   GPIO3X_INPUT_DATA                     0x22 //  GPIO2X Input Data Register
#define   GPIO3X_INPUT_DATA                     0x42 //  GPIO3X Input Data Register
//-------------------------------------------------------------------------------------------------------------
#define   GPIO1X_OUTPUT_DATA                    0x11 //  GPIO1X Output Data Register
#define   GPIO2X_OUTPUT_DATA                    0x21 //  GPIO2X Output Data Register
#define   GPIO3X_OUTPUT_DATA                    0x41 //  GPIO3X Output Data Register
//-------------------------------------------------------------------------------------------------------------
#define   GPIO1X_OUTPUT_DRIVING                 0x1B //  Select GPIO1X Output Driving Enable
#define   GPIO2X_OUTPUT_DRIVING                 0x2B //  Select GPIO2X Output Driving Enable
#define   GPIO3X_OUTPUT_DRIVING                 0x4B //  Select GPIO3X Output Driving Enable
//-------------------------------------------------------------------------------------------------------------
#define   GPIO1X_PULSE_CONTROL                  0x13 //  GPIO1x Level/Pulse Control Register
                                                     //  0:Level Mode
                                                     //  1:Pulse Mode
#define   GPIO1X_PULSE_WIDTH_CONTROL            0x14 //  GPIO1x Pulse Width Control Register
#define   GP1_PSWIDTH_500US                     0x00 //  When select Pulse mode:500    us.
#define   GP1_PSWIDTH_1MS                       0x01 //  When select Pulse mode:1      ms.
#define   GP1_PSWIDTH_20MS                      0x02 //  When select Pulse mode:20     ms.
#define   GP1_PSWIDTH_100MS                     0x03 //  When select Pulse mode:100    ms.
//-------------------------------------------------------------------------------------------------------------
#define   GPIO2X_PULSE_CONTROL                  0x23 //  GPIO2x Level/Pulse Control Register
                                                     //  0:Level Mode
                        //  1:Pulse Mode
#define   GPIO2X_PULSE_WIDTH_CONTROL            0x24  //  GPIO2x Pulse Width Control Register
#define   GP2_PSWIDTH_500US                     0x00  //  When select Pulse mode:500    us.
#define   GP2_PSWIDTH_1MS                       0x01  //  When select Pulse mode:1      ms.
#define   GP2_PSWIDTH_20MS                      0x02  //  When select Pulse mode:20     ms.
#define   GP2_PSWIDTH_100MS                     0x03  //  When select Pulse mode:100    ms.
//-------------------------------------------------------------------------------------------------------------
#define   GPIO3X_PULSE_CONTROL                  0x43 //  GPIO3x Level/Pulse Control Register
                                                     //  0:Level Mode
                                                     //  1:Pulse Mode
#define   GPIO3X_Output_Data                    0x41 //  GPIO3x Output Data Register
#define   GPIO3X_PULSE_WIDTH_CONTROL            0x44 //  GPIO3x Pulse Width Control Register
#define   GP3_PSWIDTH_500US                     0x00 //  When select Pulse mode:500    us.
#define   GP3_PSWIDTH_1MS                       0x01 //  When select Pulse mode:1      ms.
#define   GP3_PSWIDTH_20MS                      0x02 //  When select Pulse mode:20     ms.
#define   GP3_PSWIDTH_100MS                     0x03 //  When select Pulse mode:100    ms.
//-------------------------------------------------------------------------------------------------------------
```

Introduction F75113

F75113 Layout Picture



Base on 75113.Dll API function as below list

F75113_API bool _stdcall F75113_Init();

F75113_API BYTE F75113_GetDigital_Low_Input();
//BDI0-BDI7

F75113_API BYTE F75113_GetDigital_High_Input();
//BDI8-BDI15

F75113_API void F75113_SetDigital_Low_Output(BYTE byteValue);
//BDO0-BDO7

F75113_API void F75113_SetDigital_High_Output(BYTE byteValue);
//BDO8-BDO15

F75113_API void F75113_SetWDT_Enable(BYTE byteTimer);
//For the F75113 on board

F75113_API void F75113_SetWDT_Disable();
//For the F75113 on board

3-7-2 IO Device:F75111 CIO Utility under Linux

http://tprd.info/lexwiki/index.php/IO_Device:F75111_CIO_Utility_under_Linux

The Sample code source you can download from<Google Drive>

Source file: CIO_Utility_v3.2.1L_Src
Binary file: CIO_Utility_v3.2.1L_Bin_x64

<FTP>
Source file: CIO_Utility_v3.2.1L_Src
Binary file: CIO_Utility_v3.2.1L_Bin_x64

MB Support List

Ivybridge	Bay Trail	Apollo Lake	Skylake / Kabylake
2I847H 3I847A/D/CW /NX/NM/HW 3I770A/CW CI847A/C CI770A/C	1I385A/H 1I386H 2I380A 2I385A/CW 2I380NX 2I385BW/EW/HW/PW 2I386EW 2I382A 2I382DW 3I380A/CW/NX ST385W/AW/CW	2I390CW 2I392CW 3I390AW 3I390NX 3I393NX PM390C	2I610DW/HW 2I612CW 3I610DW 3I612DW 3I170DW/HW/NX ST610W CI170A/C PM610DW PM170DW

Coffee Lake	Whiskey Lake	AMD	Card	Elkhart Lake
CI370DW	2I810D 3I810DW	3A100DW	CIO116-G E691A	2I640DW

How to use this Demo Application

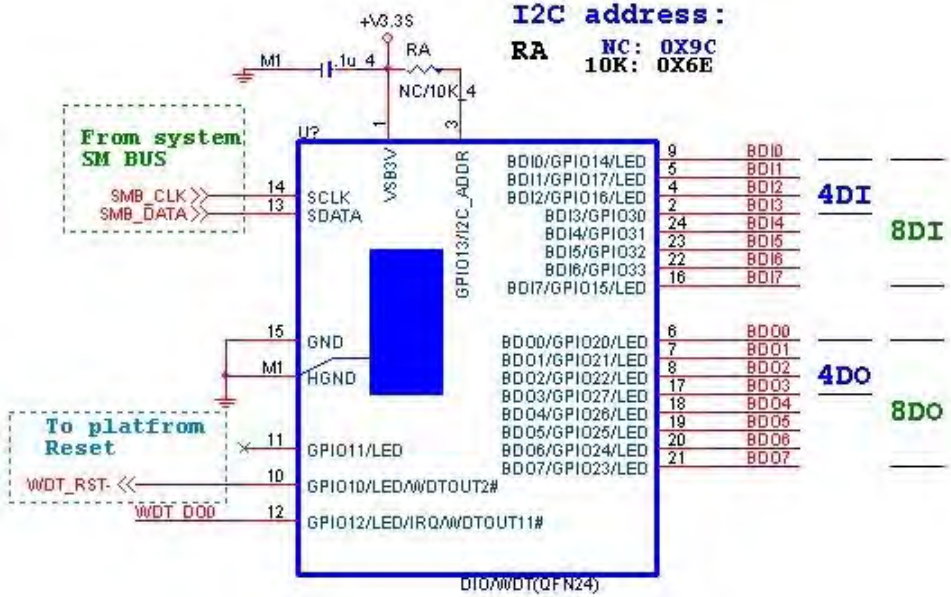
Before executing the program began, Please switch to the highest authority , continued second F75111 ,chmod 777 and root: \

Platform will detect intel/AMD , if not get SMbus signal show N/A
If get SMBUS1 / SMBUS2 F75111 / F75113 will show Sessuce, IC not get will show Fail,
Not get SMBUS1 / SMBUS2 signal will show N/A
1. Press the select your test F75111 or F75113

2. select your test SMBus1(CO1) / SMBus2(CO2) "2i2o", "4i4o", "4i4o*2", "CIO116", "8I(CIO1)+8O(CIO2)", "1I1O"

3. start button，select single mode or looptest

F75111 Layout Picture



Introduction

Initial Internal F75111 port address (0x9c)

define GPIO1X, GPIO2X, GPIO3X to input or output
and Enable WDT function pin

Set F75111 DI/DO (sample code as below Get Input value / Set output value)

DO: InterDigitalOutput(BYTE byteValue))
DI: InterDigitalInput()

PULSE mode

Sample to setting GP33, 32, 31, 30 output 1mS low pulse signal.

```
{
this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_PULSE_CONTROL,          0x00);
//This is setting low pulse output
this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_PULSE_WIDTH_CONTROL,    0x01);
//This selects the pulse width to 1mS
this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_CONTROL_MODE,           0x0F);
//This is setting the GP33, 32, 31, 30 to output function.
this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_Output_Data ,           0x0F);
//This is setting the GP33, 32, 31, 30 output data.
}
```

Initial internal F75111

```
void F75111::InitInternalF75111()
{
this->Write_Byte(F75111_INTERNAL_ADDR,GPIO1X_CONTROL_MODE,        0x00);
 //set GPIO1X to Input  function
this->Write_Byte(F75111_INTERNAL_ADDR,GPIO3X_CONTROL_MODE,        0x00);
 //set GPIO3X to Input  function
this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_CONTROL_MODE,        0xFF);
 //set GPIO2X to Output function

this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_OUTPUT_DRIVING,      0xFF);
 //set GPIO2X to Output Drving

this->Write_Byte(F75111_INTERNAL_ADDR,F75111_CONFIGURATION,       0x03);
 //Enable WDT OUT function
}
```

Set output value

```
void F75111::InterDigitalOutput(BYTE byteValue)
{
 BYTE byteData = 0;
 byteData = (byteData & 0x01 )? byteValue + 0x01 : byteValue;
 byteData = (byteData & 0x02 )? byteValue + 0x02 : byteValue;
 byteData = (byteData & 0x04 )? byteValue + 0x04 : byteValue;
 byteData = (byteData & 0x80 )? byteValue + 0x08 : byteValue;
 byteData = (byteData & 0x40 )? byteValue + 0x10 : byteValue;
 byteData = (byteData & 0x20 )? byteValue + 0x20 : byteValue;
 byteData = (byteData & 0x10 )? byteValue + 0x40 : byteValue;
 byteData = (byteData & 0x08 )? byteValue + 0x80 : byteValue;        // get value bit by bit

 this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_OUTPUT_DATA,byteData);
// write byteData value via GPIO2X output pin
}
```

Get Input value

```
BYTE F75111::InterDigitalInput()
{
BYTE byteGPIO1X = 0;
BYTE byteGPIO3X = 0;
BYTE byteData   = 0;

this->Read_Byte(F75111_INTERNAL_ADDR,GPIO1X_INPUT_DATA,&byteGPIO1X) ;
 // Get value from GPIO1X
this->Read_Byte(F75111_INTERNAL_ADDR,GPIO3X_INPUT_DATA,&byteGPIO3X) ;
 // Get value from GPIO3X

byteGPIO1X = byteGPIO1X  & 0xF0;                    // Mask unuseful value
byteGPIO3X = byteGPIO3X  & 0x0F;                    // Mask unuseful value

byteData = ( byteGPIO1X & 0x10 )? byteData + 0x01 : byteData;
byteData = ( byteGPIO1X & 0x80 )? byteData + 0x02 : byteData;
byteData = ( byteGPIO1X & 0x40 )? byteData + 0x04 : byteData;
byteData = ( byteGPIO3X & 0x01 )? byteData + 0x08 : byteData;

byteData = ( byteGPIO3X & 0x02 )? byteData + 0x10 : byteData;
byteData = ( byteGPIO3X & 0x04 )? byteData + 0x20 : byteData;
byteData = ( byteGPIO3X & 0x08 )? byteData + 0x40 : byteData;
byteData = ( byteGPIO1X & 0x20 )? byteData + 0x80 : byteData;
 // Get correct DI value from GPIO1X & GPIO3X

return byteData;
 }
```

define F75111 pin in F75111.h

```
//-------------------------------------------------------------------------------------------------
#define   F75111_INTERNAL_ADDR                0x9C //  OnBoard  F75111 Chipset
#define   F75111_EXTERNAL_ADDR                0x6E //  External F75111 Chipset
//-------------------------------------------------------------------------------------------------
#define   F75111_CONFIGURATION                0x03 //  Configure GPIO13 to WDT2 Function
//-------------------------------------------------------------------------------------------------
#define   GPIO1X_CONTROL_MODE                 0x10 //  Select Output Mode or Input Mode
#define   GPIO2X_CONTROL_MODE                 0x20 //  Select GPIO2X Output Mode or Input Mode
#define   GPIO3X_CONTROL_MODE                 0x40 //  Select GPIO3X Output Mode or Input Mode
//-------------------------------------------------------------------------------------------------
#define   GPIO1X_INPUT_DATA                   0x12 //  GPIO1X Input Data Register
#define   GPIO3X_INPUT_DATA                   0x22 //  GPIO2X Input Data Register
#define   GPIO3X_INPUT_DATA                   0x42 //  GPIO3X Input Data Register
//-------------------------------------------------------------------------------------------------
#define   GPIO1X_OUTPUT_DATA                  0x11 //  GPIO1X Output Data Register
#define   GPIO2X_OUTPUT_DATA                  0x21 //  GPIO2X Output Data Register
#define   GPIO3X_OUTPUT_DATA                  0x41 //  GPIO3X Output Data Register
//-------------------------------------------------------------------------------------------------
#define   GPIO1X_OUTPUT_DRIVING               0x1B //  Select GPIO1X Output Driving Enable
#define   GPIO2X_OUTPUT_DRIVING               0x2B //  Select GPIO2X Output Driving Enable
#define   GPIO3X_OUTPUT_DRIVING               0x4B //  Select GPIO3X Output Driving Enable
//-------------------------------------------------------------------------------------------------
#define   GPIO1X_PULSE_CONTROL                0x13 //  GPIO1x Level/Pulse Control Register
                                                   //  0:Level Mode
                                                   //  1:Pulse Mode
#define   GPIO1X_PULSE_WIDTH_CONTROL          0x14 //  GPIO1x Pulse Width Control Register
#define   GP1_PSWIDTH_500US                   0x00 //  When select Pulse mode:500    us.
#define   GP1_PSWIDTH_1MS                     0x01 //  When select Pulse mode:1      ms.
#define   GP1_PSWIDTH_20MS                    0x02 //  When select Pulse mode:20     ms.
#define   GP1_PSWIDTH_100MS                   0x03 //  When select Pulse mode:100    ms.
//-------------------------------------------------------------------------------------------------
#define   GPIO2X_PULSE_CONTROL                0x23 //  GPIO2x Level/Pulse Control Register
                                                   //  0:Level Mode
                                                   //  1:Pulse Mode
#define   GPIO2X_PULSE_WIDTH_CONTROL          0x24 //  GPIO2x Pulse Width Control Register
#define   GP2_PSWIDTH_500US                   0x00 //  When select Pulse mode:500    us.
#define   GP2_PSWIDTH_1MS                     0x01 //  When select Pulse mode:1      ms.
#define   GP2_PSWIDTH_20MS                    0x02 //  When select Pulse mode:20     ms.
#define   GP2_PSWIDTH_100MS                   0x03 //  When select Pulse mode:100    ms.
//-------------------------------------------------------------------------------------------------
#define   GPIO3X_PULSE_CONTROL                0x43 //  GPIO3x Level/Pulse Control Register
                                                   //  0:Level Mode
                                                   //  1:Pulse Mode
#define   GPIO3X_Output_Data                  0x41 //  GPIO3x Output Data Register
#define   GPIO3X_PULSE_WIDTH_CONTROL          0x44 //  GPIO3x Pulse Width Control Register
#define   GP3_PSWIDTH_500US                   0x00 //  When select Pulse mode:500    us.
#define   GP3_PSWIDTH_1MS                     0x01 //  When select Pulse mode:1      ms.
#define   GP3_PSWIDTH_20MS                    0x02 //  When select Pulse mode:20     ms.
#define   GP3_PSWIDTH_100MS                   0x03 //  When select Pulse mode:100    ms.
//-------------------------------------------------------------------------------------------------
```

3-7-3 IO Device:F75111 CIO Utility Console under linux

http://tprd.info/lexwiki/index.php/IO_Device:F75111_CIO_Utility_Console_under_linux

The Sample code source you can download from

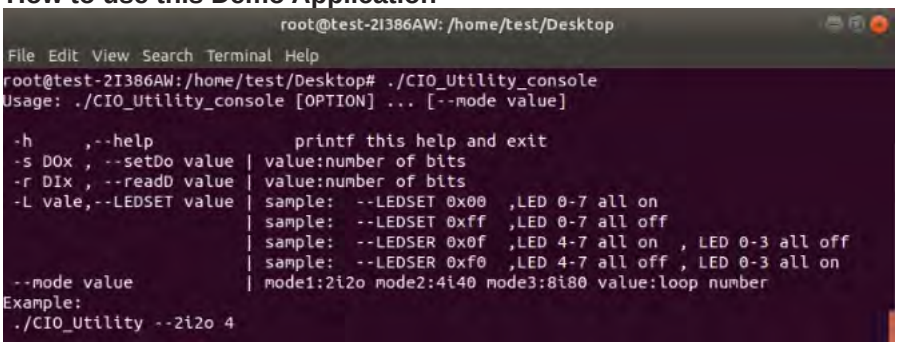
<Google Drive>
Source file: CIO_Utility_Console_v1.4L_Src
Binary file: CIO_Utility_Console_v1.4L_Bin

<FTP>
Source file: CIO_Utility_Console_v1.4L_Src
Binary file: CIO_Utility_Console_v1.4L_Bin

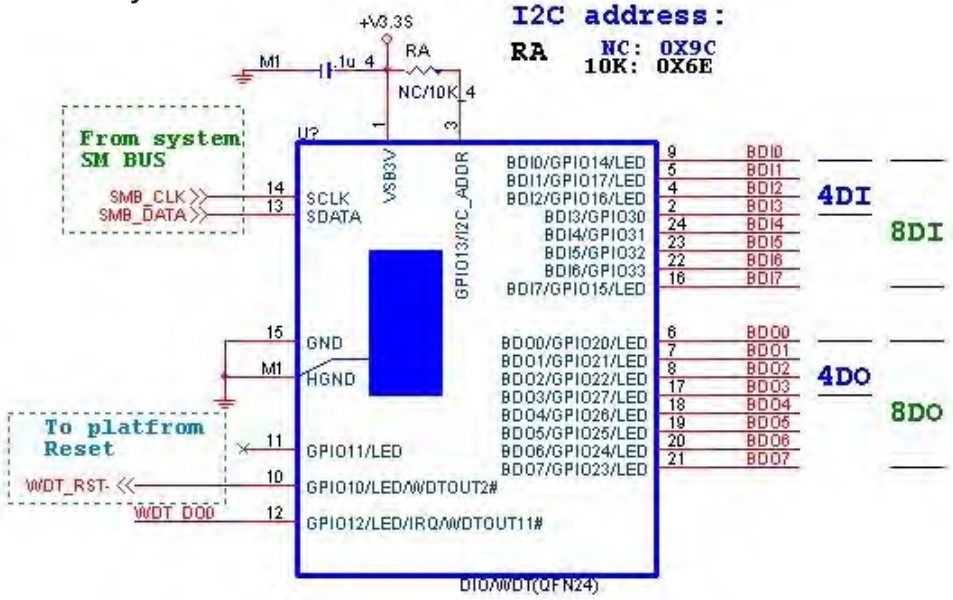
MB Support List

Ivybridge	BayTrail	Apollo Lake	Skylake/Kabylake	Card
2I847H	1I385A/H	2I390CW	2I610DW/HW	CIO116-G
3I8347A/CW	1I386HW	2I390CW	2I610HW	E691A
3I847NX/NM	2I380A/NX	3I390AW	3I610DW	
3I847D(OEM)	2I382A	3I390D(OEM)	PM610DW	
3I847HW	2I385A/BW/CW/ EW/HW/PW	3I390NX	ST610W	
CI847A/C	3I380A/CW/D/NX	3I393NX	3I170DW/HW/NX	
3I770A/CW	3I385AW/CW	PM390CW	CI170A/C	
CI770A/C	ST385W		PM170DW	

How to use this Demo Application



F75111 Layout Picture



Introduction

Initial Internal F75111 port address (0x9c)

define GPIO1X, GPIO2X, GPIO3X to input or output
and Enable WDT function pin

Set F75111 DI/DO (sample code as below Get Input value / Set output value)

DO: InterDigitalOutput (BYTE byteValue))
DI: InterDigitalInput()

PULSE mode

Sample to setting GP33, 32, 31, 30 output 1mS low pulse signal.

```
{
   this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_PULSE_CONTROL              0x00);
   //This is setting low pulse output
   this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_PULSE_WIDTH_CONTROL, 0x01);
   //This selects the pulse width to 1mS
   this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_CONTROL_MODE,              0x0F);
   //This is setting the GP33, 32, 31, 30 to output function.
   this->Write_Byte(F75111_INTERNAL_ADDR, GPIO3X_Output_Data ,              0x0F);
   //This is setting the GP33, 32, 31, 30 output data.
}
```

Initial internal F75111

```
void F75111::InitInternalF75111()
{
   this->Write_Byte(F75111_INTERNAL_ADDR,GPIO1X_CONTROL_MODE,     0x00); //set GPIO1X to Input  function
   this->Write_Byte(F75111_INTERNAL_ADDR,GPIO3X_CONTROL_MODE,     0x00); //set GPIO3X to Input  function
   this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_CONTROL_MODE,     0xFF); //set GPIO2X to Output function
   this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_OUTPUT_DRIVING,   0xFF); //set GPIO2X to Output Drving
   this->Write_Byte(F75111_INTERNAL_ADDR,F75111_CONFIGURATION,    0x03); //Enable WDT OUT function
}
```

Set output value

```
void F75111::InterDigitalOutput(BYTE byteValue)
{
    BYTE byteData = 0;
    byteData = (byteData & 0x01 )? byteValue + 0x01 : byteValue;
    byteData = (byteData & 0x02 )? byteValue + 0x02 : byteValue;
    byteData = (byteData & 0x04 )? byteValue + 0x04 : byteValue;
    byteData = (byteData & 0x80 )? byteValue + 0x08 : byteValue;
    byteData = (byteData & 0x40 )? byteValue + 0x10 : byteValue;
    byteData = (byteData & 0x20 )? byteValue + 0x20 : byteValue;
    byteData = (byteData & 0x10 )? byteValue + 0x40 : byteValue;
    byteData = (byteData & 0x08 )? byteValue + 0x80 : byteValue;          // get value bit by bit

   this->Write_Byte(F75111_INTERNAL_ADDR,GPIO2X_OUTPUT_DATA,byteData);
   // write byteData value via GPIO2X output pin
}
```

Get Input value

```
BYTE F75111::InterDigitalInput()
{
   BYTE byteGPIO1X = 0;
   BYTE byteGPIO3X = 0;
   BYTE byteData   = 0;
   this->Read_Byte(F75111_INTERNAL_ADDR,GPIO1X_INPUT_DATA,&byteGPIO1X) ;   // Get value from GPIO1X
   this->Read_Byte(F75111_INTERNAL_ADDR,GPIO3X_INPUT_DATA,&byteGPIO3X) ; // Get value from GPIO3X

   byteGPIO1X = byteGPIO1X  & 0xF0;                              // Mask unuseful value
   byteGPIO3X = byteGPIO3X  & 0x0F;                              // Mask unuseful value

   byteData = ( byteGPIO1X & 0x10 )? byteData + 0x01 : byteData;
   byteData = ( byteGPIO1X & 0x80 )? byteData + 0x02 : byteData;
   byteData = ( byteGPIO1X & 0x40 )? byteData + 0x04 : byteData;
   byteData = ( byteGPIO3X & 0x01 )? byteData + 0x08 : byteData;

   byteData = ( byteGPIO3X & 0x02 )? byteData + 0x10 : byteData;
   byteData = ( byteGPIO3X & 0x04 )? byteData + 0x20 : byteData;
   byteData = ( byteGPIO3X & 0x08 )? byteData + 0x40 : byteData;
   byteData = ( byteGPIO1X & 0x20 )? byteData + 0x80 : byteData;  // Get correct DI value from GPIO1X & GPIO3X

   return byteData;
}
```

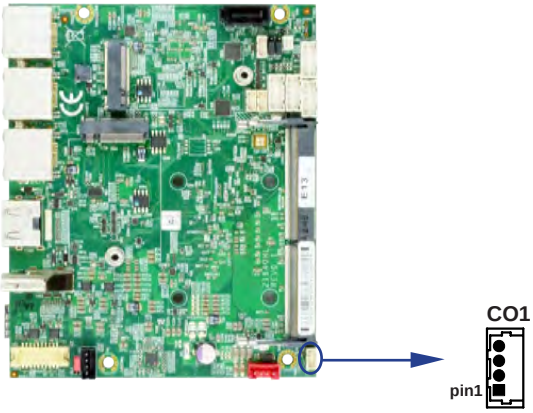
define F75111 pin in F75111.h

```
//-------------------------------------------------------------------------------------------------------------------------------
#define   F75111_INTERNAL_ADDR              0x9C   //  OnBoard  F75111 Chipset
#define   F75111_EXTERNAL_ADDR              0x6E   //  External F75111 Chipset
//-------------------------------------------------------------------------------------------------------------------------------
#define   F75111_CONFIGURATION              0x03   //  Configure GPIO13 to WDT2 Function
//-------------------------------------------------------------------------------------------------------------------------------
#define   GPIO1X_CONTROL_MODE               0x10   //  Select Output Mode or Input Mode
#define   GPIO2X_CONTROL_MODE               0x20   //  Select GPIO2X Output Mode or Input Mode
#define   GPIO3X_CONTROL_MODE               0x40   //  Select GPIO3X Output Mode or Input Mode
//-------------------------------------------------------------------------------------------------------------------------------
#define   GPIO1X_INPUT_DATA                 0x12   //  GPIO1X Input Data Register
#define   GPIO3X_INPUT_DATA                 0x22   //  GPIO2X Input Data Register
#define   GPIO3X_INPUT_DATA                 0x42   //  GPIO3X Input Data Register
//-------------------------------------------------------------------------------------------------------------------------------
#define   GPIO1X_OUTPUT_DATA                0x11   //  GPIO1X Output Data Register
#define   GPIO2X_OUTPUT_DATA                0x21   //  GPIO2X Output Data Register
#define   GPIO3X_OUTPUT_DATA                0x41   //  GPIO3X Output Data Register
//-------------------------------------------------------------------------------------------------------------------------------
#define   GPIO1X_OUTPUT_DRIVING             0x1B   //  Select GPIO1X Output Driving Enable
#define   GPIO2X_OUTPUT_DRIVING             0x2B   //  Select GPIO2X Output Driving Enable
#define   GPIO3X_OUTPUT_DRIVING             0x4B   //  Select GPIO3X Output Driving Enable
//-------------------------------------------------------------------------------------------------------------------------------
#define   GPIO1X_PULSE_CONTROL              0x13   //  GPIO1x Level/Pulse Control Register
                                                   //  0:Level Mode
                                                   //  1:Pulse Mode
#define   GPIO1X_PULSE_WIDTH_CONTROL        0x14   //  GPIO1x Pulse Width Control Register
#define   GP1_PSWIDTH_500US                 0x00   //  When select Pulse mode:     500us.
#define   GP1_PSWIDTH_1MS                   0x01   //  When select Pulse mode:     1ms.
#define   GP1_PSWIDTH_20MS                  0x02   //  When select Pulse mode:     20ms.
#define   GP1_PSWIDTH_100MS                 0x03   //  When select Pulse mode:     100ms.
//-------------------------------------------------------------------------------------------------------------------------------
#define   GPIO2X_PULSE_CONTROL              0x23   //  GPIO2x Level/Pulse Control Register
                                                   //  0:Level Mode
                                                   //  1:Pulse Mode
#define   GPIO2X_PULSE_WIDTH_CONTROL        0x24   //  GPIO2x Pulse Width Control Register
#define   GP2_PSWIDTH_500US                 0x00   //  When select Pulse mode:     500us.
#define   GP2_PSWIDTH_1MS                   0x01   //  When select Pulse mode:     1ms.
#define   GP2_PSWIDTH_20MS                  0x02   //  When select Pulse mode:     20ms.
#define   GP2_PSWIDTH_100MS                 0x03   //  When select Pulse mode:     100ms.
//-------------------------------------------------------------------------------------------------------------------------------
#define   GPIO3X_PULSE_CONTROL              0x43   //  GPIO3x Level/Pulse Control Register
                                                   //  0:Level Mode
                                                   //  1:Pulse Mode
#define   GPIO3X_Output_Data                0x41   //  GPIO3x Output Data Register
#define   GPIO3X_PULSE_WIDTH_CONTROL        0x44   //  GPIO3x Pulse Width Control Register
#define   GP3_PSWIDTH_500US                 0x00   //  When select Pulse mode:     500     us.
#define   GP3_PSWIDTH_1MS                   0x01   //  When select Pulse mode:     1       ms.
#define   GP3_PSWIDTH_20MS                  0x02   //  When select Pulse mode:     20      ms.
#define   GP3_PSWIDTH_100MS                 0x03   //  When select Pulse mode:     100     ms.
//-------------------------------------------------------------------------------------------------------------------------------
```

3-8 SMBus Interface

- **CO1: SMBus 1x4 pin (1.25mm) wafer**

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	+3.3V	2	GND
3	SMB-Clock	4	SMB-Data

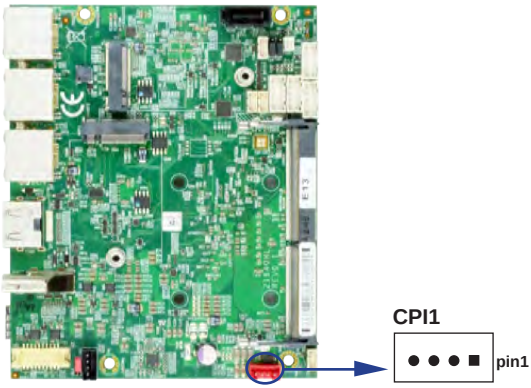


3-9 DC Power Input

- **CPI1: DC Power input 1x4 pin (2.0mm) wafer (RED)**

PIN NO.	DESCRIPTION
1,4	GND
2,3	DC-IN

Note: Very important check DC-in Voltage.

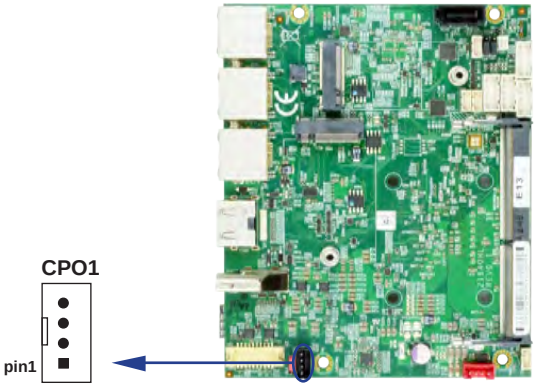


3-10 DC +12V / +5 Voltage Power Output

- **CPO1: +12V / +5V DC voltage output 1x4 pin (2.0mm) wafer (Black)**

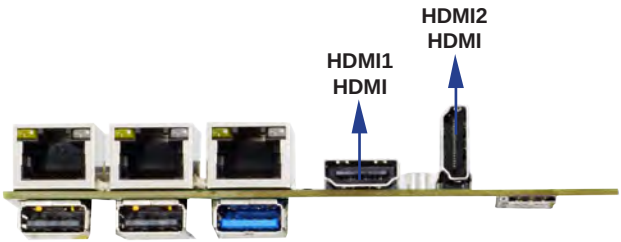
PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	+5V	2	GND
3	GND	4	+12V

Note: Attention! Check Device Power in spec.



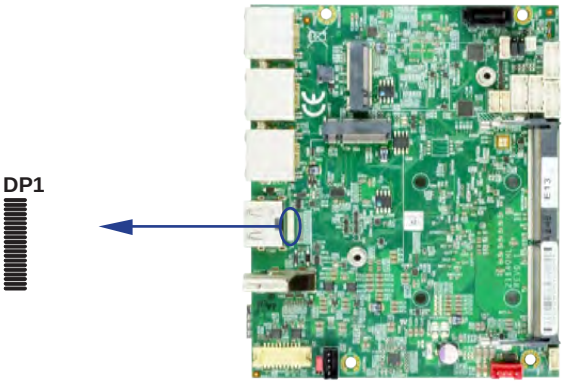
3-11 HDMI1 / HDMI2: HDMI type A connector

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	TMDS DATA2+	2	GND
3	TMDS DATA2-	4	TMDS DATA1+
5	GND	6	TMDS DATA1-
7	TMDS DATA0+	8	GND
9	TMDS DATA0-	10	TMDS CLK+
11	GND	12	TMDS CLK-
13	NC	14	NC
15	DDC CLOCK	16	DDC DATA
17	GND	18	+5V
19	H.P. Detect		



3-12 DP1: DisplayPort connector (option)

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	Lane0+	2	GND
3	Lane0-	4	Lane1+
5	GND	6	Lane1-
7	Lane2+	8	GND
9	Lane2-	10	Lane3+
11	GND	12	Lane3-
13	GND	14	GND
15	AUX_CH+	16	GND
17	AUX_CH-	18	H.P. Detect
19	GND	20	+3.3V

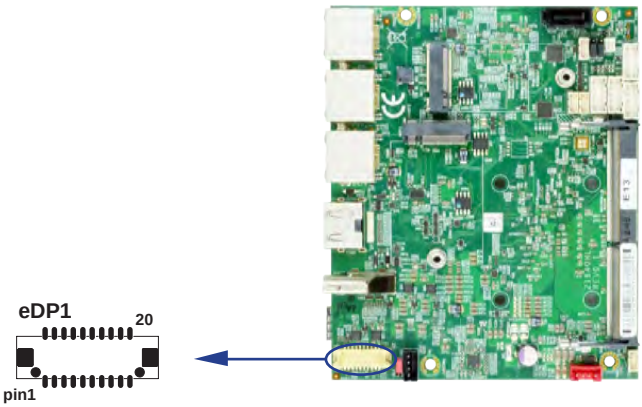


3-13 EDP1: eDP interface 2x10 pin (1.25mm) wafer

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	Lane-0-DATA-	2	+12V
3	Lane-0-DATA+	4	+12V
5	Lane-1-DATA-	6	GND
7	Lane-1-DATA+	8	GND
9	Backlight Enable	10	GND
11	PWM dimming	12	GND
13	I2C Clock	14	+LCD (5V or 3.3V)
15	I2C Data	16	+LCD (5V or 3.3V)
17	eDP Aux+	18	+LCD (5V or 3.3V)
19	eDP Aux-	20	EDP_HPD

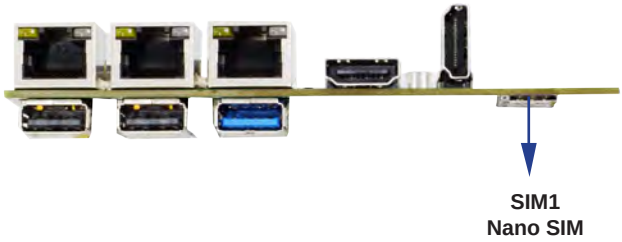
Note:

1. eDP interface support 2 lanes.
2. JVL1: eDP panel +5V / +3.3V (default) Voltage select.
3. eDP1 PIN 9 for panel backlight enable. +3.3V Level
4. eDP1 PIN 11 for panel backlight dimming control



3-14 SIM1: Nano SIM Card Push-Push
Follow ISO 7816-2 Smart Card Standard.

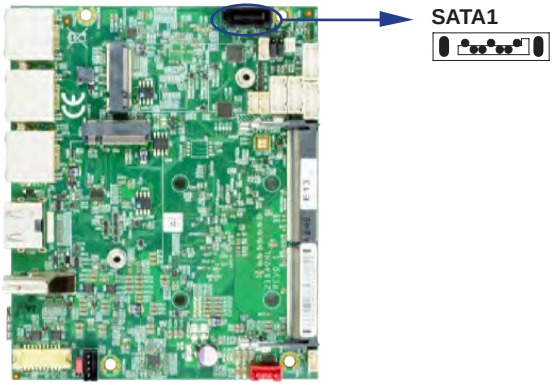
PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	VCC	2	RST
3	CLK	4	NC
5	GND	6	VPP
7	DATA	8	NC



3-15 SATA Interface

- **SATA1: SATA port 1x7 pin Connector**

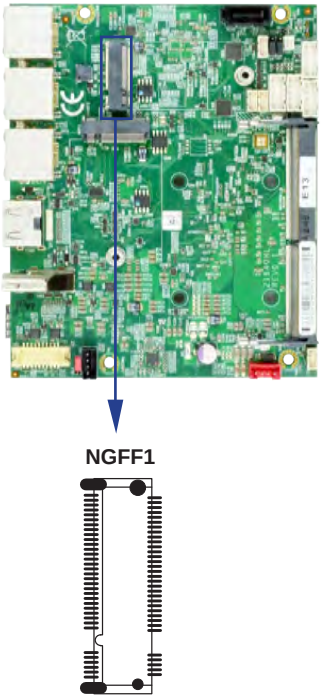
PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	GND	2	TX+
3	TX-	4	GND
5	RX-	6	RX+
7	GND		



3-16 NGFF socket

• NGFF1: PCI Express M.2 B key 2242 H=8.5 sockets 75pin

PIN NO.	Description	PIN NO.	Description
1	GND	2	+3.3V
3	GND	4	+3.3V
5	GND	6	FULL_CARD_PWR
7	USB 2.0_P	8	NC
9	USB 2.0_N	10	M2_LED
11	GND		
B Key notch			
		20	NC
21	GND	22	NC
23	NC	24	NC
25	NC	26	NC
27	GND	28	NC
29	PCIE_5_RX_DN	30	NC
31	PCIE_5_RX_DP	32	NC
33	GND	34	NC
35	PCIE_5_TX_DN	36	NC
37	PCIE_5_TX_DP	38	NC
39	GND	40	NC
41	PCIE_4_RX_DN	42	NC
43	PCIE_4_RX_DP	44	NC
45	GND	46	NC
47	PCIE_4_TX_DN	48	NC
49	PCIE_4_TX_DP	50	PREST
51	GND	52	SRCCLKREQ_N
53	PCIE_CLK_N0	54	NC
55	PCIE_CLK_P0	56	NC
57	GND	58	NC
59	NC	60	NC
61	NC	62	NC
63	NC	64	NC
65	NC	66	NC
67	NC	68	NC
69	NC	70	+3.3V
71	GND	72	+3.3V
73	GND	74	+3.3V
75	NC		

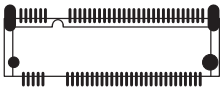


Note: NGFF1 support PCIe x2 & USB 2.0

- **NGFF2: PCI Express M.2 B key 2242 / 3042 H=8.5 sockets 75pin**

PIN NO.	Description	PIN NO.	Description
1	NC	2	+3.3V / +3.7V
3	GND	4	+3.3V / +3.7V
5	GND	6	FULL_CARD_PWR
7	USB 2.0_P	8	W_DISABLE_1
9	USB 2.0_N	10	M2_LED
11	GND		
B Key notch			
		20	NC
21	GND	22	NC
23	NC	24	NC
25	NC	26	W_DISABLE_2
27	GND	28	NC
29	USB31_1_RX_DN	30	SIM_RST_M2
31	USB31_1_RX_DP	32	SIM_CLK_M2
33	GND	34	SIM_DATA_M2
35	USB31_1_TX_DN	36	SIM_PWR_M2
37	USB31_1_TX_DP	38	NC
39	GND	40	NC
41	M2_PERn0_MSRp	42	NC
43	M2_PERp0_MSRn	44	NC
45	GND	46	NC
47	M2_PETn0_MSTn	48	NC
49	M2_PETp0_MSTp	50	PREST
51	GND	52	SRCCLKREQ_N
53	PCIE_CLK_N0	54	NC
55	PCIE_CLK_P0	56	NC
57	GND	58	NC
59	NC	60	NC
61	ANTCTL1	62	NC
63	ANTCTL2	64	NC
65	NC	66	SIM_DET
67	MD_RESET_N	68	NC
69	CFG1_PCIE_SATA_M2	70	+3.3V / +3.7V
71	GND	72	+3.3V / +3.7V
73	GND	74	+3.3V / +3.7V
75	CONFIG_2		

NGFF2



Note:
1. NGFF2 support SATA / PCIe x1 Auto detect.
2. USB 3.1 function by BOM control.
3. VCC voltage default support +3.3V.
4. BOM control, if need 4G LTE device VCC voltage is +3.7V.

3-17 CRFP1: Antenna control 1x4 pin (1.25mm) wafer (OEM)

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	ANTCTL1	2	ANTCTL2
3	+5V	4	GND

Note:
1. Antenna control by OEM
2. Antenna control with NGFF2

3-18 CL12.CL22.CL32:
LAN LED indicator 1x4 pin (1.0mm) wafer (option)

PIN NO.	DESCRIPTION	PIN NO.	DESCRIPTION
1	VCC	2	Speed 100M
3	Speed 1G	4	Speed 2.5G

Note: Can use CL001 connector Board to RJ45.

Chapter-4

Introduction of BIOS

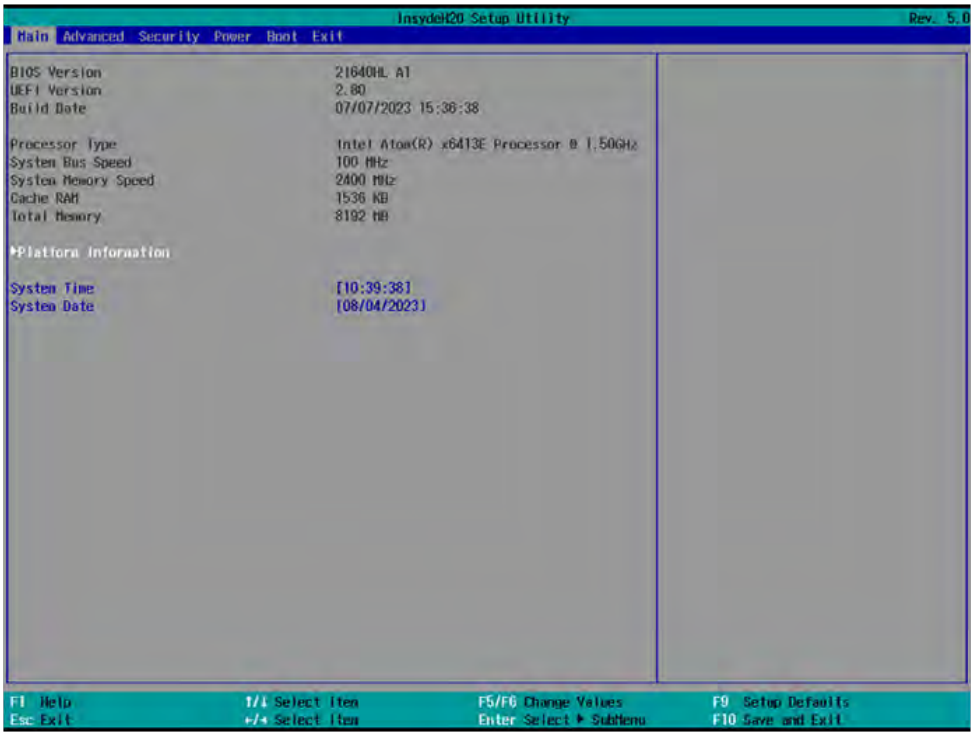
The BIOS is a program located in the Flash Memory on the motherboard.
This program is a bridge between motherboard and operating system.
When you start the computer, the BIOS program gains control.
The BIOS first operates an auto-diagnostic test called POST (Power on Self Test) for all the necessary hardware, it detects the entire hardware devices and configures the parameters of the hardware synchronization. After these tasks are completed, BIOS will give control of the computer back to operating system (OS). Since the BIOS is the only channel for hardware and software to communicate with, it is the key factor of system stability and of ensuring your system performance at best.
In the BIOS Setup main menu, you can see several options. We will explain these options in the following pages. First, let us see the function keys you may use here:

Press <Esc> to quit the BIOS Setup.
Press ↑↓←→(up, down, left, right) to choose the option you want to confirm or modify.
Press <F10> to save these parameters and to exit the BIOS Setup menu after you complete the setup of BIOS parameters.
Press Page Up/Page Down or +/- keys to modify the BIOS parameters for the active option.

4-1 Enter Setup

Power on the computer and press <Del> key immediately to enter Setup.
If the message disappears before your respond but you still wish to enter Setup, restart the system by turning it OFF then ON or pressing the "RESET" button on the system case. You may also restart the system by simultaneously pressing <Ctrl>, <Alt> and <Delete> keys.

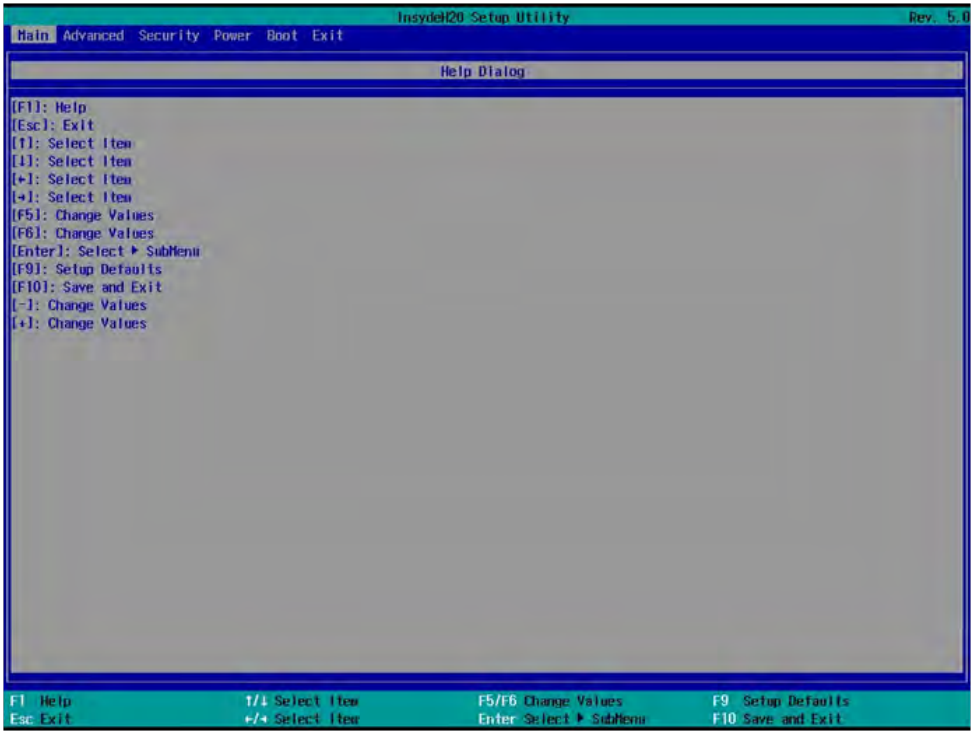
4-2 BIOS Menu Screen & Function Keys



In the above BIOS Setup main menu of, you can see several options.
We will explain these options step by step in the following pages of this chapter, but let us first see a short description of the function keys you may use here:

- Press >< (right, left) to select screen;
- Press ↑↓ (up, down) to choose, in the main menu, the option you want to confirm or to modify.
- Press <Enter> to select.
- Press <+>/<–> or <F5>/<F6> keys when you want to modify the BIOS parameters for the active option.
- [F1]: General help.
- [F2]: Previous values.
- [F3]: Optimized defaults.
- [F4]: Save & Reset.
- Press <Esc> to quit the BIOS Setup.

4-3 Getting Help



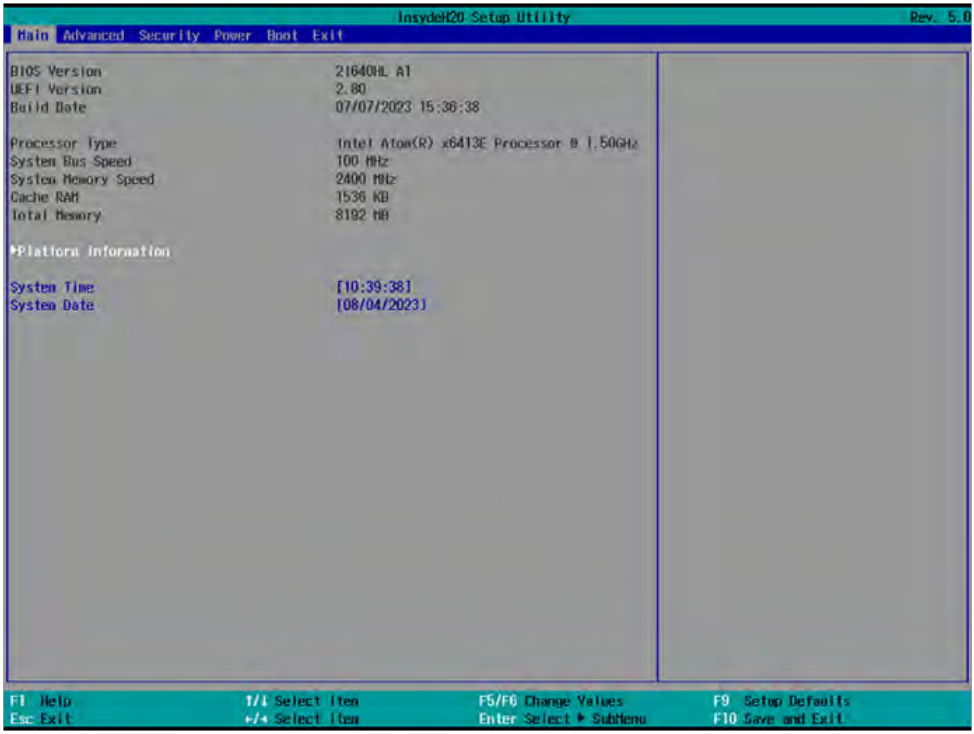
Status Page Setup Menu / Option Page Setup Menu

Press F1 to pop up a help window that describes the appropriate keys to use and the possible selections for the highlighted item. To exit the Help Window, press <Esc>.

4-4 Menu Bars

There are six menu bars on top of BIOS screen:
Main To change system basic configuration
Advanced To change system advanced configuration
Chipset To change PCH IO configuration
Security Password settings
Boot Quiet boot or boot from USB selected.
Save & Exit Save setting, loading and exit options.
User can press the right or left arrow key on the keyboard to switch from menu bar.
The selected one is highlighted.

4-5 Main

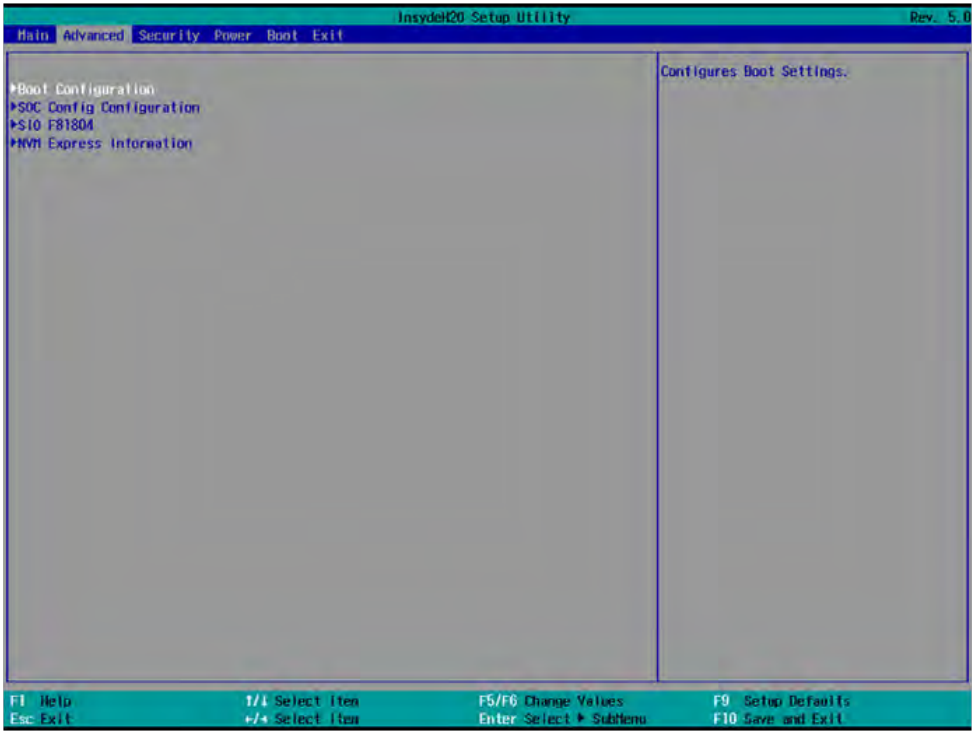


Main menu screen includes some basic system information. Highlight the item and then use the <+> or <-> and numerical keyboard keys to select the value you want in each item.

System Date
Set the Date. Please use [Tab] to switch between data elements.

System Time
Set the Time. Please use [Tab] to switch between data elements.

4-6 Advanced



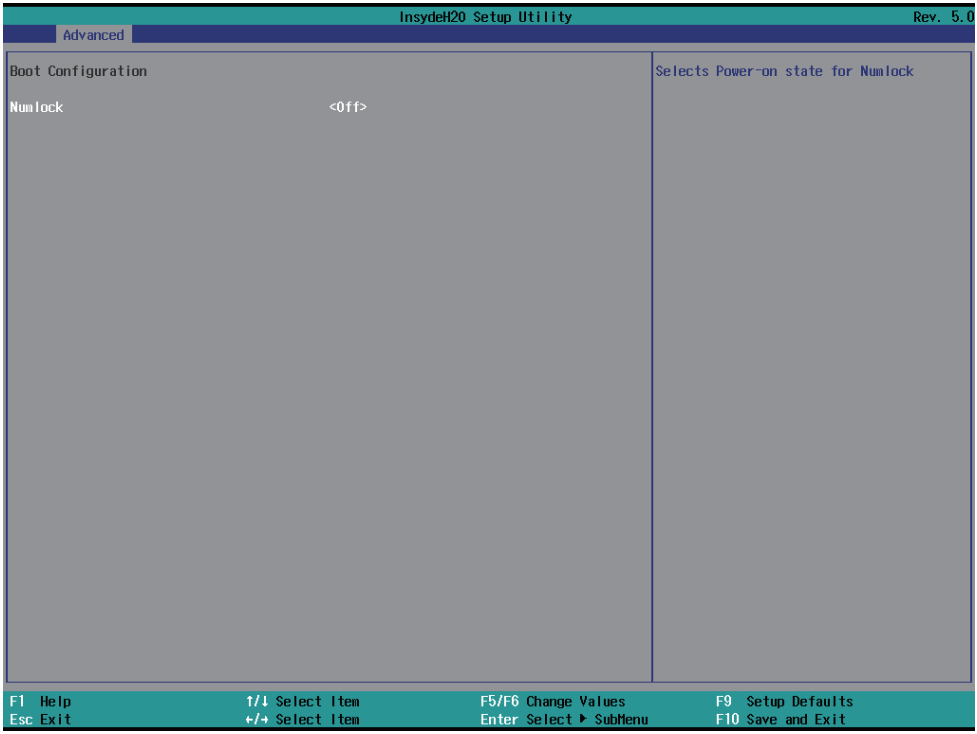
Boot Configuration
Please refer section 4-6-1

SOC Config Configuration
Please refer section 4-6-2

SIO F81804
Please refer section 4-6-3

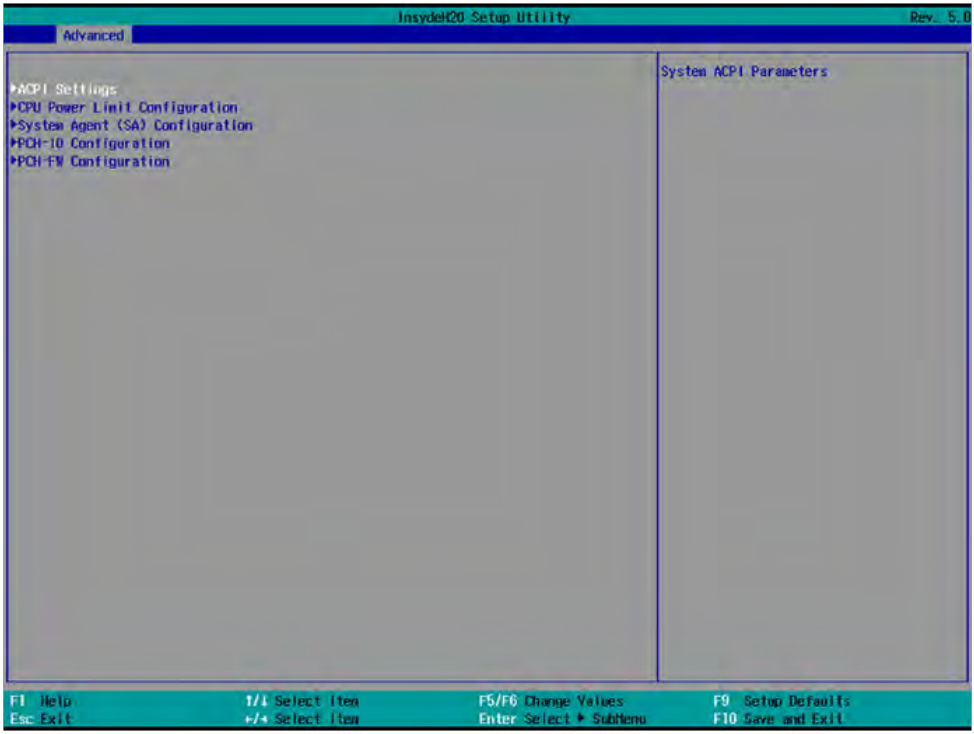
NVM Express information
Please refer section 4-6-4

4-6-1 Boot Configuration



To select Power-on state for NumLock, default is <off>

4-6-2 SOC Config Configuration



ACPI Settings

Please refer section 4-6-2-1

CPU Power Limit Configuration

Please refer section 4-6-2-2

System Agent (SA) Configuration

Please refer section 4-6-2-3

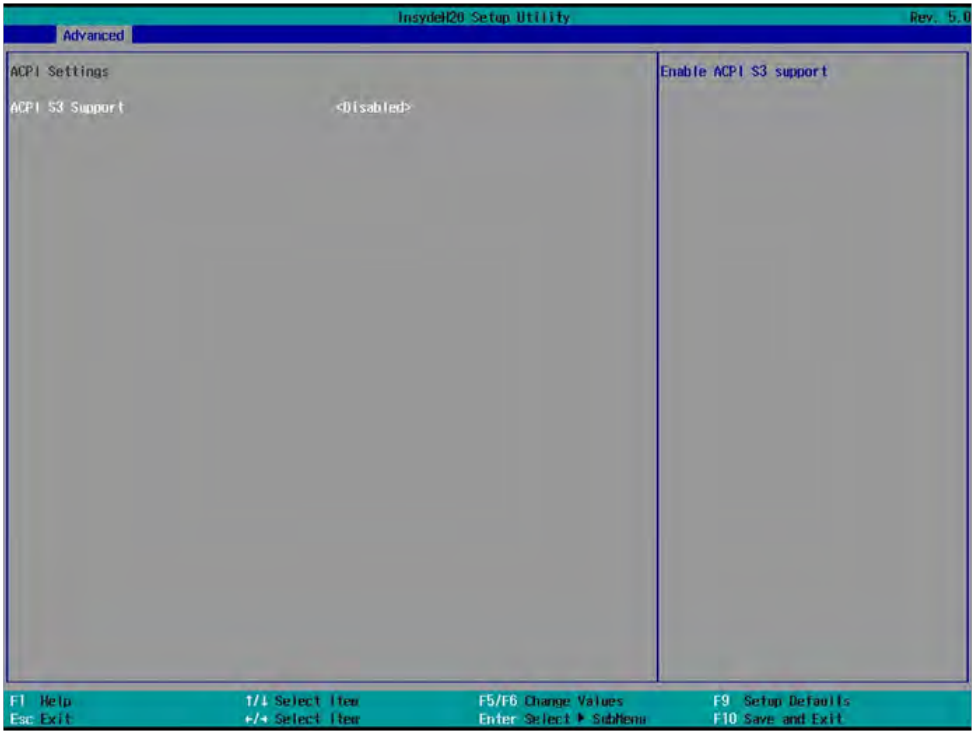
PCH-IO Configuration

Please refer section 4-6-2-4

PCH-FW Configuration

Please refer section 4-6-2-5

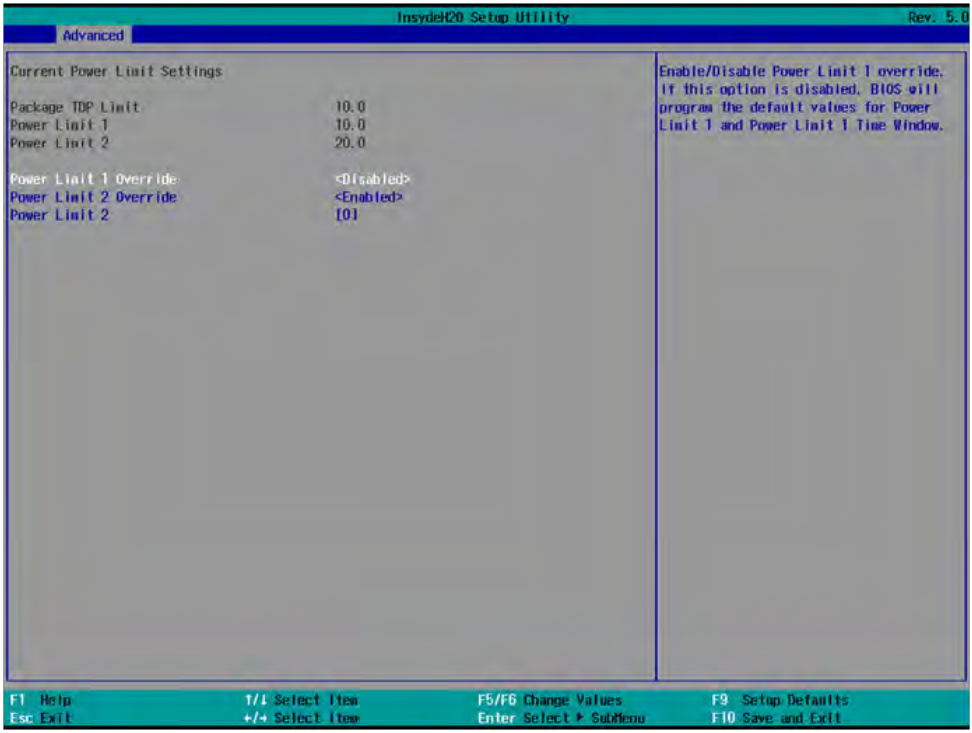
4-6-2-1 ► ACPI Settings



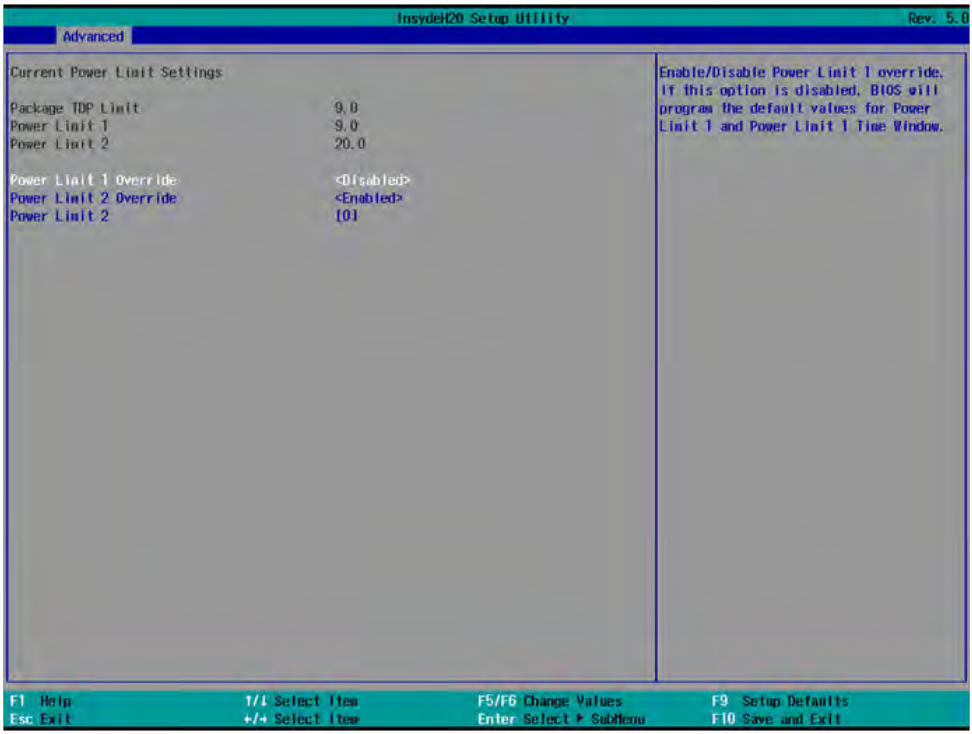
ACPI S3 Support

To enable BIOS support security device or not, default is Enabled.

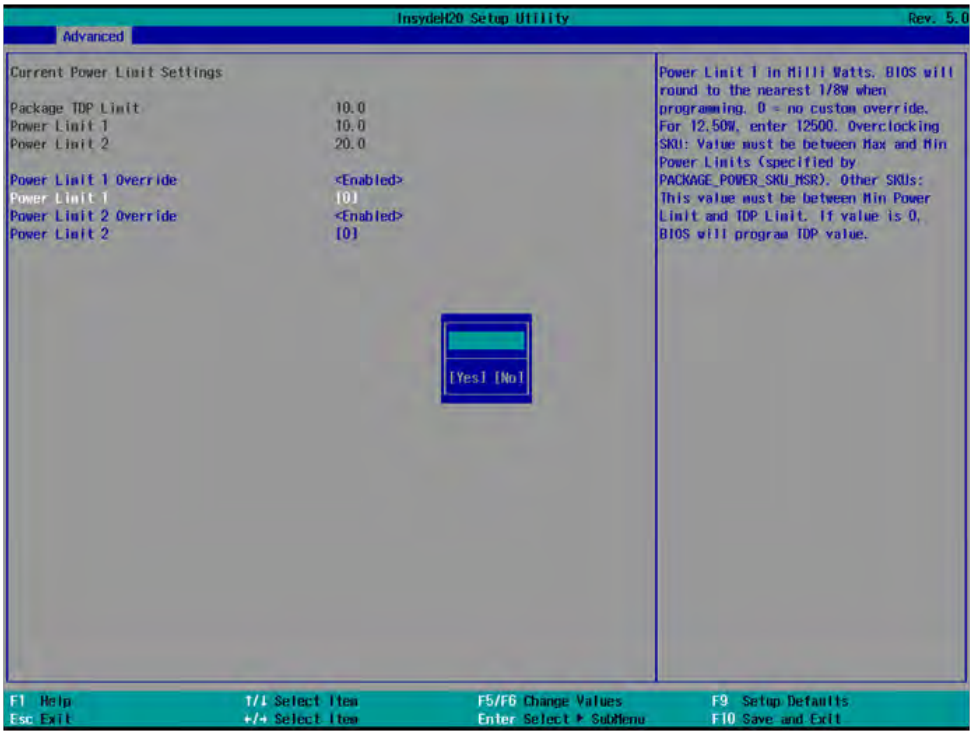
4-6-2-2 ► CPU Power Limit Configuration



The setting follows INTEL Celeron J6412 CPU power limit default configuration.



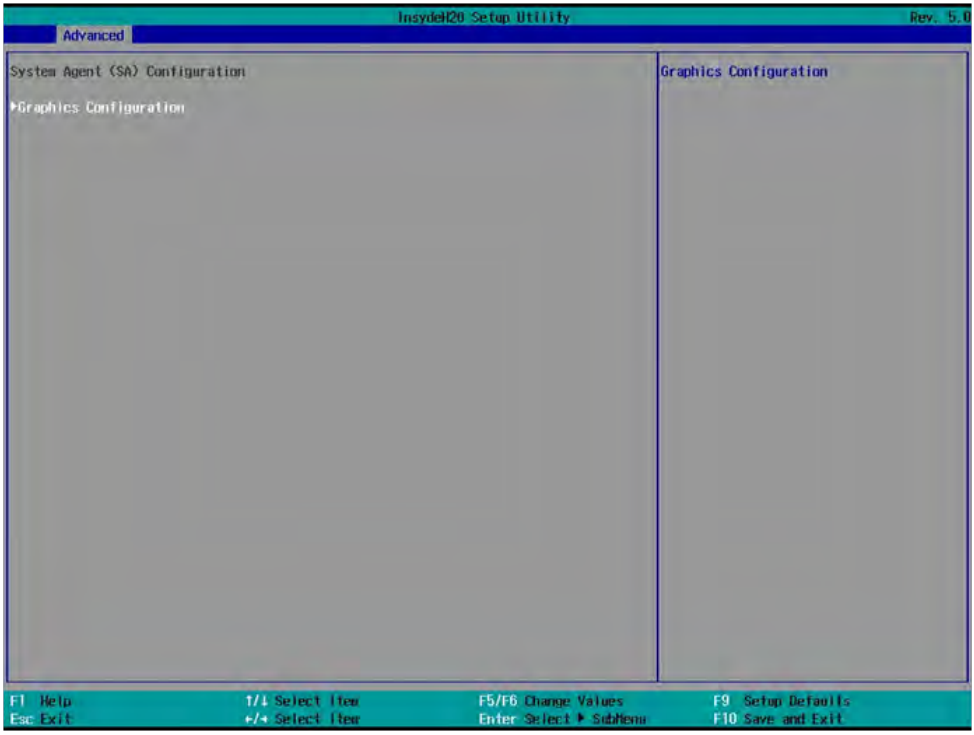
The setting follows INTEL Atom x6413E power limit default configuration.

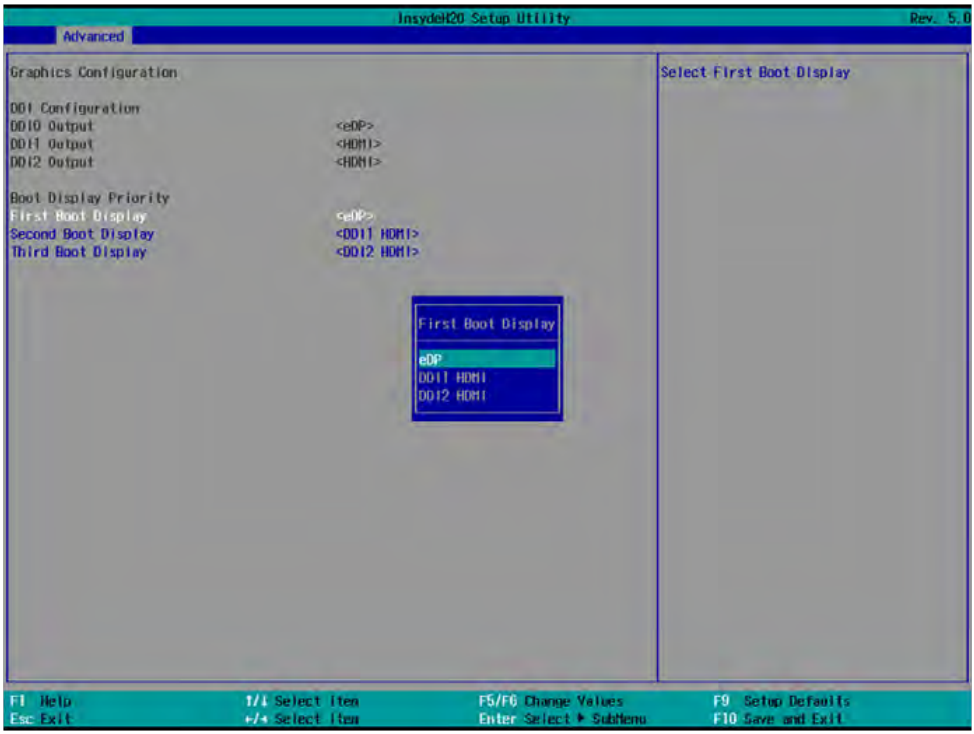


Power Limit Override

Eenable / disable PL1 / PL2 and enter the power numerical value from 0 to 20000 to get higher or lower CPU TDP

4-6-2-3 ► System Agent (SA) Configuration





Boot Display Priority

First Boot Display

To select First Boot Display priority, there are eDP, DDI1 HDMI, DDI2 HDMI, default is eDP

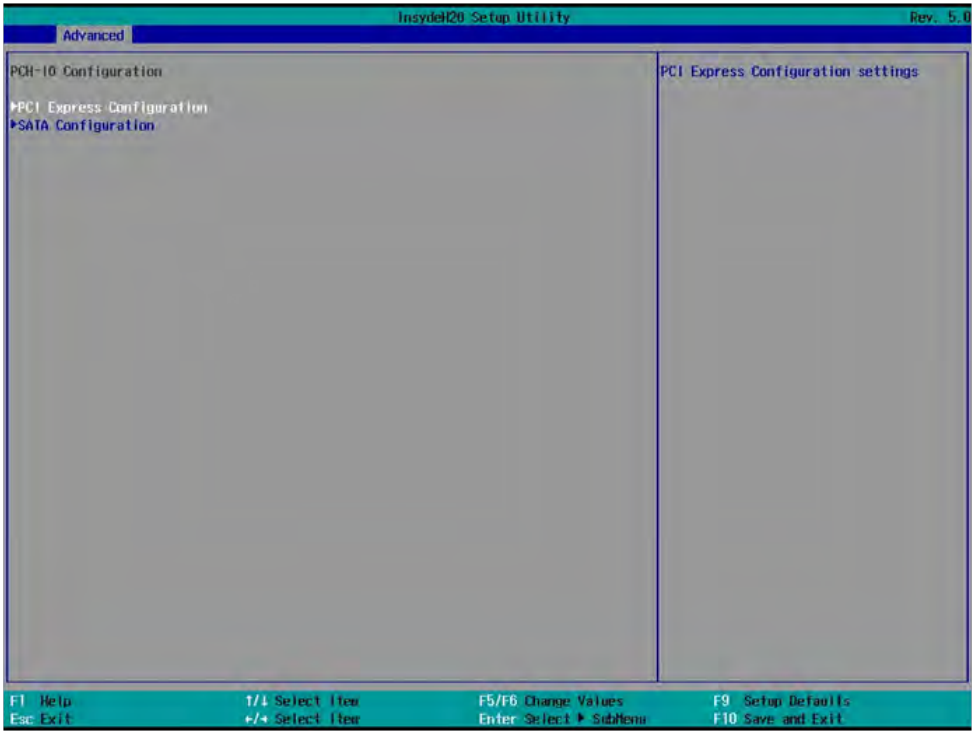
Second Boot Display

To select Second Boot Display priority, there are DDI1 HDMI, DDI2 HDMI, default is DDI1 HDMI

Third Boot Display

To select Third Boot Display priority, there is DDI2 HDMI

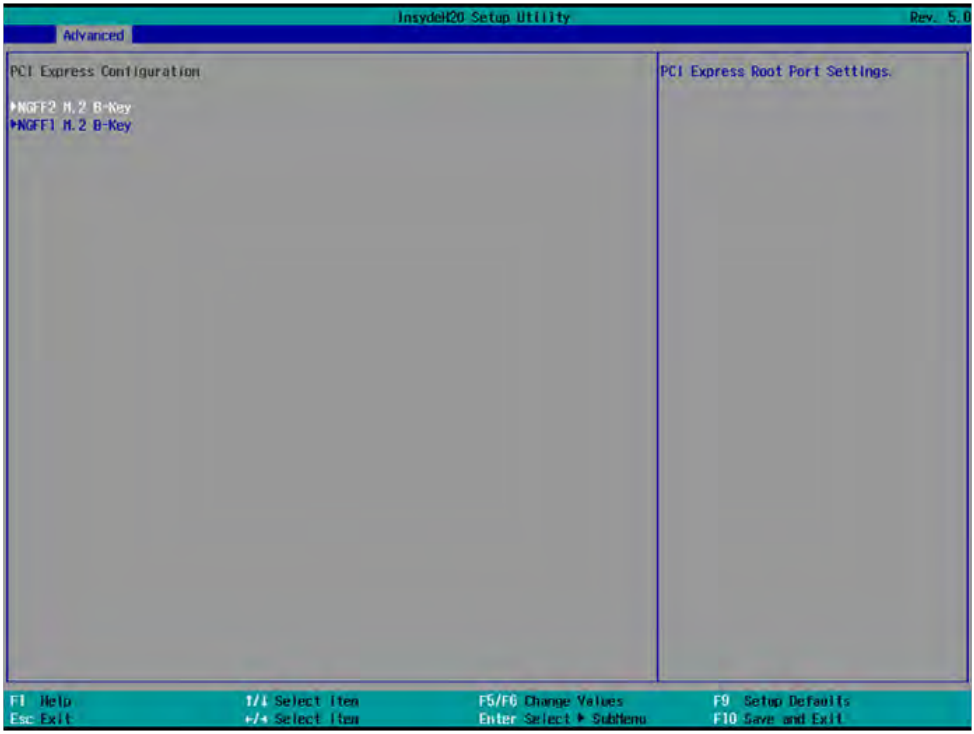
4-6-2-4 ► PCH-IO Configuration

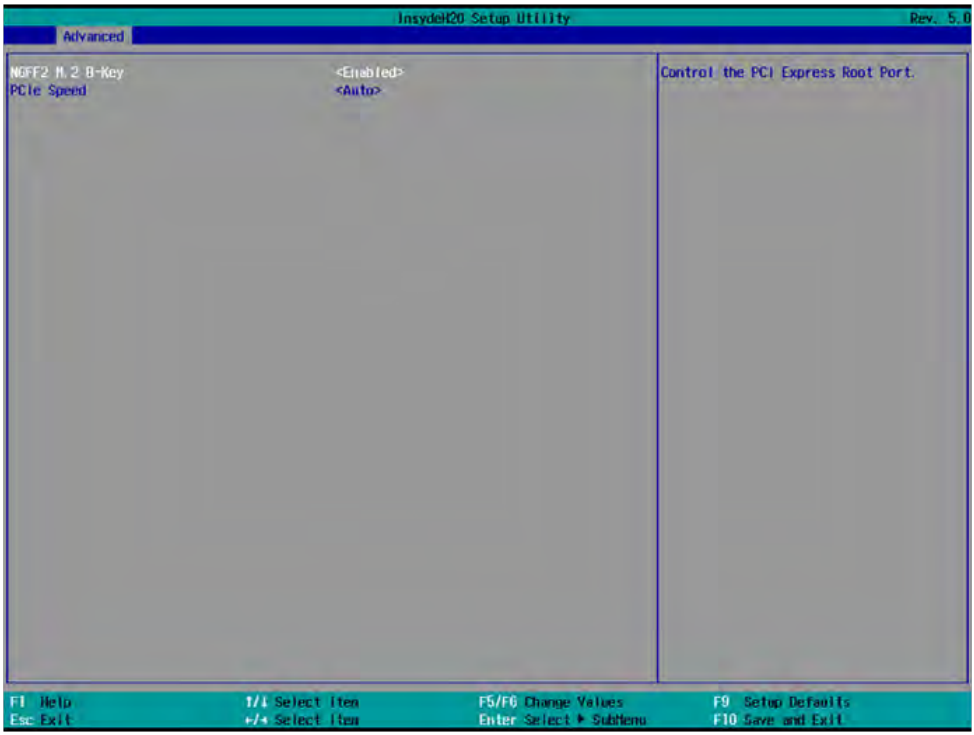


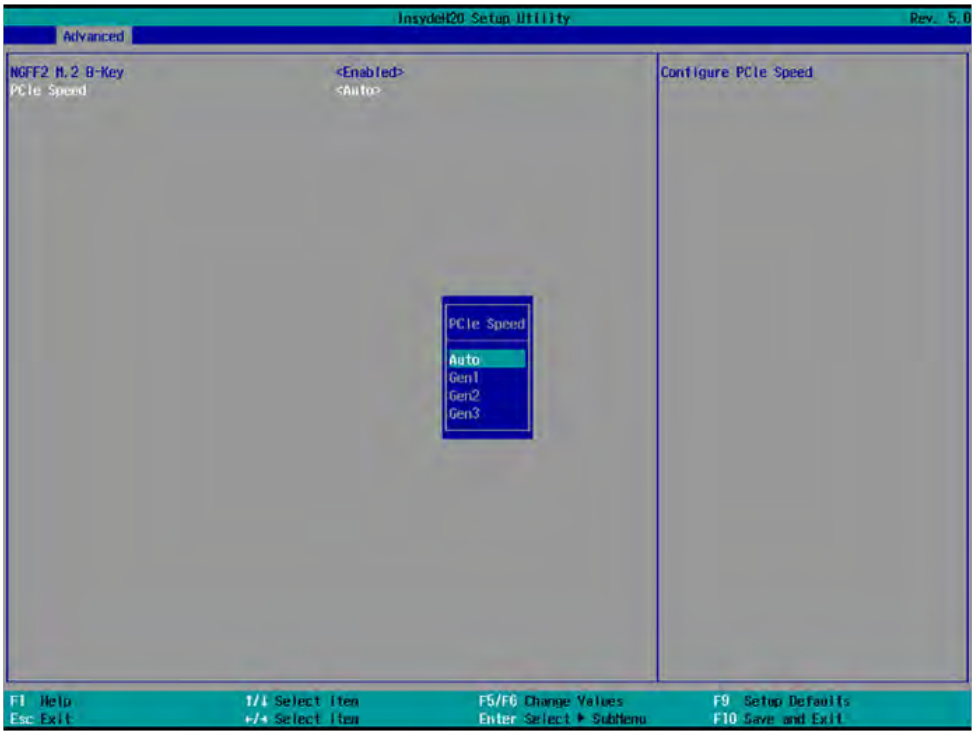
PCI Express Configuration
Please refer section 4-6-2-4-1

SATA Configuration
Please refer section 4-6-2-4-2

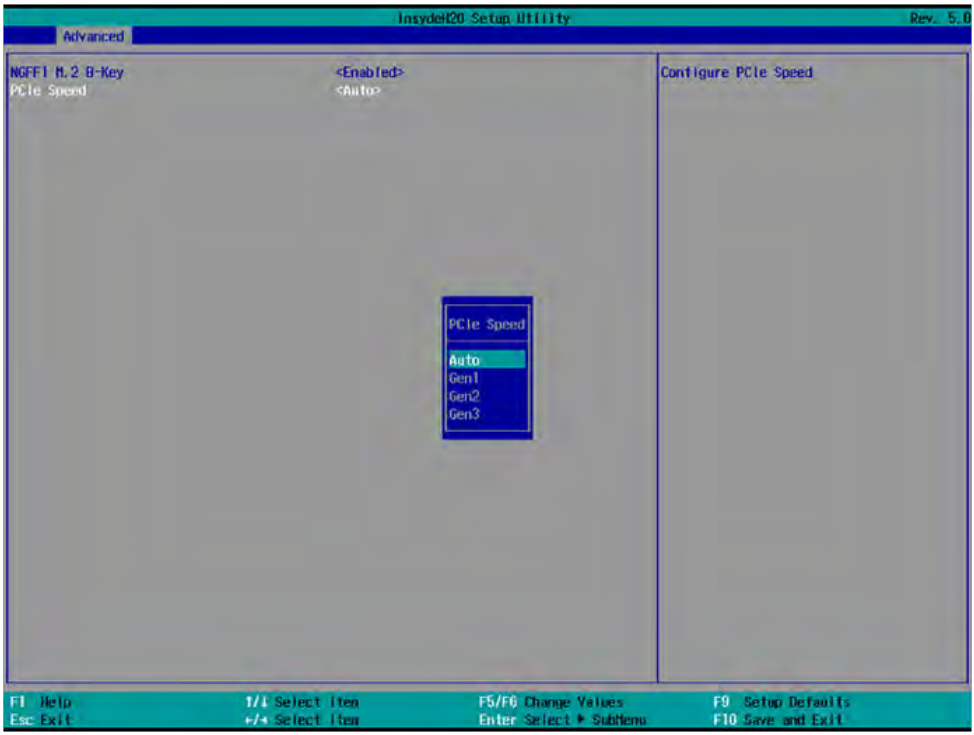
4-6-2-4-1 ► PCI Express Configuration




InsydeH20 Setup Utility
Rev. 5.0
Advanced
M2F2 M.2 B-Key
<Enabled>
PCIe Speed
<Auto>
Control the PCI Express Root Port.
F1 Help
↑/↓ Select Item
F5/F6 Change Values
F9 Setup Defaults
Esc Exit
←/→ Select Item
Enter Select ▶ SubMenu
F10 Save and Exit

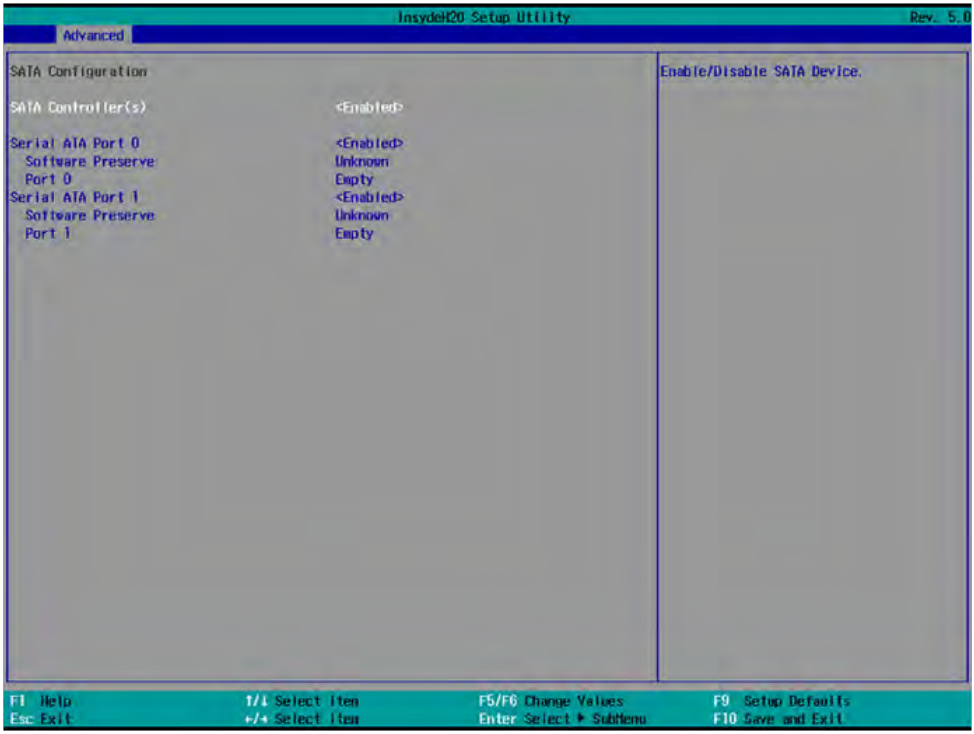


To select NGFF2 device enabled or not and to change the PCIe Speed, there are Auto, Gen1, Gen2, Gen3, default is Auto



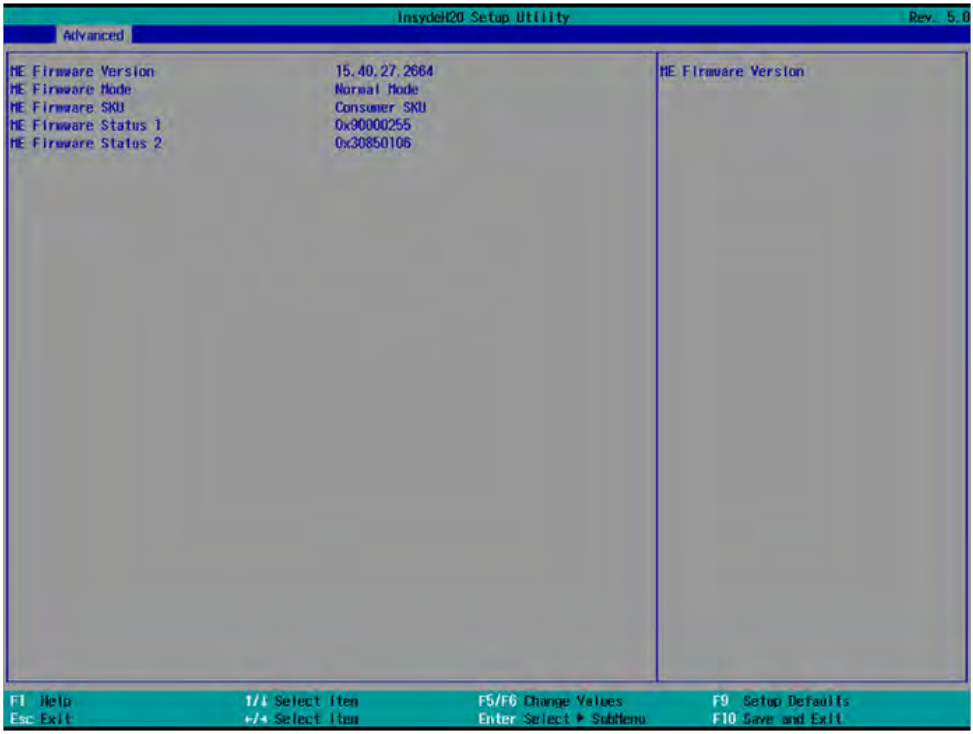
To select NGFF1 device enabled or not and to change the PCIe Speed, there are Auto, Gen1, Gen2, Gen3, default is Auto

4-6-2-4-2 ► SATA Configuration

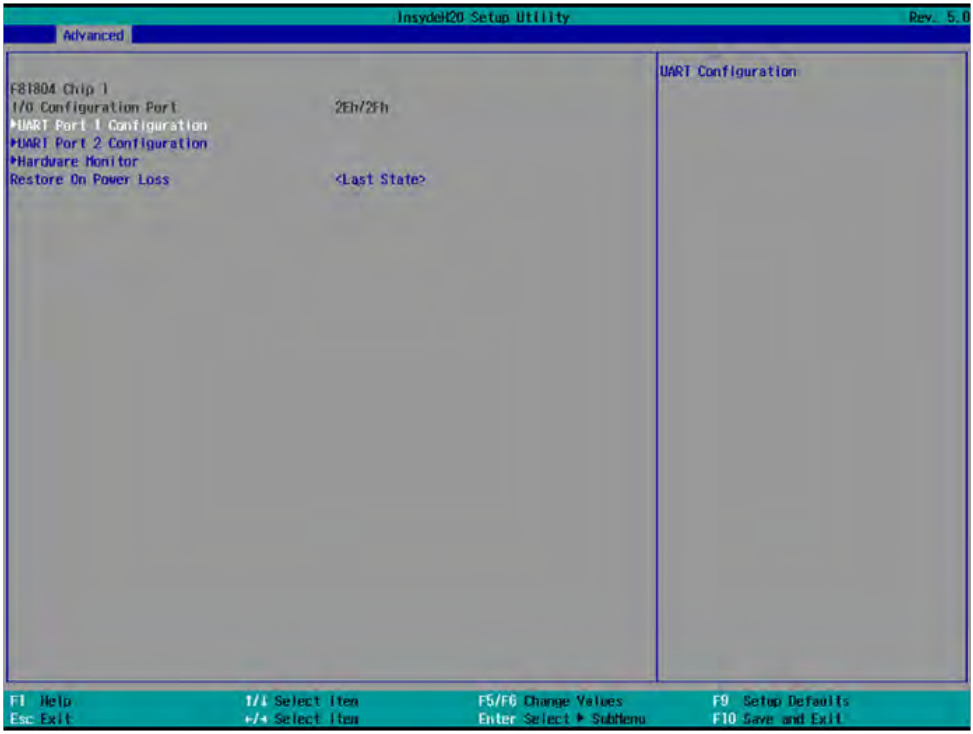


To select SATA port & NGFF2 M.2 SATA device enabled or not.

4-6-2-5 ► PCH-FW Configuration



4-6-3 SIO F81804



UART Port 1 Configuration

Please refer section 4-6-3-1

UART Port 2 Configuration

Please refer section 4-6-3-2

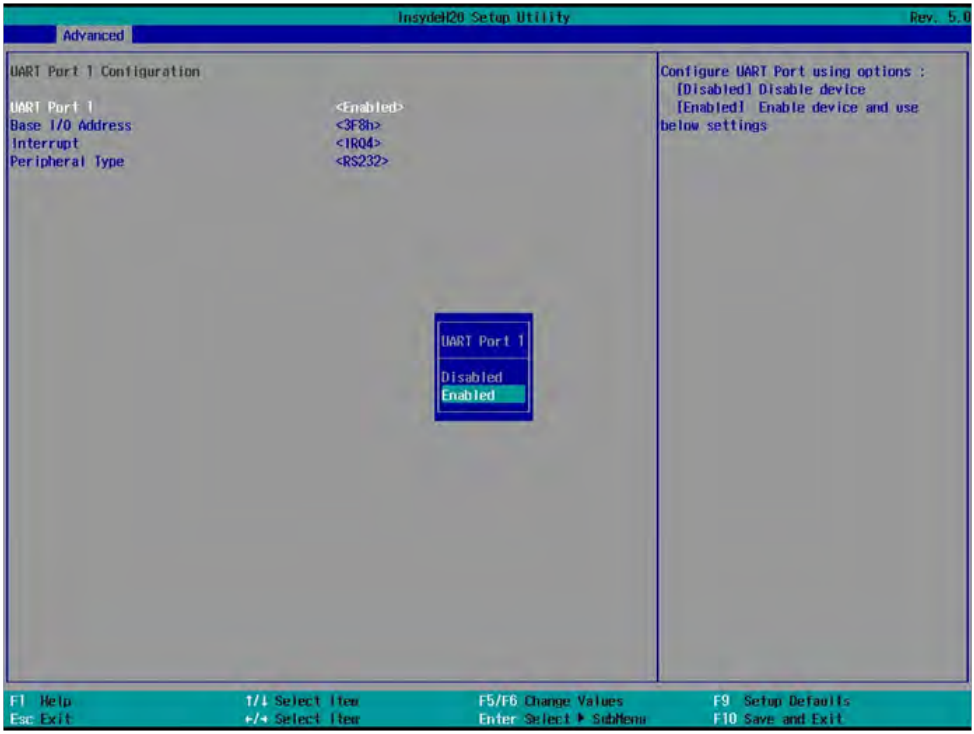
Hardware Monitor

Please refer section 4-6-3-3

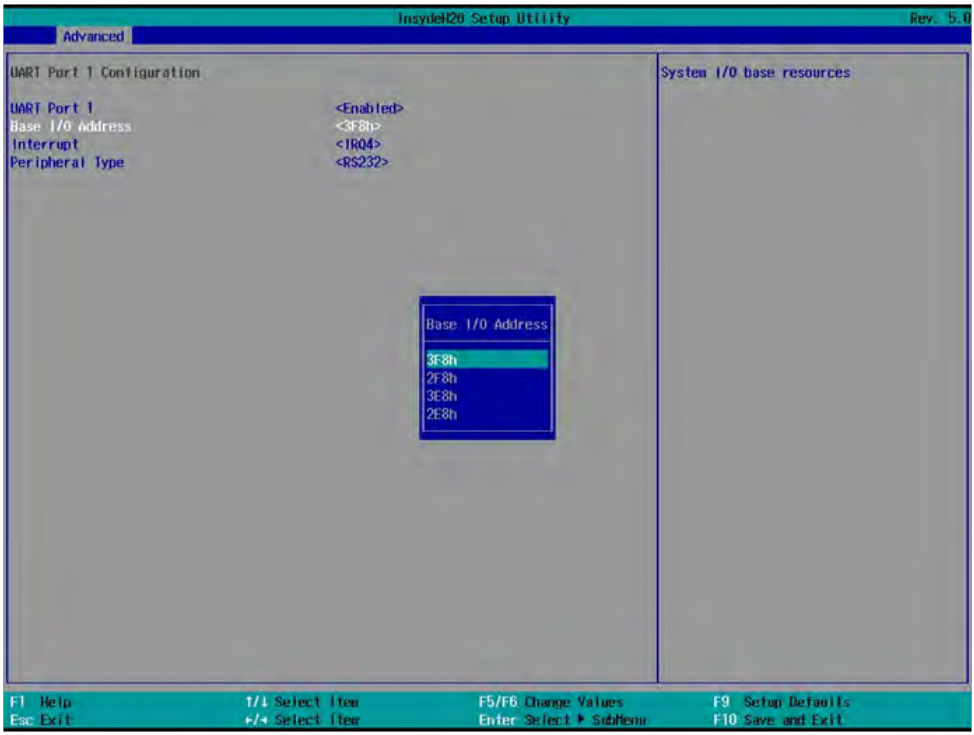
Restore on Power Loss

Please refer section 4-6-3-4

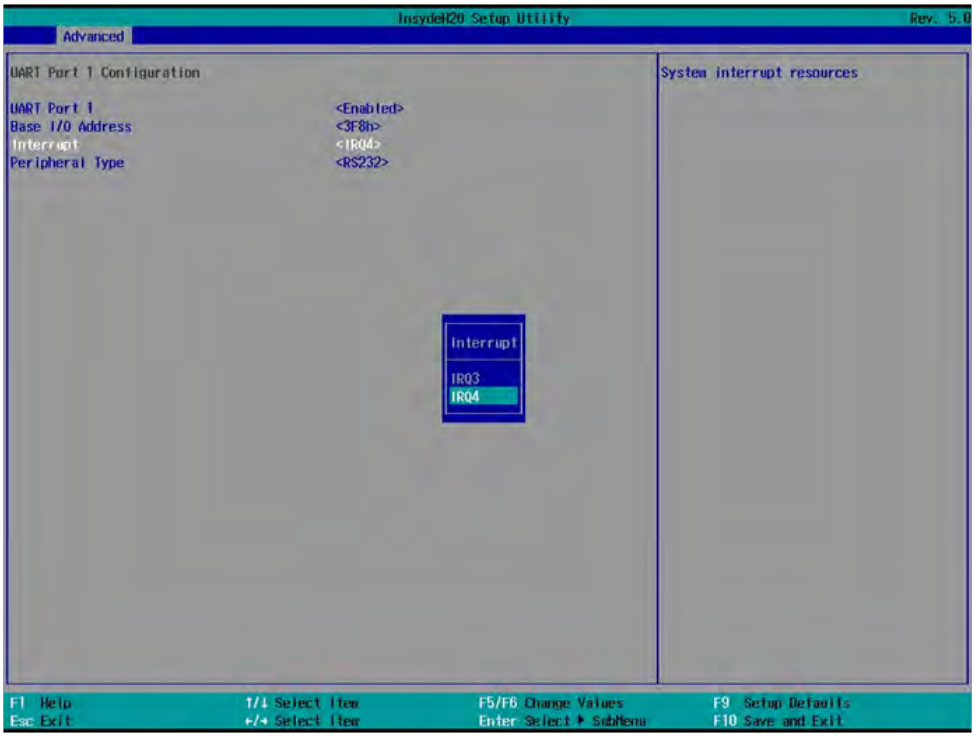
4-6-3-1 ► UART Port 1 Configuration



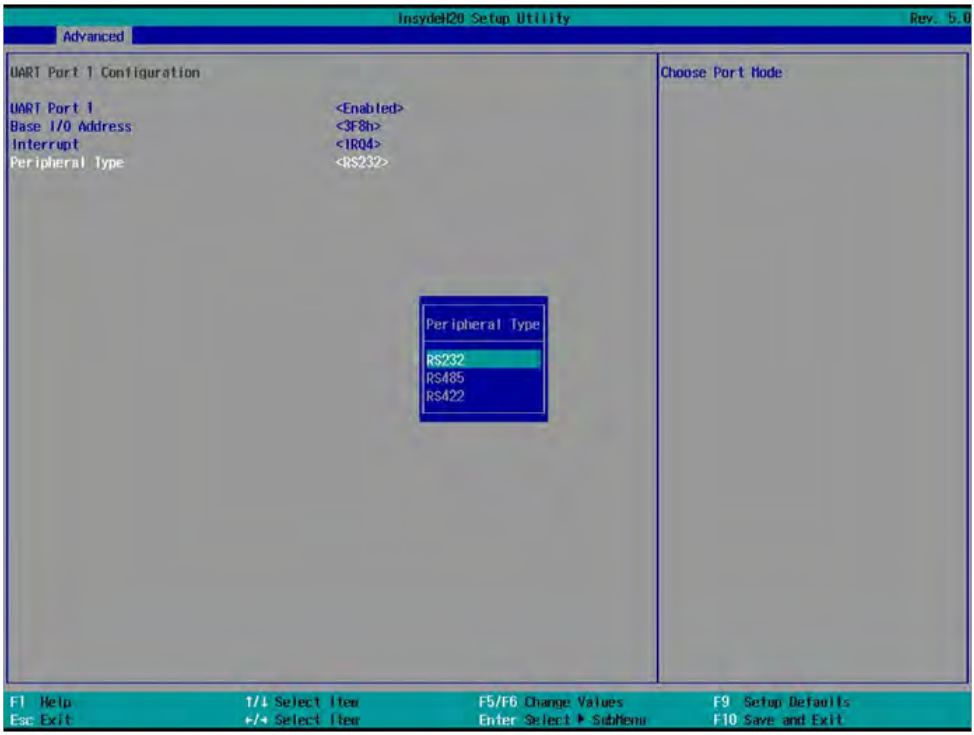
To Enable Serial port or not, default is Enabled.



Base I/O Address, default is 3F8h.

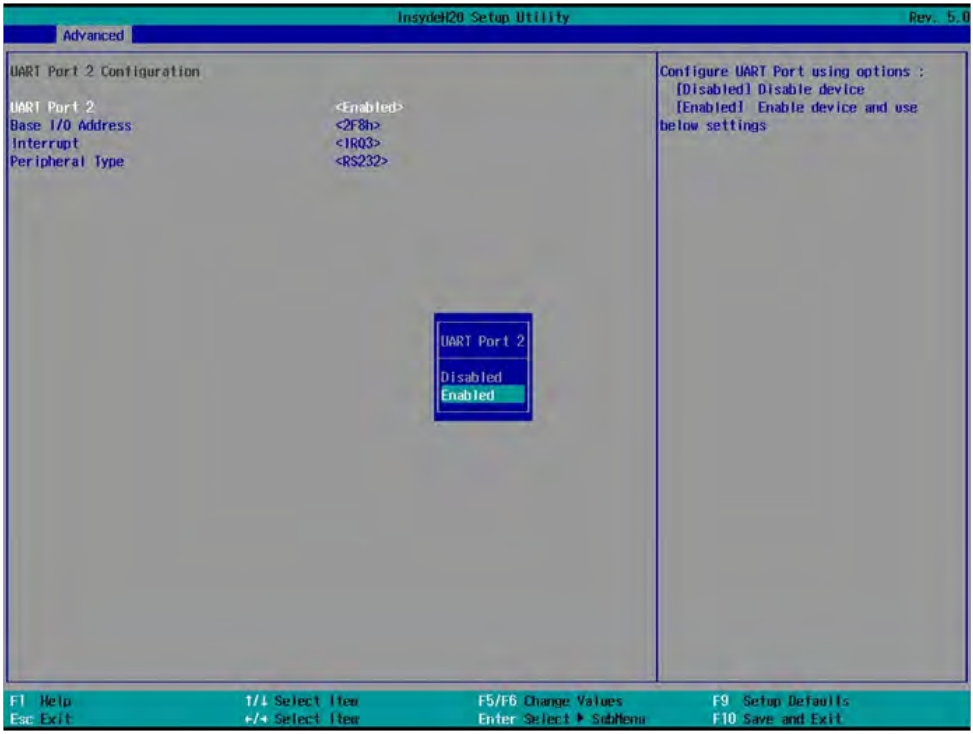


Interrupt, default is IRQ4.

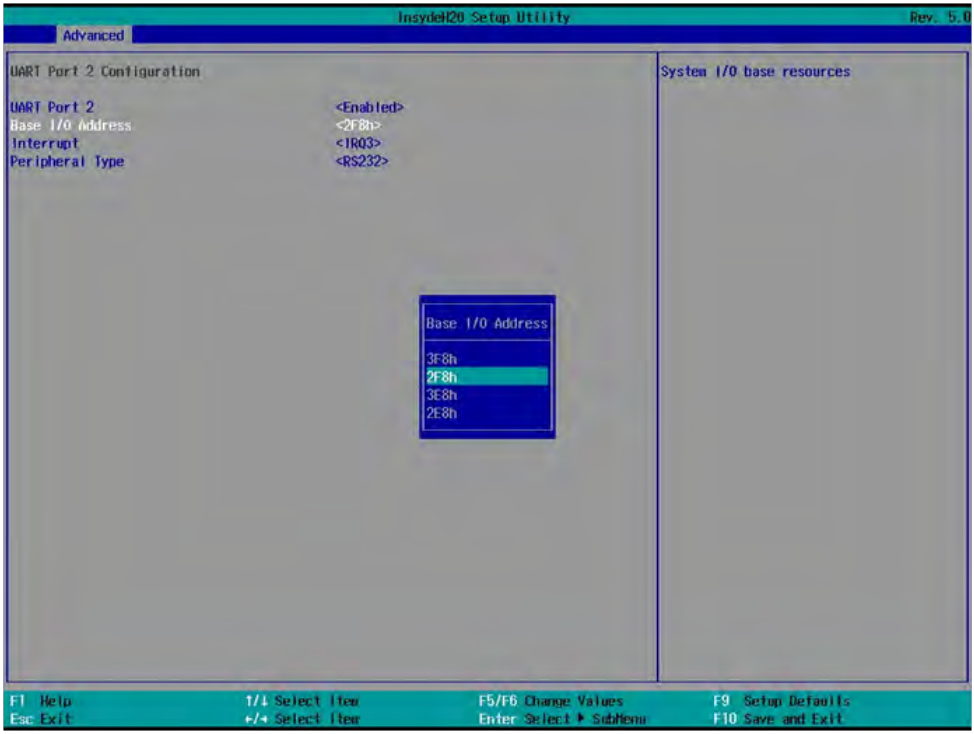


Peripheral, to select the Serial port to RS232 / RS422 / RS485, default is RS232.

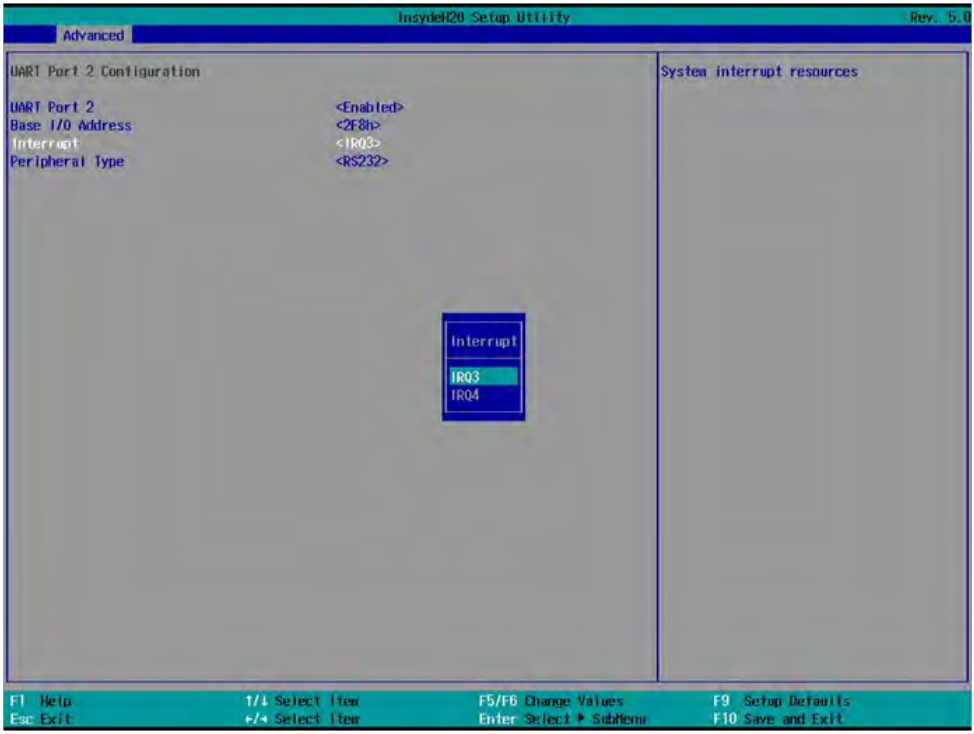
4-6-3-2 ► UART Port 2 Configuration



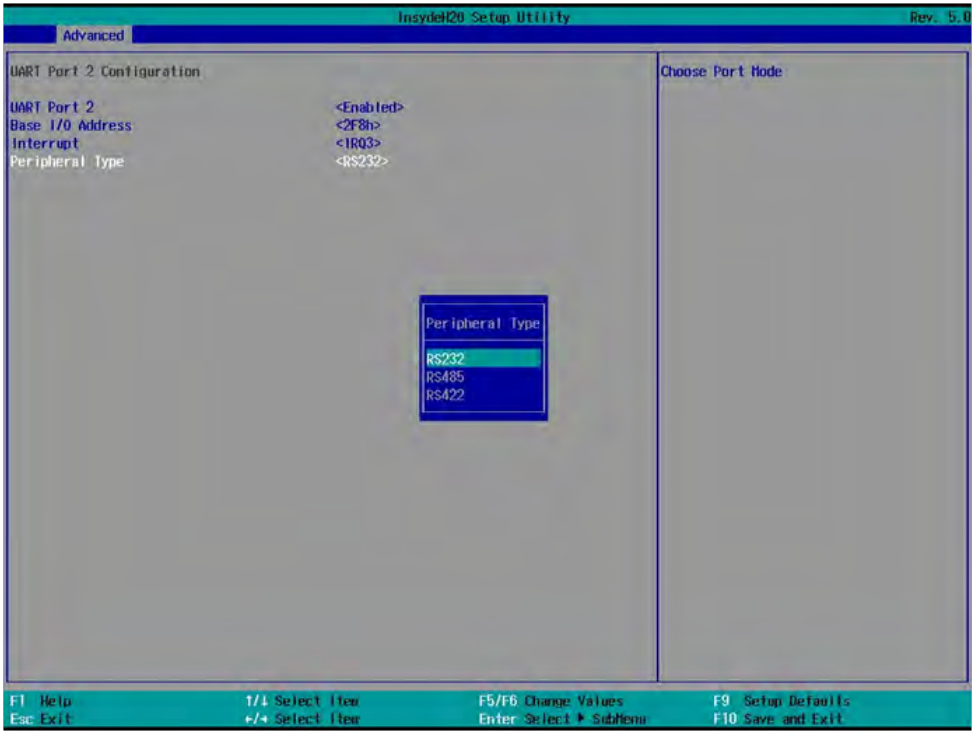
To Enable Serial port or not, default is Enabled.



Base I/O Address, default is 2F8h.



Interrupt, default is IRQ3.



Peripheral, to select the Serial port to RS232 / RS422 / RS485, default is RS232.

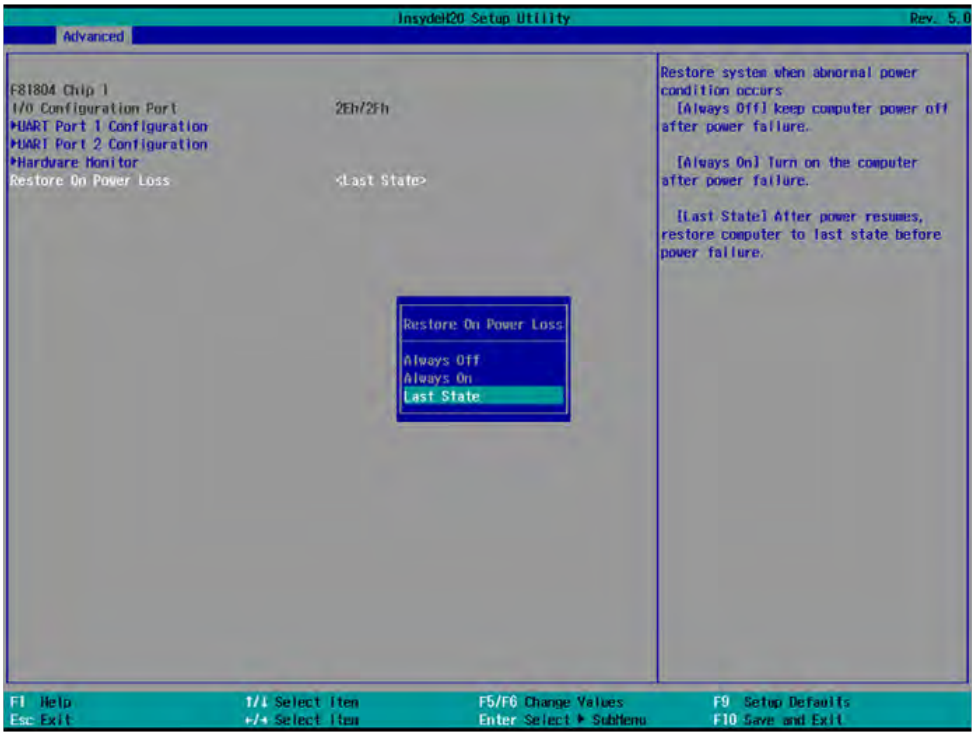
4-6-3-3 ► Hardware Monitor

Press [Enter] to view PC health status.

This section shows the status of your CPU, Fan, and overall system.

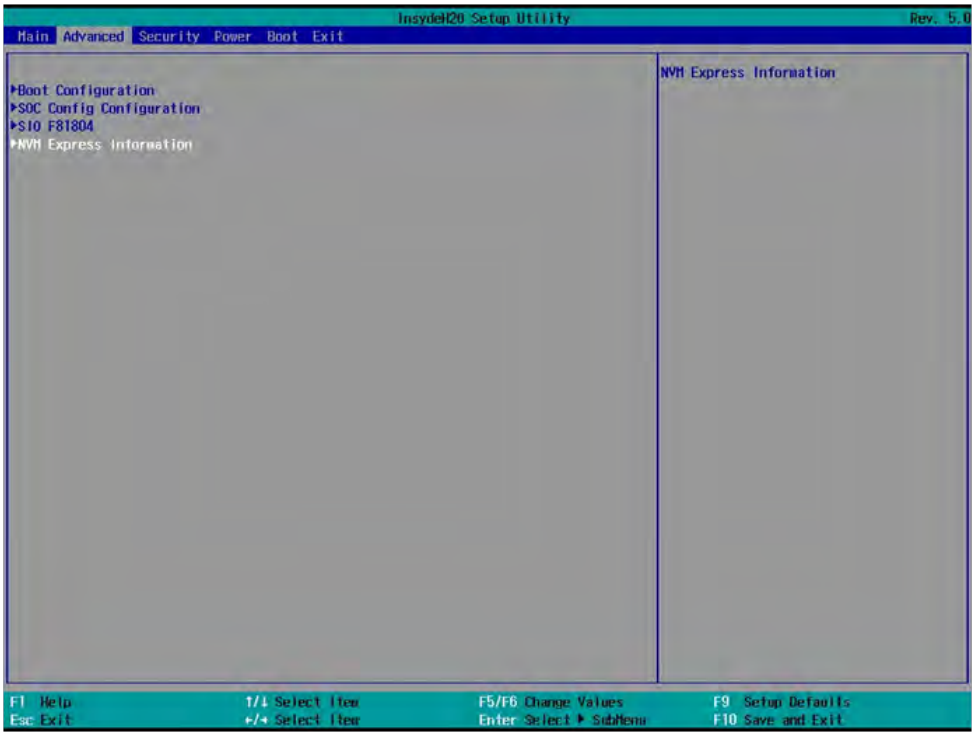
This is only available when there is Hardware Monitor function onboard.

4-6-3-4 Restore On Power Loss



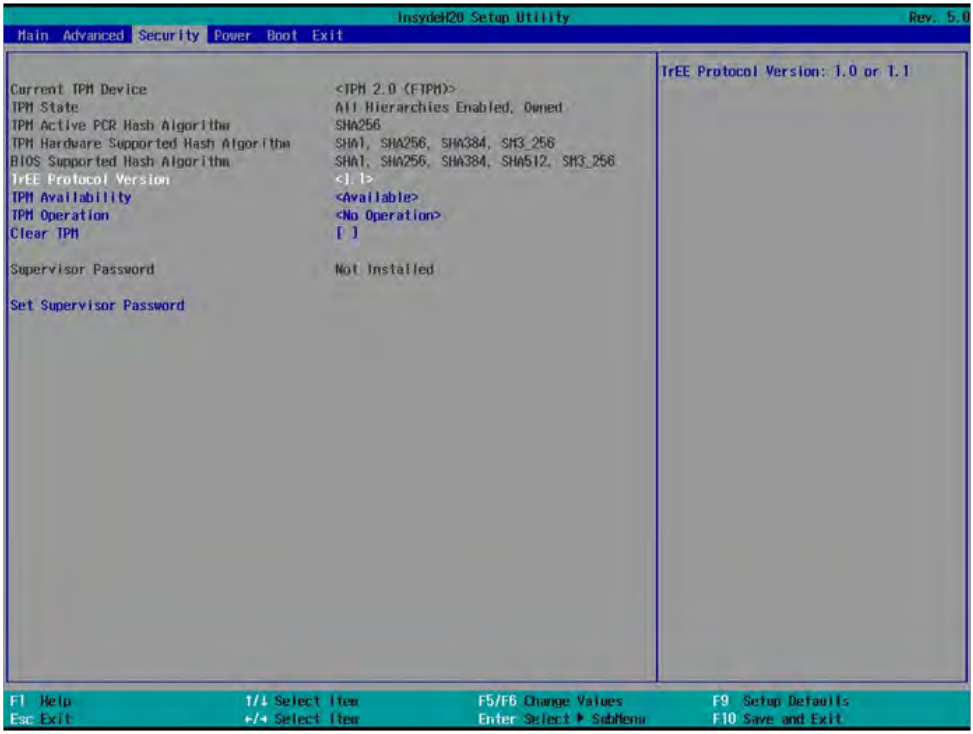
To select the power behavior after power fail, default is last state.

4-6-4 NVM Express Information



Press [Enter] to view the NVMe storage devices information.

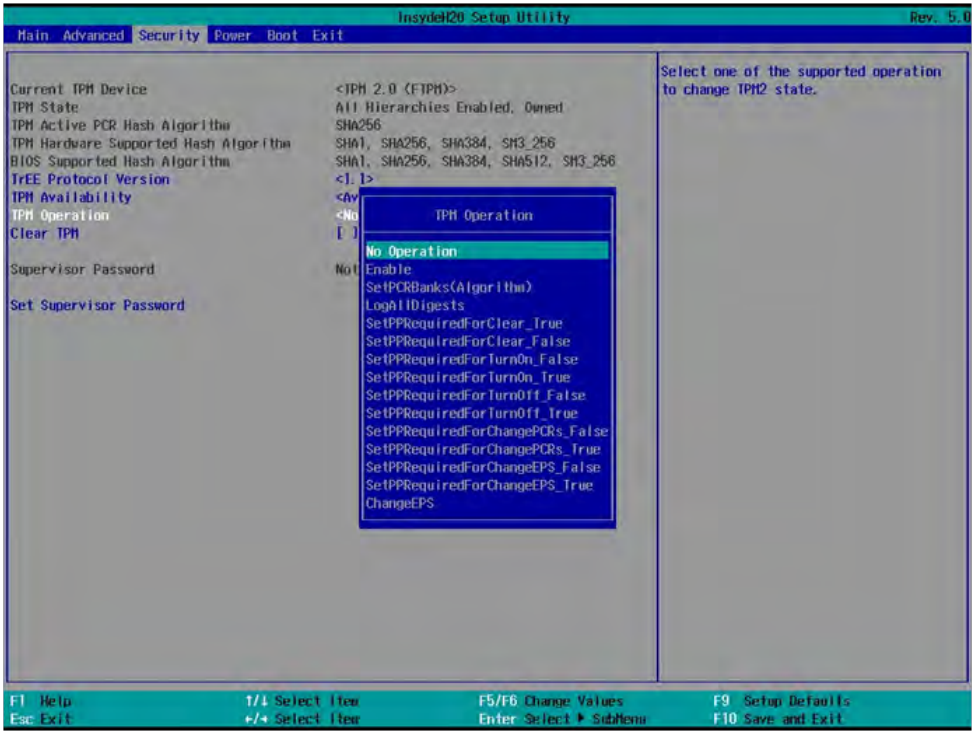
4-7 Security



TrEE Protocol Version
There are 1.0 and 1.1 versions.

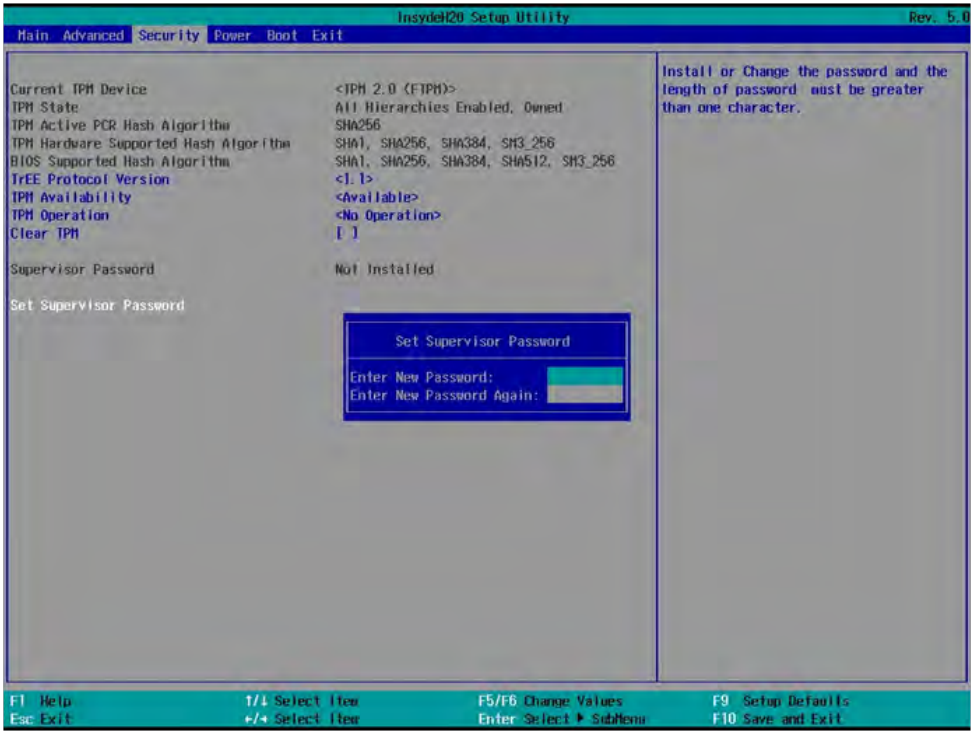
TPM Availability
To select TPM available or hidden

TPM Operation



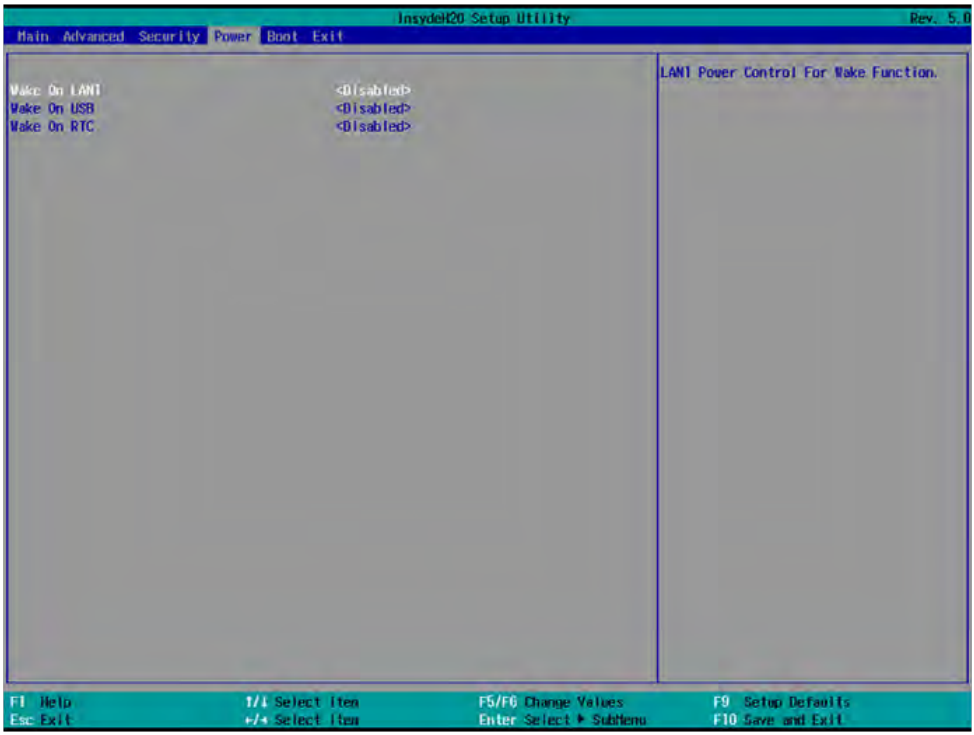
To select TPM operations

Set Supervisor Password



To set up an Supervisor password

4-8 Power

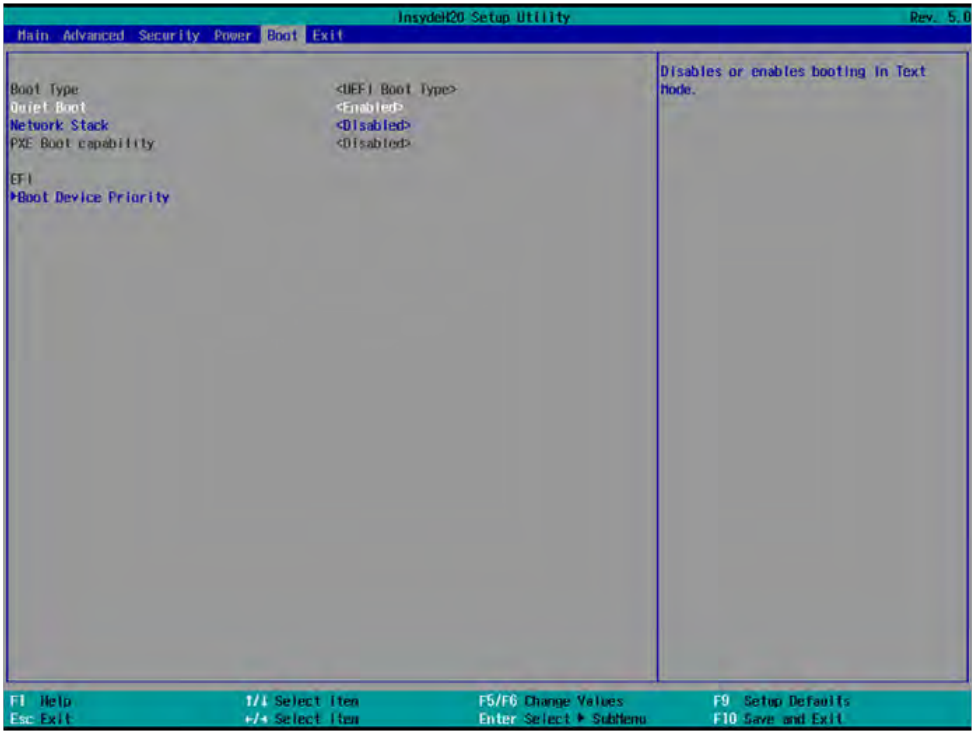


Wake On LAN1
To select S3, S5 or S3 / S5 wake on LAN1, default is Disabled.

Wake On USB
To select S3 wake on USB, default is Disabled.

Wake On RTC
The optional settings are: Disabled (default), By every day, By day of month.

4-9 Boot

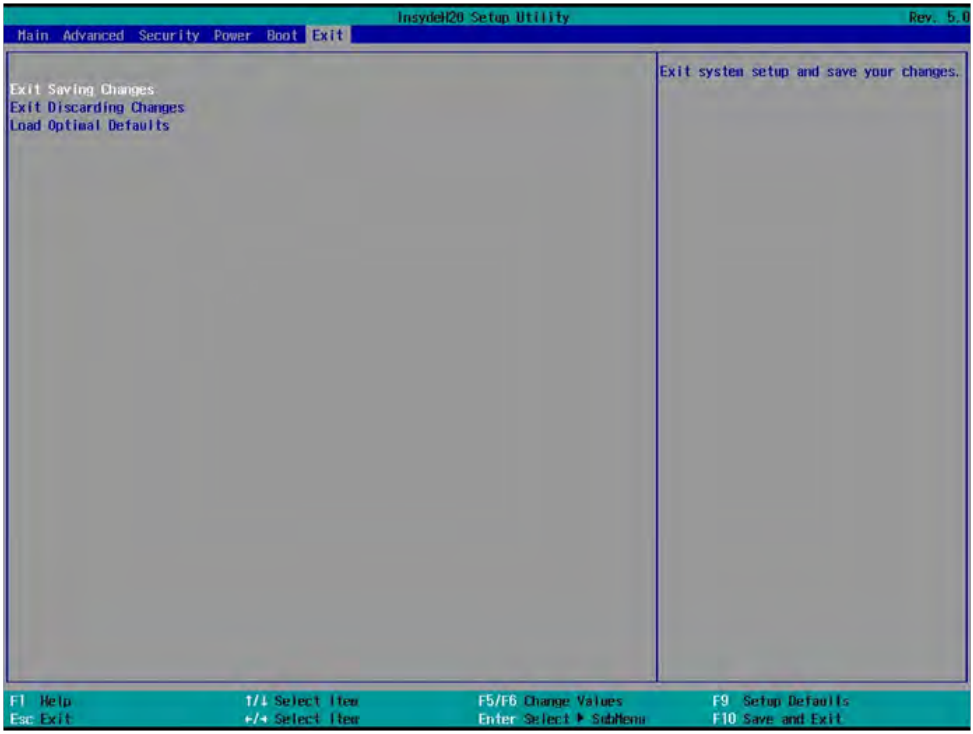


Quiet Boot
The optional settings are: Enabled (default), Disabled.

Network Stack
The optional settings are: Enabled, Disabled (default).

EFI Boot Device Priority
Determine which EFI storage device for booting, this item will not show on this page if there is no any storage device found.

4-10 Save & Exit



Exit Saving Changes
Save configuration and reset

Exit Discarding Changes
Reset without saving the changes

Load Optimal Defaults
To restore the optimal default for all the setup options

4-11 How to update Insyde BIOS

Under DOS Mode

STEP 1. Prepare a bootable disc.
(Storage device could be USB FDD or USB pen drive.)

STEP 2. Copy utility program to your bootable disc. You may download it from our website.

STEP 3. Copy the latest BIOS for your LEX motherboard from our website to your bootable disc.

STEP 4. (Here take 2I640HL as an example, please enter your motherboard's name)
Insert your bootable disc into X: (X could be C:, A: or others.
It depends on which type of storage device you use.)

Start the computer and type

X:\: H2OFFT-D.EXE 2I640HLA2.ROM –BIOS -ALL

2I640HLA2.ROM is the file name of the latest BIOS.

It may be 2I640HLA1.ROM or 2I640HLA2.ROM, etc.

Please leave one space between .ROM & -BIOS -ALL

By Bay Trail series mainboard, please type

X:\: H2OFFT-D.EXE 2I640HLA2.ROM –BIOS -ALL

-BIOS : Flash BIOS region

-ALL : Flash all

STEP 5. Press ENTER and the BIOS will be updated,
Computer will restart automatically.

Appendix B: Resolution list

640 x 480 x (256 / 16bit / 32bit)
800 x 600 x (256 / 16bit / 32bit)
1024 x 768 x (256 / 16bit / 32bit)
1152 x 864 x (256 / 16bit / 32bit)
1280 x 600 x (256 / 16bit / 32bit)
1280 x 720 x (256 / 16bit / 32bit)
1280 x 768 x (256 / 16bit / 32bit)
1280 x 800 x (256 / 16bit / 32bit)
1280 x 960 x (256 / 16bit / 32bit)
1280 x 1024 x (256 / 16bit / 32bit)
1400 x 1050 x (256 / 16bit / 32bit)
1440 x 900 x (256 / 16bit / 32bit)
1600 x 900 x (256 / 16bit / 32bit)
1600 x 1200 x (256 / 16bit / 32bit)
1680 x 1050 x (256 / 16bit / 32bit)
1920 x 1080 x (256 / 16bit / 32bit)
1920 x 1200 x (256 / 16bit / 32bit)