MB838 series

4-port (Rangeley SoC)
Networking Motherboard

USER'S MANUAL

Version A1

Acknowledgments

AMI is a registered trademark of American Megatrends, Inc.
PS/2 is a trademark of International Business Machines Corporation.
Intel is a registered trademark of Intel Corporation.
Microsoft Windows is a registered trademark of Microsoft Corporation.
Winbond is a registered trademark of Winbond Electronics Corporation.
All other product names or trademarks are properties of their respective owners.

Table of Contents

Introduction 5

Product Description..5
Specifications 6
Checklist..8
Board Dimensions 9
Block Diagram 10

Installations 11

Installing the Memory 12
Setting the Jumpers 13
Jumper Locations on MB838... 14
JP2: Function reserved... 15
JP3: DDR power selection... 15
JP4: AT / ATX mode selection... 15
JP7: Clear CMOS 15
JP8: LED5, LED6 Bypass or HDD LED selection............ 15
JP12: Function reserved.. 15
Connectors on MB838 16
J3: System Function Connector 17
J8: Power connector for ATX power supply 18
J9: 2-pin Power connector for 12V DC power 18
J11: USB pin header... 18
J12: COM2 pin header.. 18
J28: Power Connector, Pitch 2.0mm................................. 19
CN1: 22-pin right angle SATA connector 19
CN2, CN3: RJ-45 LAN port 3 and 4 19
CN4: USB 2.0 connectors... 19
CN5: COM1 (Console Port) 19
CN6, CN7: RJ-45 LAN port 1 and 2 19
FAN1, FAN2, FAN3: System Fan Connectors.................. 19
LED1: LAN port 1 LED.. 19
LED2: LAN port 2 LED.. 19
LED3: LAN port 3 LED.. 19
LED4: LAN port 4 LED.. 19
LED5, 6: Status LED... 20
SW2: Push button (GPI, porting by software) 20

Digital I/O Sample Configuration 21

Watchdog Timer Configuration 25

BIOS Setup 29

BIOS Introduction 30
BIOS Setup 30
Advanced Settings 31
Security 41

Drivers Installation 44

LAN Drivers Installation 45
A. I/O Port Address Map 48
B. Interrupt Request Lines (IRQ) 48
C. Register of the LAN Bypass Controller 49

Introduction

Product Description

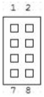
The MB838 networking motherboard is based on the latest Intel® Atom™ C2000 series SoC. Two DDR3 UDIMM sockets allows up to 32GB system memory.

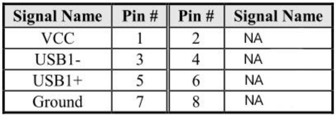
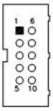
The motherboard supports a total of eight Ethernet ports with the port 1 and port2 supporting Bypass function. It also has one fast SATA III 6Gbps ports, two USB 2.0 ports and one USB 2.0 for Mini PCI-e. MB838 utilizes the dramatic increase in performance provided this Intel's latest cutting-edge technology. Measuring 203mm x 180mm.

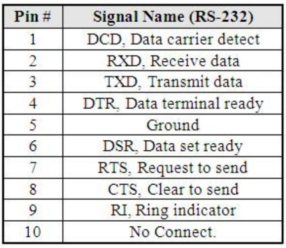
MB838 Features

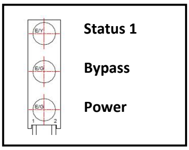
- Supports Intel® Atom™ C2000 series processors
- Two DDR3 UDIMM for maximum 32GB total, 1600MHz, Unbuffered
- Eth1~2: Intel® I347-AT4 GbE PHY, support Bypass
  Eth3~4: Intel® I347-AT4 GbE PHY, No Bypass
- One SATA III (6.0Gb/s)
- One Mini PCI-E socket (mSATA compatible)

Specifications

<table>
<tr><td><b>Processor</b></td><td>Support Intel® Rangeley processors:
<table>
<tr><th></th><th>Core#</th><th>SoC</th><th>TDP</th><th>QuickAssist</th></tr>
<tr><td>MB838-2C</td><td>2</td><td>Intel® Atom™ Processor C2358
(1M Cache, 1.70 GHz)</td><td>7W</td><td>Yes</td></tr>
<tr><td>MB838-4C</td><td>4</td><td>Intel® Atom™ Processor C2558
(2M Cache, 2.40 GHz)</td><td>15W</td><td>Yes</td></tr>
<tr><td>MB838-8C</td><td>8</td><td>Intel® Atom™ Processor C2758
(4M Cache, 2.40 GHz)</td><td>20W</td><td>Yes</td></tr>
</table></td></tr>
<tr><td><b>BIOS</b></td><td>AMI BIOS</td></tr>
<tr><td><b>Memory</b></td><td>• Two DDR3/L UDIMM total for 32GB max memory
• Dual channel DDR3 up to 1600 MHz
• Unbuffered
• ECC or non-ECC
• Support 1.35V and 1.5V</td></tr>
<tr><td><b>Video</b></td><td>NA</td></tr>
<tr><td><b>Network PHY</b></td><td>• Eth1~2: Intel® I347-AT4 GbE PHY. One segment Bypass
• Eth3~4: Intel® I347-AT4 GbE PHY. No Bypass</td></tr>
<tr><td><b>Network Bypass</b></td><td>• One segment hardware Bypass (Eth1 & 2)
Control by GPIO / Watchdog / Electrical Disconnect (Power Off)
• Bypass mode selection (BIOS)</td></tr>
<tr><td><b>SATA Connector</b></td><td>• SATA III (6.0Gb/s): One to mSATA
One to right angle 22-pin SATA connector
• SATA II (3.0Gb/s): One to SATA, Black connector
One to golden finger 1</td></tr>
<tr><td><b>Storage</b></td><td>• One onboard 22-pin SATA connector support 2.5” HDD
• One mSATA (Mini PCI-e socket)</td></tr>
<tr><td><b>USB 2.0</b></td><td>• One to Mini PCI-e
• Two to Front panel
• One 2x4 pin header support one USB port



<table>
<tr><th>Signal Name</th><th>Pin #</th><th>Pin #</th><th>Signal Name</th></tr>
<tr><td>VCC</td><td>1</td><td>2</td><td>NA</td></tr>
<tr><td>USB1-</td><td>3</td><td>4</td><td>NA</td></tr>
<tr><td>USB1+</td><td>5</td><td>6</td><td>NA</td></tr>
<tr><td>Ground</td><td>7</td><td>8</td><td>NA</td></tr>
</table></td></tr>
<tr><td><b>LPC I/O</b></td><td>Fintek F81866AD-I:
• RJ-45 Console x1
• Hardware monitors
• 4-pin Smart Fan (PWM) Connector x3
• RS-232 [2x5] Box Header Onboard x1 (COM2, located close to LCM)



<table>
<tr><th>Pin #</th><th>Signal Name (RS-232)</th></tr>
<tr><td>1</td><td>DCD, Data carrier detect</td></tr>
<tr><td>2</td><td>RXD, Receive data</td></tr>
<tr><td>3</td><td>TXD, Transmit data</td></tr>
<tr><td>4</td><td>DTR, Data terminal ready</td></tr>
<tr><td>5</td><td>Ground</td></tr>
<tr><td>6</td><td>DSR, Data set ready</td></tr>
<tr><td>7</td><td>RTS, Request to send</td></tr>
<tr><td>8</td><td>CTS, Clear to send</td></tr>
<tr><td>9</td><td>RI, Ring indicator</td></tr>
<tr><td>10</td><td>No Connect.</td></tr>
</table></td></tr>
</table>

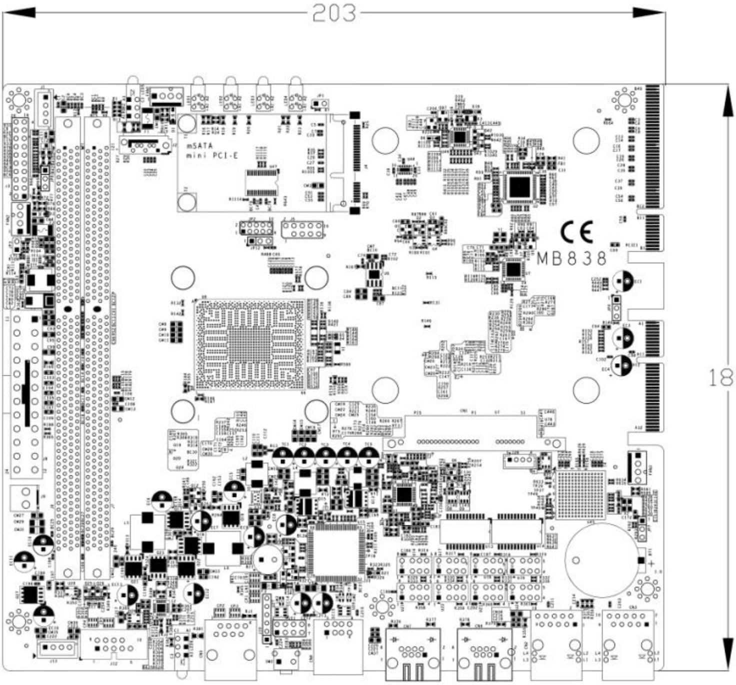
<table>
<tr><td><b>Fan Connector</b></td><td>Three 4-pin smart fan connectors</td></tr>
<tr><td><b>Smart Fan Control</b></td><td>The active temperature may be adjusted based on system thermal test result
<table>
<tr><th></th><th>Active Temperature</th><th>Tolerance</th><th>Default Smart Fan Enable or Disable</th></tr>
<tr><td>CPU Fan</td><td>50</td><td>+/- 3</td><td>Enable</td></tr>
<tr><td>System Fan</td><td>50</td><td>+/- 3</td><td>Enable</td></tr>
</table></td></tr>
<tr><td><b>RTC</b></td><td>Rangeley built-in RTC with on-board lithium battery & holder</td></tr>
<tr><td><b>Expansion Slot</b>
<b>(Golden Finger)</b></td><td>• #1: PCI-e x8, for IP332
• #2: PCI-e x4, for IP334</td></tr>
<tr><td><b>Front Panel</b>
<b>Function LED</b></td><td>#1 LED: Power
(Green = Power On, Off= No Power)
#2 LED: Bypass or Status 2 (Pin header control)
Bypass: Green = LAN 1-2 or 3-4 Bypass,
Off = LAN Normal
#3 LED: Status 1 (GPIO control, Yellow / Red)

</td></tr>
<tr><td><b>LCM</b></td><td>Optional, iIO 2x16 characters LCM (COM2)</td></tr>
<tr><td><b>Front Panel Buttons & Connector</b></td><td>• Four RJ-45 connectors for Eth1~4 with LEDs
• USB 2.0 x2
• RJ-45 (for console, COM1)
• Three LEDs for Power, Bypass & Status
• Factory Mode Restore Reset Switch</td></tr>
<tr><td><b>Rear I/O interface</b></td><td>• PSU AC inlet
• 1x Slot Opening</td></tr>
<tr><td><b>Jumper /</b>
<b>Pin Header /</b>
<b>Switch</b></td><td>• AT or ATX mode selection jumper
• ATX mode power on / off pin header
• HDD active LED pin header
• System Reset pin header
• Clear CMOS</td></tr>
<tr><td><b>Power Connector</b></td><td>• 24-pin ATX standard (connected to system power supply)
• 2-pin 12V DC-In (connected to external power adapter)</td></tr>
<tr><td><b>TPM</b></td><td>TPM 1.2 (INFINEON SLB9655TT1.2)</td></tr>
<tr><td><b>Watchdog Timer</b></td><td>Yes (256 segments, 0, 1, 2…255 sec/min)</td></tr>
<tr><td><b>Operating Temperature</b></td><td><b>0°C ~ 60°C</b></td></tr>
<tr><td><b>Storage Temperature</b></td><td><b>-20°C ~ 80°C</b></td></tr>
<tr><td><b>Operational Humidity</b></td><td><b>10% ~ 90% Relative Humidity (non-condensing)</b></td></tr>
<tr><td><b>RoHS Compliant</b></td><td><b>Yes</b></td></tr>
<tr><td><b>Dimensions</b></td><td>203 mm x 180 mm</td></tr>
</table>

Checklist

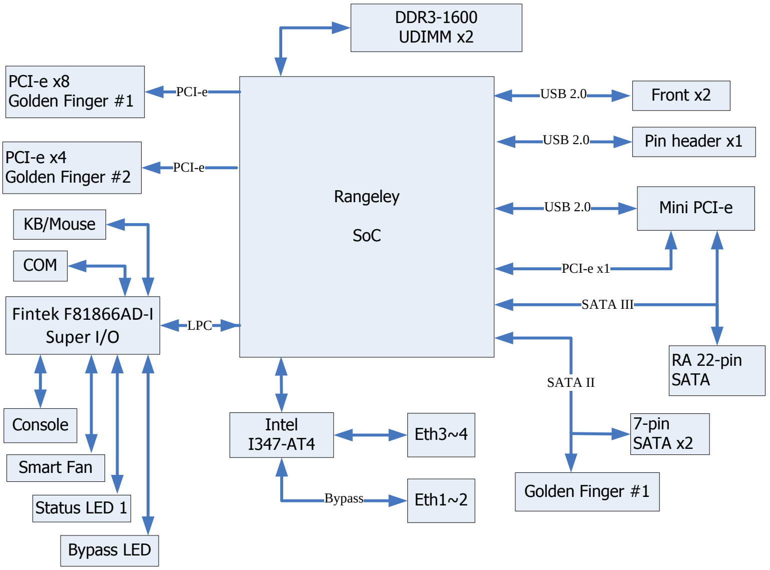
Your MB838 package should include the items listed below.

- MB838 motherboard
- Driver DVD

Board Dimensions



Block Diagram



Installations

This section provides information on how to use the jumpers and connectors on the MB838 in order to set up a workable system. The topics covered are:

Installing the Memory.. 12
Setting the Jumpers.. 13
Jumper Locations on MB838.. 14
Connectors on MB838.. 16

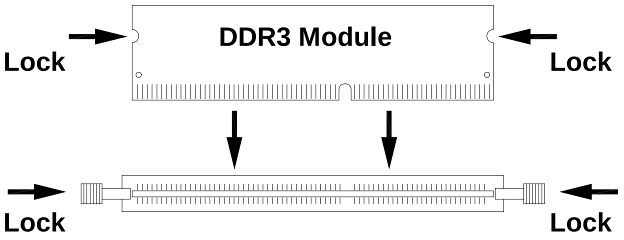
Installing the Memory

The MB838 board supports four DDR3 memory socket for a maximum total memory of 32GB in DDR3 DIMM memory type.

Installing and Removing Memory Modules

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

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

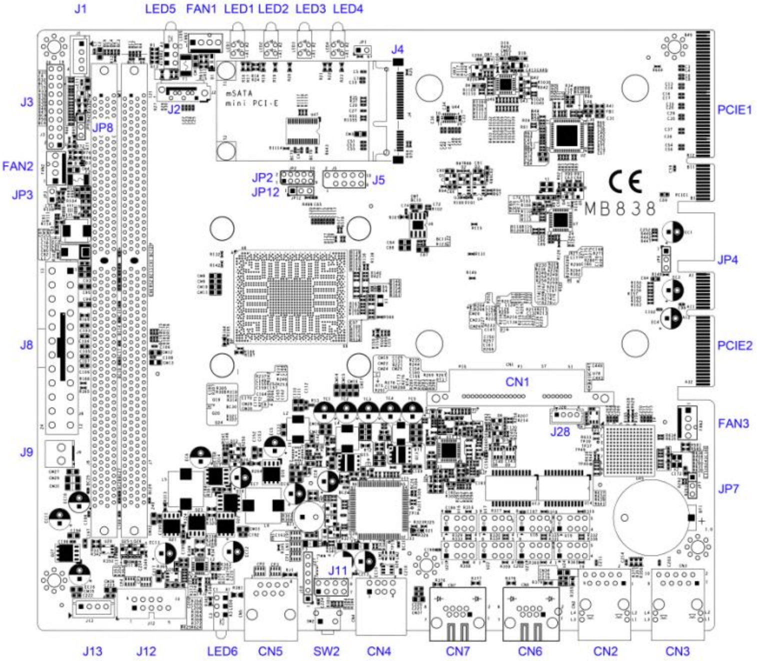


Setting the Jumpers

Jumpers are used on MB838 to select various settings and features according to your needs and applications. Contact your supplier if you have doubts about the best configuration for your needs. The following lists the connectors on MB838 and their respective functions.

JP2: Function reserved.. 14
JP3: DDR power selection.. 15
JP4: AT / ATX mode selection... 15
JP7: Clear CMOS 15
JP8: LED5, LED6 Bypass or HDD LED selection............................. 15
JP12: Function reserved.. 15

Jumper Locations on MB838



JP2: Function reserved

JP3: DDR power selection

JP3	DDR3 power
Short (Default)	1.5V
Open	1.35V

JP4: AT / ATX mode selection

JP4	AT/ATX
1-2 Short	ATX
2-3 Short (Default)	AT

JP7: Clear CMOS

JP7	CMOS
1-2 Short (Default)	Normal
2-3 Short	Clear CMOS

JP8: LED5, LED6 Bypass or HDD LED selection

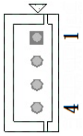
JP8	CMOS
1-2 Short	HDD
2-3 Short (Default)	LAN Bypass

JP12: Function reserved

Connectors on MB838

J1, J13: Power Connector, Pitch 2.54mm 17
J2: 7-pin SATA connector 17
J3: System Function Connector 17
J4: Mini PCI-e (supports mSATA) 18
J5: Function reserved 18
J8: Power connector for ATX power supply 18
J9: 2-pin Power connector for 12V DC power 18
J11: USB pin header 18
J12: COM2 pin header 18
J28: Power Connector, Pitch 2.0mm 19
CN1: 22-pin right angle SATA connector 19
CN2, CN3: RJ-45 LAN port 3 and 4 19
CN4: USB 2.0 connectors 19
CN5: COM1 (Console Port) 19
CN6, CN7: RJ-45 LAN port 1 and 2 19
FAN1, FAN2, FAN3: System Fan Power Connector 19
LED1: LAN port 1 LED 19
LED2: LAN port 2 LED 19
LED3: LAN port 3 LED 19
LED4: LAN port 4 LED 19
LED5, 6: Status LED 20
SW2: Push button (GPI, porting by software) 20

J1, J13: Power Connector, Pitch 2.54mm

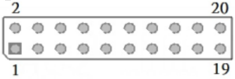


Pin #	Signal Name
1	+5V
2	GND
3	GND
4	+12V

J2: 7-pin SATA connector

J3: System Function Connector

J3 provides connectors for system indicators that provide light indication of the computer activities and switches to change the computer status. J13 is a 20-pin header that provides interfaces for the following functions



Pin 1, 2: Speaker

This connector provides an interface to a speaker for audio tone generation. An 8-ohm speaker is recommended.

Pin #	Signal Name
1	+5V
2	GND

Pin 13, 14: ATX Power ON Switch

This 2-pin connector is an “ATX Power Supply On/Off Switch” on the system that connects to the power switch on the case. When pressed, the power switch will force the system to power on. When pressed again, it will force the system to power off.

Pin #	Signal Name
13	GND
14	Power_ON

Pin 15, 16: Power LED

The power LED indicates the status of the main power switch.

Pin #	Signal Name
15	+5V
16	GND

Pin 17, 18: Reset Switch

The reset switch allows the user to reset the system without turning the main power switch off and then on again. Orientation is not required when making a connection to this header.

Pin #	Signal Name
17	GND
18	PM_SYSRST#

Pin 19, 20: HDD LED:

Pin #	Signal Name
19	+5V
20	GND

J4: Mini PCI-e (supports mSATA)

J5: Function reserved

J8: Power connector for ATX power supply

J9: 2-pin Power connector for 12V DC power

Pin #	Signal Name
1	+12V
2	GND

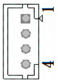
J11: USB pin header

Signal Name	Pin #	Pin #	Signal Name
5V	1	2	NC
D-	3	4	NC
D+	5	6	NC
GND	7	8	NC

J12: COM2 pin header

Signal Name	Pin #	Pin #	Signal Name
DCD	1	6	DSR
RXD	2	7	RTS
TXD	3	8	CTS
DTR	4	9	RI
GND	5		

J28: Power Connector, Pitch 2.0mm



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

CN1: 22-pin right angle SATA connector

CN2, CN3: RJ-45 LAN port 3 and 4

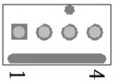
CN4: USB 2.0 connectors

CN5: COM1 (Console Port)

CN6, CN7: RJ-45 LAN port 1 and 2

FAN1, FAN2, FAN3: System Fan Connectors

FAN1, FAN2, FAN3 is a 4-pin header for system fans.
The fan must be 12V (Max. 1A).



Pin #	Signal Name
1	Ground
2	+12V
3	Rotation detection
4	Rotation control

LED1: LAN port 1 LED

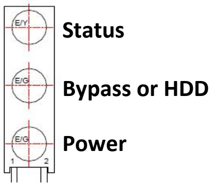
LED2: LAN port 2 LED

LED3: LAN port 3 LED

LED4: LAN port 4 LED

LED5, 6: Status LED

A1 & C1 : Status LED
A2 & C2 : Bypass or HDD status LED
A3 & C3 : Power LED



Signal Name	Pin #	Pin #	Signal Name
SIO_GPIO 33	A1	C1	SIO_GPIO 32
+5 V	A2	C2	JP15 Selection
+3.3 V	A3	C3	GND

SW2: Push button (GPI, porting by software)

Digital I/O Sample Configuration

Filename：Main.cpp

```
//---------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//---------------------------------------------------------------------------
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include "F81865.H"

#define BIT0 0x01
#define BIT1 0x02
#define BIT2 0x04
#define BIT3 0x08
#define BIT4 0x10
#define BIT5 0x20
#define BIT6 0x40
#define BIT7 0x80

//---------------------------------------------------------------------------
int main (void);

void Dio5Initial(void);
void Dio5SetOutput(unsigned char);
unsigned char Dio5GetInput(void);
void Dio5SetDirection(unsigned char);
unsigned char Dio5GetDirection(void);
//---------------------------------------------------------------------------
int main (void)
{
        char SIO;
        unsigned char DIO;

        printf("Fintek 81865/81866 digital I/O test program\n");

        SIO = Init_F81865();
        if (SIO == 0)
        {
                printf("Can not detect Fintek 81865/81866, program abort.\n");
                return(1);
        }//if (SIO == 0)

        Dio5Initial();
/*
        //for GPIO50..57
        Dio5SetDirection(0xF0);        //GP50..53 = input, GP54..57=output
        printf("Current DIO direction = 0x%X\n", Dio5GetDirection());

        printf("Current DIO status = 0x%X\n", Dio5GetInput());

        printf("Set DIO output to high\n");
        Dio5SetOutput(0x0F);

        printf("Set DIO output to low\n");
        Dio5SetOutput(0x00);
*/

        //for GPIO50..57
        Dio5SetDirection(0xF0);        //GP50..53 = input, GP54..57=output

        Dio5SetOutput(0x00);                            //clear
//      DIO = Dio5GetInput() & 0x0F;
```

```
        Dio5SetOutput(0x00);                    //clear
        DIO = Dio5GetInput() & 0x0F;
        if (DIO != 0x0A)
        {
                printf("The Fintek 81865 digital IO abnormal, abort.\n");
                return(1);
        }//if (DIO != 0x0A)

        Dio5SetOutput(0xA0);                    //clr# is high
        Dio5SetOutput(0xF0);                    //clk and clr# is high
        Dio5SetOutput(0xA0);                    //clr# is high

        DIO = Dio5GetInput() & 0x0F;
        if (DIO != 0x05)
        {
                printf("The Fintek 81865 digital IO abnormal, abort.\n");
                return(1);
        }
        printf("!!! Pass !!!\n");
        return 0;
}
//---------------------------------------------------------------------------
void Dio5Initial(void)
{
        unsigned char ucBuf;

        //switch GPIO multi-function pin for  gpio 50~57

//gpio53~57 UR5_FULL_EN(bit1), clear UR6_FULL_EN(bit3)
        //set UR5_FULL_EN,should set UR_GP_PROG_EN = 1 (reg26,bit0) first

        ucBuf = Get_F81865_Reg(0x26);
        ucBuf |= BIT0;
        Set_F81865_Reg(0x26, ucBuf);
        //set UR5_FULL_EN(bit1), clear UR6_FULL_EN(bit3)
        ucBuf = Get_F81865_Reg(0x2A);
        ucBuf &= ~BIT3;//clear bit 3,
        ucBuf |= BIT1;//set bit 1,
        Set_F81865_Reg(0x2a, ucBuf);

//GPIO51 ~ GPIO52
        //clear UR6_ALT_EN(bit5),  IR_ALT_EN(bit4),set FDC_GP_EN(bit3)

//GPIO50
//set FDC_GP_EN(bit3), clear  RTS6_ALT_EN(RTS6_2_ALT_EN)(bit6)

        ucBuf = Get_F81865_Reg(0x2A);
        ucBuf &= ~(BIT4+BIT5+BIT6);      //clear UR6_ALT_EN(bit5),  IR_ALT_EN(bit4),
RTS6_ALT_EN(RTS6_2_ALT_EN)(bit6)
        Set_F81865_Reg(0x2a, ucBuf);
        //set FDC_GP_EN(bit3), should clear UR_GP_PROG_EN (reg26,bit0) first
        ucBuf = Get_F81865_Reg(0x26);
        ucBuf &= ~BIT0;
        Set_F81865_Reg(0x26, ucBuf);//clear UR_GP_PROG_EN = 0 (reg26,bit0)

        ucBuf = Get_F81865_Reg(0x2A);
        ucBuf |= BIT3;          //set FDC_GP_EN(bit3),
        Set_F81865_Reg(0x2a, ucBuf);


        Set_F81865_LD(0x06);                                            //switch to logic device 6

        //enable the GP5 group
        ucBuf = Get_F81865_Reg(0x30);
        ucBuf |= 0x01;
        Set_F81865_Reg(0x30, ucBuf);

        Set_F81865_Reg(0xA0, 0x00);                                     //define as input mode
        Set_F81865_Reg(0xA3, 0xFF);                                     //push pull mode
}
//---------------------------------------------------------------------------
```

```
void Dio5SetOutput(unsigned char NewData)
{
        Set_F81865_LD(0x06);                                            //switch to logic device 6
        Set_F81865_Reg(0xA1, NewData);
}
//-------------------------------------------------------------------------
unsigned char Dio5GetInput(void)
{
        unsigned char result;

        Set_F81865_LD(0x06);                                            //switch to logic device 6
        result = Get_F81865_Reg(0xA2);
        return (result);
}
//-------------------------------------------------------------------------
void Dio5SetDirection(unsigned char NewData)
{
        //NewData : 1 for input, 0 for output
        Set_F81865_LD(0x06);                                            //switch to logic device 6
        Set_F81865_Reg(0xA0, NewData);
}
//-------------------------------------------------------------------------
unsigned char Dio5GetDirection(void)
{
        unsigned char result;

        Set_F81865_LD(0x06);                                            //switch to logic device 6
        result = Get_F81865_Reg(0xA0);
        return (result);
}
//-------------------------------------------------------------------------
```

Filename：81865.cpp

```
//-------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//-------------------------------------------------------------------------
#include "F81865.H"
#include <dos.h>
//-------------------------------------------------------------------------
unsigned int F81865_BASE;
void Unlock_F81865 (void);
void Lock_F81865 (void);
//-------------------------------------------------------------------------
unsigned int Init_F81865(void)
{
        unsigned int result;
        unsigned char ucDid;

        F81865_BASE = 0x4E;
        result = F81865_BASE;

        ucDid = Get_F81865_Reg(0x20);
        if (ucDid == 0x07||ucDid == 0x10)                               //Fintek 81865/66
        {       goto Init_Finish;       }

        F81865_BASE = 0x2E;
        result = F81865_BASE;

        ucDid = Get_F81865_Reg(0x20);
        if (ucDid == 0x07||ucDid == 0x10)                               //Fintek 81865/66
        {       goto Init_Finish;       }

        F81865_BASE = 0x00;
        result = F81865_BASE;
Init_Finish:
        return (result);
}
//---------------------------------------------------------------------------
void Unlock_F81865 (void)
{
        outportb(F81865_INDEX_PORT, F81865_UNLOCK);
        outportb(F81865_INDEX_PORT, F81865_UNLOCK);
}
//---------------------------------------------------------------------------
void Lock_F81865 (void)
{
        outportb(F81865_INDEX_PORT, F81865_LOCK);
}
//---------------------------------------------------------------------------
void Set_F81865_LD( unsigned char LD)
{
        Unlock_F81865();
        outportb(F81865_INDEX_PORT, F81865_REG_LD);
        outportb(F81865_DATA_PORT, LD);
        Lock_F81865();
}
//---------------------------------------------------------------------------
void Set_F81865_Reg( unsigned char REG, unsigned char DATA)
{
        Unlock_F81865();
        outportb(F81865_INDEX_PORT, REG);
        outportb(F81865_DATA_PORT, DATA);
        Lock_F81865();
}
//---------------------------------------------------------------------------
unsigned char Get_F81865_Reg(unsigned char REG)
{
        unsigned char Result;
        Unlock_F81865();
        outportb(F81865_INDEX_PORT, REG);
        Result = inportb(F81865_DATA_PORT);
        Lock_F81865();
        return Result;
}
//---------------------------------------------------------------------------
```

Filename：81865.h

```
//---------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//---------------------------------------------------------------------------
#ifndef __F81865_H
#define __F81865_H                      1
//---------------------------------------------------------------------------
#defineF81865_INDEX_PORT                (F81865_BASE)
#defineF81865_DATA_PORT                 (F81865_BASE+1)
//---------------------------------------------------------------------------
#defineF81865_REG_LD                    0x07
//---------------------------------------------------------------------------
#define F81865_UNLOCK                   0x87
#defineF81865_LOCK                              0xAA
//---------------------------------------------------------------------------
unsigned int Init_F81865(void);
void Set_F81865_LD( unsigned char);
void Set_F81865_Reg( unsigned char, unsigned char);
unsigned char Get_F81865_Reg( unsigned char);
//---------------------------------------------------------------------------
#endif  //__F81865_H
```

Watchdog Timer Configuration

The WDT is used to generate a variety of output signals after a user programmable count. The WDT is suitable for use in the prevention of system lock-up, such as when software becomes trapped in a deadlock. Under these sorts of circumstances, the timer will count to zero and the selected outputs will be driven. Under normal circumstance, the user will restart the WDT at regular intervals before the timer counts to zero.

SAMPLE CODE:
This code and information is provided "as is" without warranty of any kind, either expressed or implied, including but not limited to the implied warranties of merchantability and/or fitness for a particular purpose.

```
//---------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//---------------------------------------------------------------------------
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include "F81866.H"
//---------------------------------------------------------------------------
int main (int argc, char *argv[]);
void EnableWDT(int);
void DisableWDT(void);
//---------------------------------------------------------------------------
int main (int argc, char *argv[])
{
        unsigned char bBuf;
        unsigned char bTime;
        char **endptr;

        char SIO;

        printf("Fintek 81866 watch dog program\n");

        SIO = Init_F81866();
        if (SIO == 0)
        {
                printf("Can not detect Fintek 81866, program abort.\n");
                return(1);
        }//if (SIO == 0)

        if (argc != 2)
        {
                printf(" Parameter incorrect!!\n");
                return (1);
        }

        bTime = strtol (argv[1], endptr, 10);
        printf("System will reset after %d seconds\n", bTime);

        if (bTime)
        {       EnableWDT(bTime);  }
        else
        {       DisableWDT();       }
        return 0;
}

//---------------------------------------------------------------------------
void EnableWDT(int interval)
{
        unsigned char bBuf;

        bBuf = Get_F81866_Reg(0x2B);
        bBuf &= (~0x20);
        Set_F81866_Reg(0x2B, bBuf);                     //Enable WDTO

        Set_F81866_LD(0x07);                            //switch to logic device 7
        Set_F81866_Reg(0x30, 0x01);                     //enable timer

        bBuf = Get_F81866_Reg(0xF5);
        bBuf &= (~0x0F);
        bBuf |= 0x52;
        Set_F81866_Reg(0xF5, bBuf);                     //count mode is second

        Set_F81866_Reg(0xF6, interval);             //set timer

        bBuf = Get_F81866_Reg(0xFA);
        bBuf |= 0x01;
        Set_F81866_Reg(0xFA, bBuf);                     //enable WDTO output

        bBuf = Get_F81866_Reg(0xF5);
        bBuf |= 0x20;
        Set_F81866_Reg(0xF5, bBuf);                     //start counting
}
//---------------------------------------------------------------------------
void DisableWDT(void)
{
        unsigned char bBuf;

        Set_F81866_LD(0x07);                            //switch to logic device 7

        bBuf = Get_F81866_Reg(0xFA);
        bBuf &= ~0x01;
        Set_F81866_Reg(0xFA, bBuf);                     //disable WDTO output

        bBuf = Get_F81866_Reg(0xF5);
        bBuf &= ~0x20;
        bBuf |= 0x40;
        Set_F81866_Reg(0xF5, bBuf);                     //disable WDT
}
//---------------------------------------------------------------------------
//---------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//---------------------------------------------------------------------------
#include "F81866.H"
#include <dos.h>
//---------------------------------------------------------------------------
unsigned int F81866_BASE;
void Unlock_F81866 (void);
void Lock_F81866 (void);
//---------------------------------------------------------------------------
unsigned int Init_F81866(void)
{
        unsigned int result;
        unsigned char ucDid;

        F81866_BASE = 0x4E;
        result = F81866_BASE;
        ucDid = Get_F81866_Reg(0x20);
        if (ucDid == 0x07)                                  //Fintek 81866
        {       goto Init_Finish;       }

        F81866_BASE = 0x2E;
        result = F81866_BASE;
        ucDid = Get_F81866_Reg(0x20);
        if (ucDid == 0x07)                                  //Fintek 81866
        {       goto Init_Finish;       }

        F81866_BASE = 0x00;
        result = F81866_BASE;

Init_Finish:
        return (result);
}
//---------------------------------------------------------------------------
void Unlock_F81866 (void)
{
        outportb(F81866_INDEX_PORT, F81866_UNLOCK);
        outportb(F81866_INDEX_PORT, F81866_UNLOCK);
}
//---------------------------------------------------------------------------
void Lock_F81866 (void)
{
        outportb(F81866_INDEX_PORT, F81866_LOCK);
}
//---------------------------------------------------------------------------
void Set_F81866_LD( unsigned char LD)
{
        Unlock_F81866();
        outportb(F81866_INDEX_PORT, F81866_REG_LD);
        outportb(F81866_DATA_PORT, LD);
        Lock_F81866();
}
//---------------------------------------------------------------------------
void Set_F81866_Reg( unsigned char REG, unsigned char DATA)
{
        Unlock_F81866();
        outportb(F81866_INDEX_PORT, REG);
        outportb(F81866_DATA_PORT, DATA);
        Lock_F81866();
}
//---------------------------------------------------------------------------
unsigned char Get_F81866_Reg(unsigned char REG)
{
        unsigned char Result;
        Unlock_F81866();
        outportb(F81866_INDEX_PORT, REG);
        Result = inportb(F81866_DATA_PORT);
        Lock_F81866();
        return Result;
}
//---------------------------------------------------------------------------
//---------------------------------------------------------------------------
//
// THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
// KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
// IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
// PURPOSE.
//
//---------------------------------------------------------------------------
#ifndef __F81866_H
#define __F81866_H                          1
//---------------------------------------------------------------------------
#define F81866_INDEX_PORT                   (F81866_BASE)
#define F81866_DATA_PORT                    (F81866_BASE+1)
//---------------------------------------------------------------------------
#define F81866_REG_LD                       0x07
//---------------------------------------------------------------------------
```

```
#define F81866_UNLOCK                    0x87
#defineF81866_LOCK                                0xAA
//---------------------------------------------------------------------------
unsigned int Init_F81866(void);
void Set_F81866_LD( unsigned char);
void Set_F81866_Reg( unsigned char, unsigned char);
unsigned char Get_F81866_Reg( unsigned char);
//---------------------------------------------------------------------------
#endif  //__F81866_H
```

BIOS Setup

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

BIOS Introduction 30
BIOS Setup.. 30
Advanced Settings 31
IntelRCSetup Settings.. 38
Security Settings 41
Boot Settings.. 42
Save & Exit Settings... 43

BIOS Introduction

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

BIOS Setup

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

```
        Press <DEL> / <F2> to Enter Setup
```

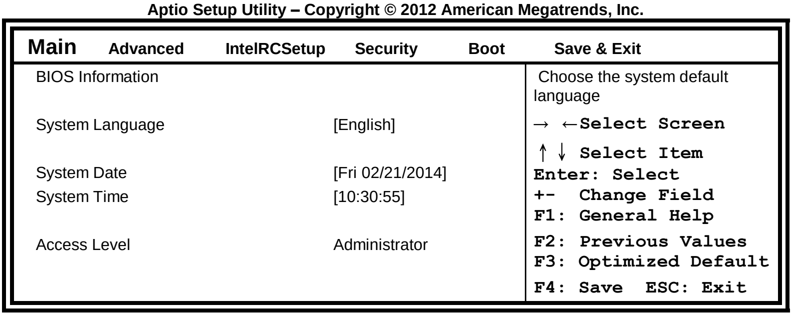
In general, you press the arrow keys to highlight items, <Enter> to select, the <PgUp> and <PgDn> keys to change entries, <F1> for help and <Esc> to quit.

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

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

Main Settings

Aptio Setup Utility – Copyright © 2012 American Megatrends, Inc.



```
Main    Advanced    IntelRCSetup    Security    Boot    Save & Exit

BIOS Information                                  Choose the system default
                                                  language
System Language            [English]              → ←Select Screen
                                                  ↑ ↓ Select Item
System Date                [Fri 02/21/2014]       Enter: Select
System Time                [10:30:55]             +-  Change Field
                                                  F1: General Help
Access Level               Administrator          F2: Previous Values
                                                  F3: Optimized Default
                                                  F4: Save  ESC: Exit
```

System Language

Choose the system default language.

System Date

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

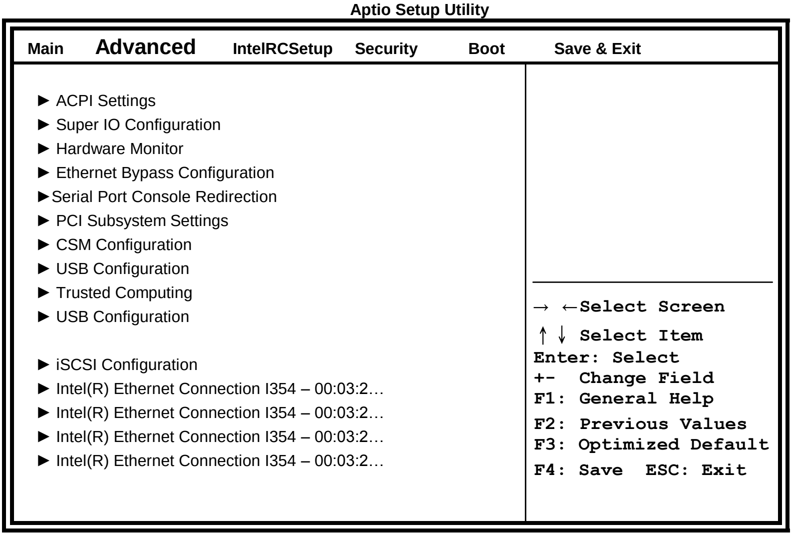
System Time

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

Advanced Settings

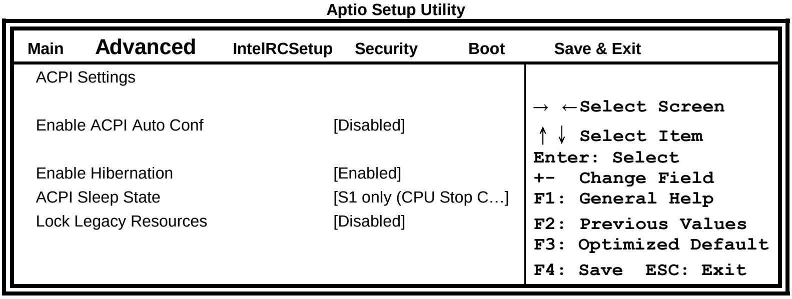
This section allows you to configure and improve your system and allows you to set up some system features according to your preference.

Aptio Setup Utility



```
Main    Advanced    IntelRCSetup    Security    Boot    Save & Exit

 ► ACPI Settings
 ► Super IO Configuration
 ► Hardware Monitor
 ► Ethernet Bypass Configuration
 ►Serial Port Console Redirection
 ► PCI Subsystem Settings
 ► CSM Configuration
 ► USB Configuration
 ► Trusted Computing
 ► USB Configuration                              → ←Select Screen
                                                  ↑ ↓ Select Item
 ► iSCSI Configuration                            Enter: Select
 ► Intel(R) Ethernet Connection I354 – 00:03:2…   +-  Change Field
 ► Intel(R) Ethernet Connection I354 – 00:03:2…   F1: General Help
 ► Intel(R) Ethernet Connection I354 – 00:03:2…   F2: Previous Values
 ► Intel(R) Ethernet Connection I354 – 00:03:2…   F3: Optimized Default
                                                  F4: Save  ESC: Exit
```

ACPI Settings



```
                         Aptio Setup Utility
Main   Advanced   IntelRCSetup   Security   Boot   Save & Exit
  ACPI Settings
                                                  → ←Select Screen
  Enable ACPI Auto Conf        [Disabled]         ↑ ↓ Select Item
                                                  Enter: Select
  Enable Hibernation           [Enabled]          +-  Change Field
  ACPI Sleep State             [S1 only (CPU Stop C…]  F1: General Help
  Lock Legacy Resources        [Disabled]         F2: Previous Values
                                                  F3: Optimized Default
                                                  F4: Save  ESC: Exit
```

Enable Hibernation

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

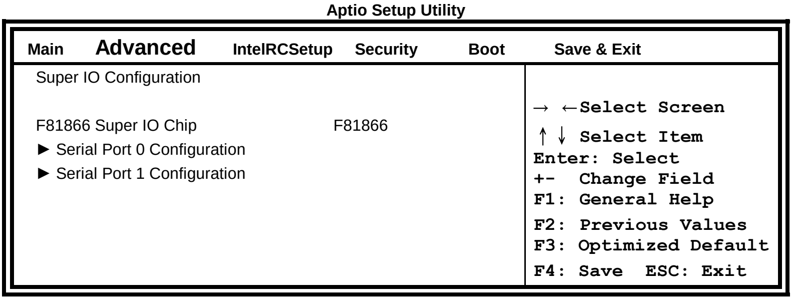
ACPI Sleep State

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

Lock Legacy Resources

Enabled or Disabled Lock of Legacy Resources.

Super IO Configuration

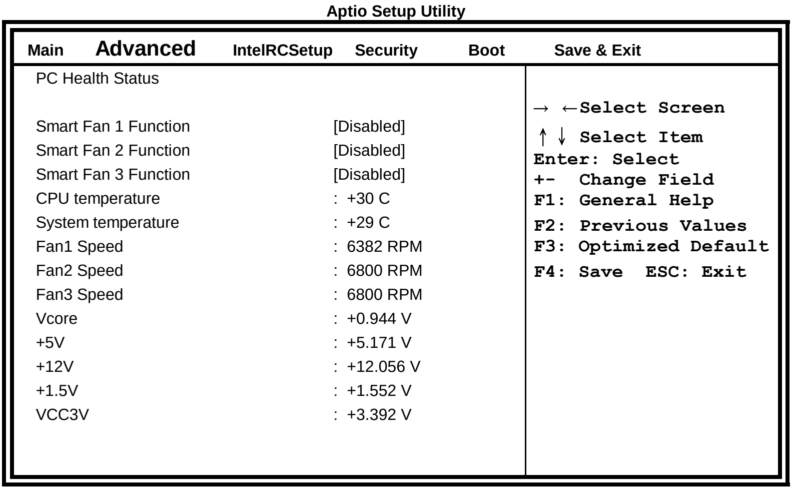


```
                         Aptio Setup Utility
Main   Advanced   IntelRCSetup   Security   Boot   Save & Exit
  Super IO Configuration
                                                  → ←Select Screen
  F81866 Super IO Chip         F81866             ↑ ↓ Select Item
  ► Serial Port 0 Configuration                   Enter: Select
  ► Serial Port 1 Configuration                   +-  Change Field
                                                  F1: General Help
                                                  F2: Previous Values
                                                  F3: Optimized Default
                                                  F4: Save  ESC: Exit
```

Serial Port Configuration

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

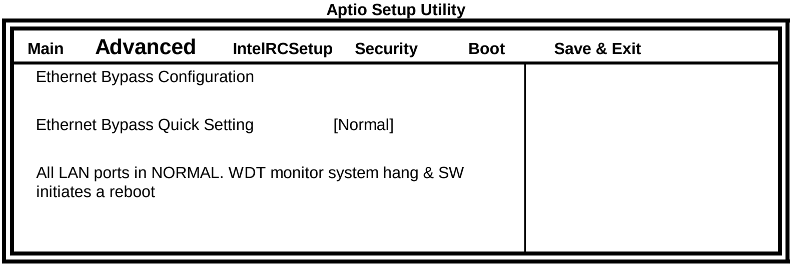
Hardware Monitor



```
                              Aptio Setup Utility
 Main   Advanced   IntelRCSetup   Security   Boot   Save & Exit
 PC Health Status
                                                  → ←Select Screen
 Smart Fan 1 Function            [Disabled]      ↑ ↓ Select Item
 Smart Fan 2 Function            [Disabled]      Enter: Select
 Smart Fan 3 Function            [Disabled]      +-  Change Field
 CPU temperature                : +30 C          F1: General Help
 System temperature             : +29 C          F2: Previous Values
 Fan1 Speed                     : 6382 RPM       F3: Optimized Default
 Fan2 Speed                     : 6800 RPM       F4: Save  ESC: Exit
 Fan3 Speed                     : 6800 RPM
 Vcore                          : +0.944 V
 +5V                            : +5.171 V
 +12V                           : +12.056 V
 +1.5V                          : +1.552 V
 VCC3V                          : +3.392 V
```

Ethernet Bypass Configuration

Aptio Setup Utility



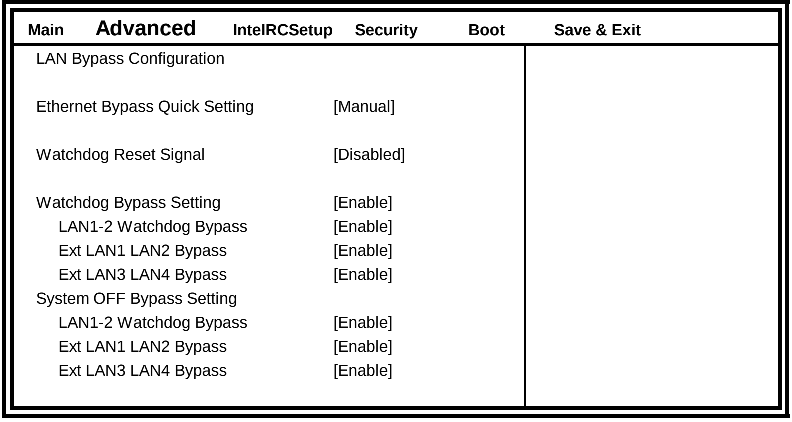
```
Ethernet Bypass Configuration

Ethernet Bypass Quick Setting              [Normal]

All LAN ports in NORMAL. WDT monitor system hang & SW
initiates a reboot
```

Bypass Quick Setting

Set LAN Bypass to Normal, Bypass, Firewall or Custom Define Mode

Normal mode: All LAN ports in NORMAL. When Watchdog monitor system hangs, software will initiates a system reboot.
Bypass mode: All LAN ports in BYPASS during power-off or watchdog initiates Bypass. System will not reboot.
Firewall mode: All LAN ports in BYPASS until software change it to NORMAL under OS. When watchdog monitors system hang, software will initiates a system reboot.
Manual mode: Customer defines watchdog reset, watchdog Bypass and power-off Bypass settings.

Aptio Setup Utility

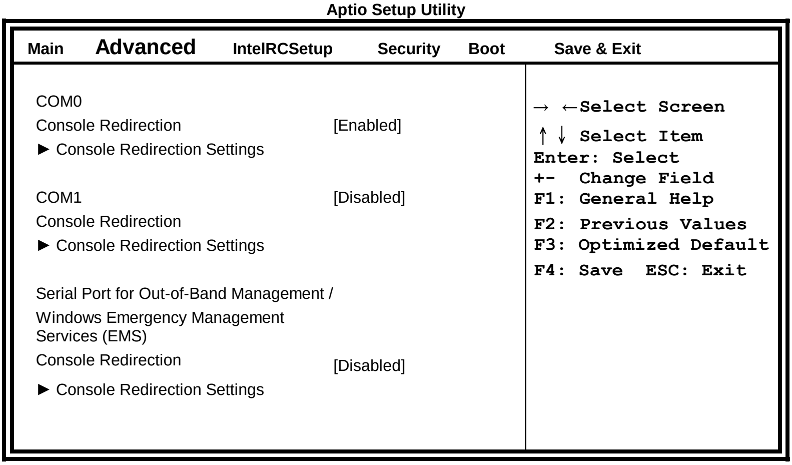


```
LAN Bypass Configuration

Ethernet Bypass Quick Setting              [Manual]

Watchdog Reset Signal                      [Disabled]

Watchdog Bypass Setting                    [Enable]
   LAN1-2 Watchdog Bypass                  [Enable]
   Ext LAN1 LAN2 Bypass                    [Enable]
   Ext LAN3 LAN4 Bypass                    [Enable]
System OFF Bypass Setting
   LAN1-2 Watchdog Bypass                  [Enable]
   Ext LAN1 LAN2 Bypass                    [Enable]
   Ext LAN3 LAN4 Bypass                    [Enable]
```

Note: “Ext LAN Bypass” items only appear when extended IBASE LAN module card installed.

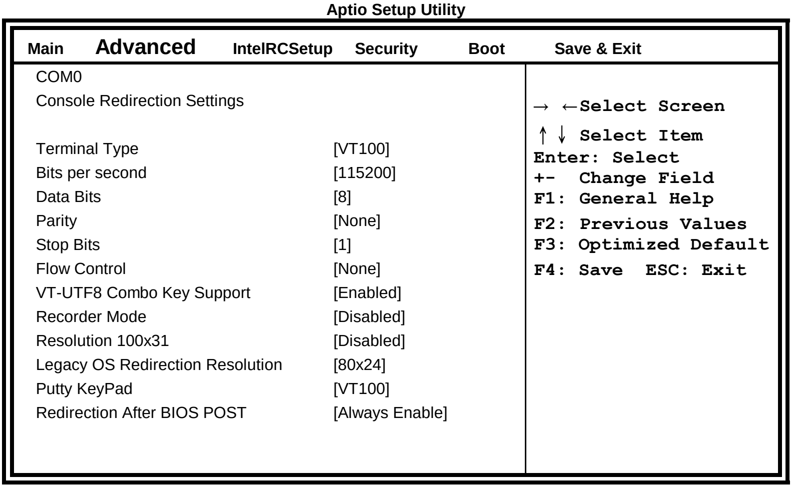
Serial Port Console Redirection

Aptio Setup Utility



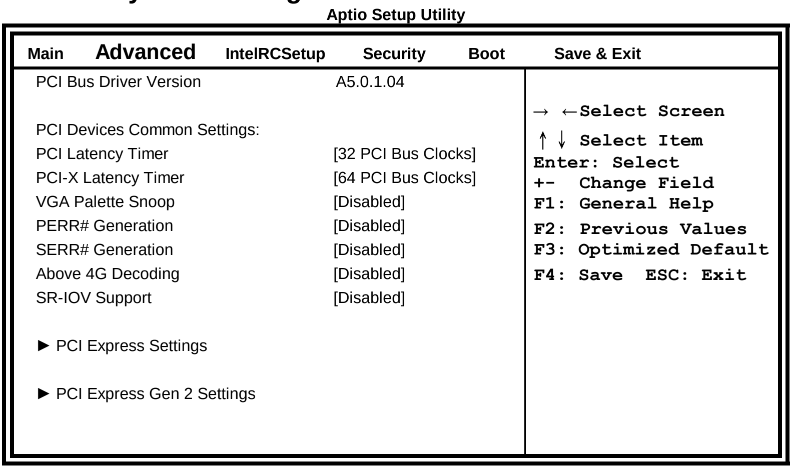
```
Main   Advanced   IntelRCSetup   Security   Boot   Save & Exit

COM0                                            → ←Select Screen
Console Redirection            [Enabled]        ↑ ↓ Select Item
► Console Redirection Settings                  Enter: Select
                                                +-  Change Field
COM1                           [Disabled]       F1: General Help
Console Redirection                             F2: Previous Values
► Console Redirection Settings                  F3: Optimized Default
                                                F4: Save  ESC: Exit
Serial Port for Out-of-Band Management /
Windows Emergency Management
Services (EMS)
Console Redirection            [Disabled]
► Console Redirection Settings
```

Aptio Setup Utility



```
Main   Advanced   IntelRCSetup   Security   Boot   Save & Exit
COM0
Console Redirection Settings                    → ←Select Screen
                                                ↑ ↓ Select Item
Terminal Type                  [VT100]          Enter: Select
Bits per second                [115200]         +-  Change Field
Data Bits                      [8]              F1: General Help
Parity                         [None]           F2: Previous Values
Stop Bits                      [1]              F3: Optimized Default
Flow Control                   [None]           F4: Save  ESC: Exit
VT-UTF8 Combo Key Support      [Enabled]
Recorder Mode                  [Disabled]
Resolution 100x31              [Disabled]
Legacy OS Redirection Resolution [80x24]
Putty KeyPad                   [VT100]
Redirection After BIOS POST    [Always Enable]
```

PCI Subsystem Settings



```
                                   Aptio Setup Utility
 Main   Advanced   IntelRCSetup    Security    Boot     Save & Exit
   PCI Bus Driver Version                A5.0.1.04
                                                          → ←Select Screen
   PCI Devices Common Settings:                           ↑ ↓ Select Item
   PCI Latency Timer                     [32 PCI Bus Clocks]  Enter: Select
   PCI-X Latency Timer                   [64 PCI Bus Clocks]  +-  Change Field
   VGA Palette Snoop                     [Disabled]       F1: General Help
   PERR# Generation                      [Disabled]       F2: Previous Values
   SERR# Generation                      [Disabled]       F3: Optimized Default
   Above 4G Decoding                     [Disabled]       F4: Save  ESC: Exit
   SR-IOV Support                        [Disabled]

   ► PCI Express Settings

   ► PCI Express Gen 2 Settings
```

PCI Latency Timer

Value to be programmed into PCI Latency Timer Register.

VGA Palette Snoop

Enables or disables VGA Palette Registers Snooping.

PERR# Generation

Enables or disables PCI device to generate PERR#.

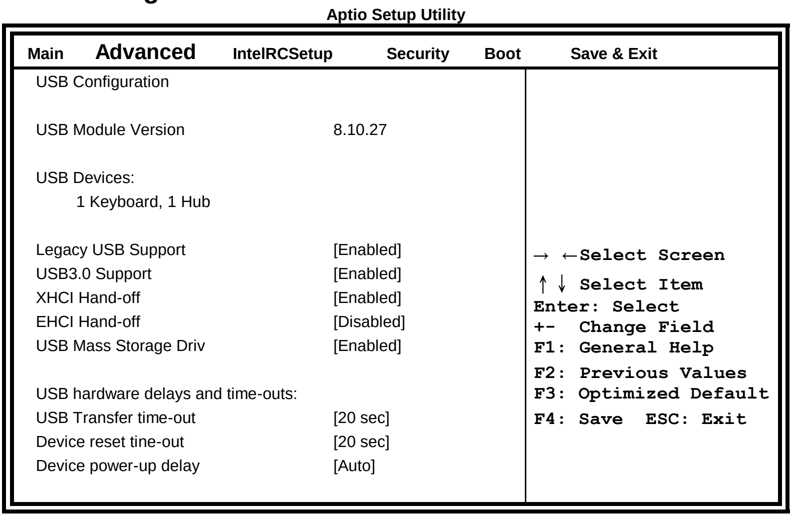
SERR# Generation

Enables or disables PCI device to generate SERR#.

PCI Express Settings

Change PCI Express devices settings.

USB Configuration



```
                         Aptio Setup Utility
Main   Advanced   IntelRCSetup    Security   Boot     Save & Exit

USB Configuration

USB Module Version                     8.10.27

USB Devices:
     1 Keyboard, 1 Hub

Legacy USB Support                     [Enabled]        → ←Select Screen
USB3.0 Support                         [Enabled]        ↑ ↓ Select Item
XHCI Hand-off                          [Enabled]        Enter: Select
EHCI Hand-off                          [Disabled]       +-  Change Field
USB Mass Storage Driv                  [Enabled]        F1: General Help
                                                        F2: Previous Values
USB hardware delays and time-outs:                      F3: Optimized Default
USB Transfer time-out                  [20 sec]         F4: Save  ESC: Exit
Device reset tine-out                  [20 sec]
Device power-up delay                  [Auto]
```

Legacy USB Support

Enables Legacy USB support.
AUTO option disables legacy support if no USB devices are connected.
DISABLE option will keep USB devices available only for EFI applications.

USB3.0 Support

Enable/Disable USB3.0 (XHCI) Controller support.

XHCI Hand-off

This is a workaround for OSes without XHCI hand-off support. The XHCI ownership change should be claimed by XHCI driver.

EHCI Hand-off

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

Port 64/60 Emulation

Enables I/O port 60h/64h emulation support. This should be enabled for the complete USB keyboard legacy support for non-USB aware OSes.

USB Transfer time-out

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

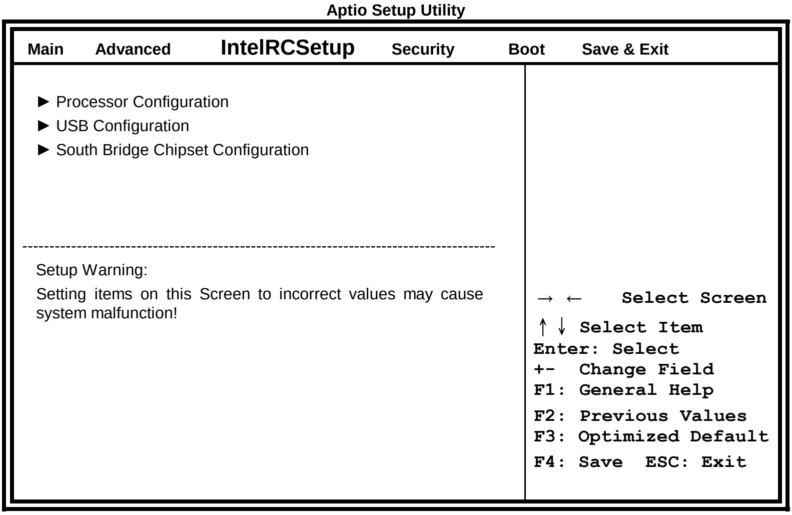
Device reset tine-out

USB mass Storage device start Unit command time-out.

Device power-up delay

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

IntelRCSetup



Aptio Setup Utility

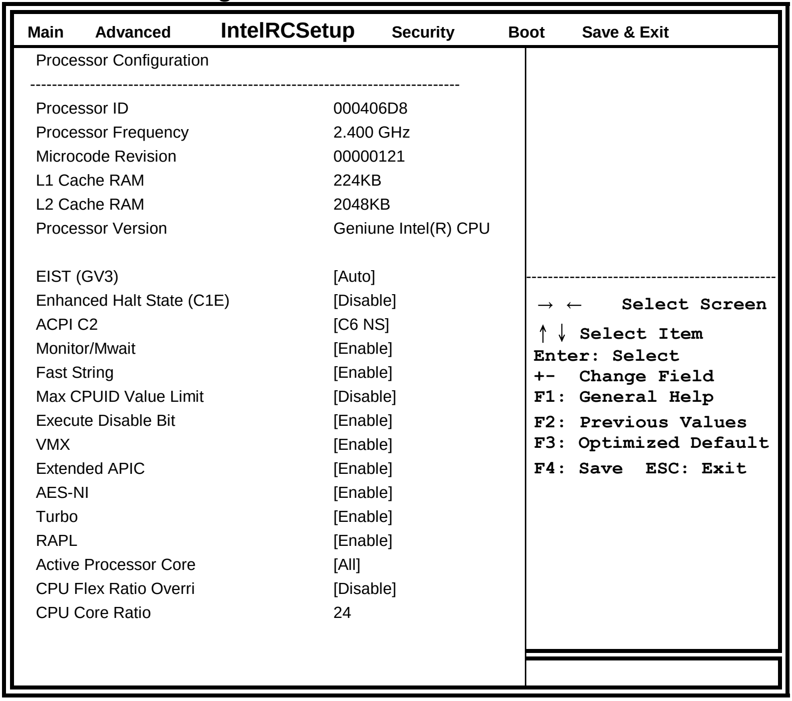
► Processor Configuration
► USB Configuration
► South Bridge Chipset Configuration

Setup Warning:

Setting items on this Screen to incorrect values may cause system malfunction!

```
→ ←    Select Screen
↑ ↓ Select Item
Enter: Select
+-  Change Field
F1: General Help
F2: Previous Values
F3: Optimized Default
F4: Save  ESC: Exit
```

Processor Configuration



```
Main    Advanced    IntelRCSetup    Security    Boot    Save & Exit

 Processor Configuration
--------------------------------------------------------------------
 Processor ID                         000406D8
 Processor Frequency                  2.400 GHz
 Microcode Revision                   00000121
 L1 Cache RAM                         224KB
 L2 Cache RAM                         2048KB
 Processor Version                    Geniune Intel(R) CPU

 EIST (GV3)                           [Auto]
 Enhanced Halt State (C1E)            [Disable]      → ←   Select Screen
 ACPI C2                              [C6 NS]        ↑ ↓ Select Item
 Monitor/Mwait                        [Enable]       Enter: Select
 Fast String                          [Enable]       +-  Change Field
 Max CPUID Value Limit                [Disable]      F1: General Help
 Execute Disable Bit                  [Enable]       F2: Previous Values
 VMX                                  [Enable]       F3: Optimized Default
 Extended APIC                        [Enable]       F4: Save  ESC: Exit
 AES-NI                               [Enable]
 Turbo                                [Enable]
 RAPL                                 [Enable]
 Active Processor Core                [All]
 CPU Flex Ratio Overri                [Disable]
 CPU Core Ratio                       24
```

EIST (GV3)

Enable/Disable EIST. GV3 and TM1 must be enabled for TM2 to be available. GV3 must be enabled for Turbo. Auto – Enable for B0 CPU stepping, all others disabled, change setting to override.

Fast String

When enabled, enable fast string for REP MOVS/STOS.

MAX CPUID Value Limit

This should be enabled in order to boot legacy OSes that cannot support CPUs with extended CPUID functions.

Execute Disable Bit

When disabled, force the XD feature flag to always return 0.

VMX

Enables the Vanderpool Technology, takes effect after reboot.

Extended APIC

Enable/Disable extended APIC support.

AES-NI

Enable/Disable AES-NI support

Turbo

Enable or Disable CPU Turbo capability. This option only applies to ES2 and above.

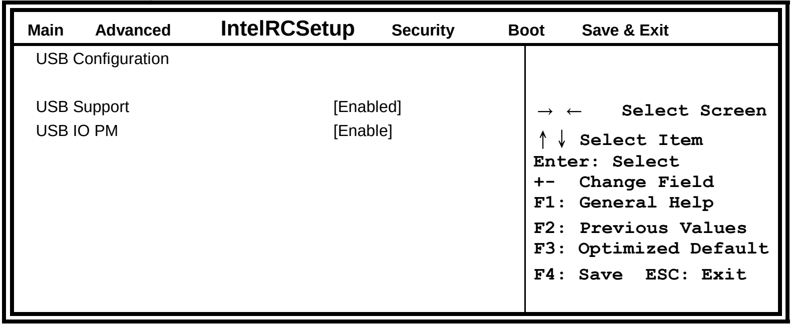
Active Processor Core

Number of cores to enable in SoC package.

CPU Flex Ratio Override

Enable/Disable CPU Flex Ration Programming.

USB Configuration



```
Main    Advanced    IntelRCSetup    Security    Boot    Save & Exit

USB Configuration

USB Support                     [Enabled]          → ←    Select Screen
USB IO PM                       [Enable]           ↑ ↓ Select Item
                                                   Enter: Select
                                                   +-  Change Field
                                                   F1: General Help
                                                   F2: Previous Values
                                                   F3: Optimized Default
                                                   F4: Save  ESC: Exit
```

USB Support

USB Support Parameters.

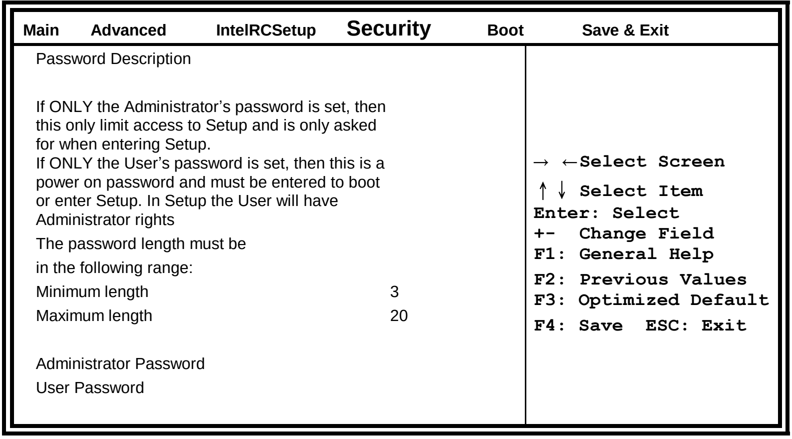
USB IO PM

Enable/Disable IO PM.

Security

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

Aptio Setup Utility



```
Main   Advanced   IntelRCSetup   Security   Boot   Save & Exit

Password Description                                 → ←Select Screen
                                                     ↑ ↓ Select Item
If ONLY the Administrator's password is set, then    Enter: Select
this only limit access to Setup and is only asked    +-  Change Field
for when entering Setup.                             F1: General Help
If ONLY the User's password is set, then this is a   F2: Previous Values
power on password and must be entered to boot        F3: Optimized Default
or enter Setup. In Setup the User will have          F4: Save  ESC: Exit
Administrator rights
The password length must be
in the following range:
Minimum length                               3
Maximum length                               20

Administrator Password
User Password
```

Administrator Password

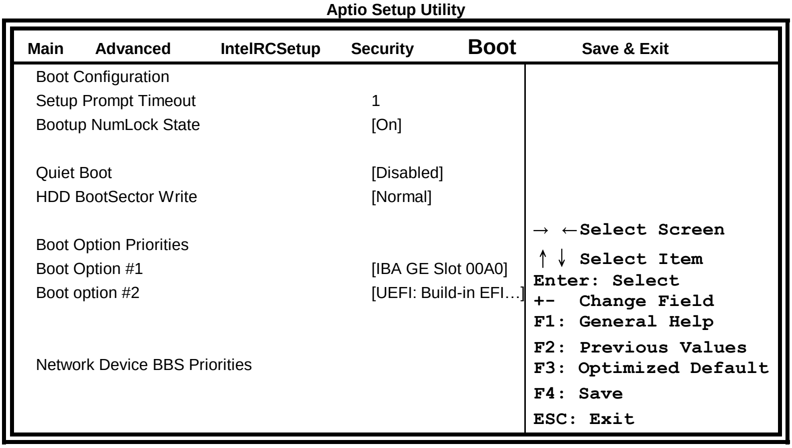
Set Setup Administrator Password.

User Password

Set User Password.

Boot Settings

Aptio Setup Utility



```
Main   Advanced   IntelRCSetup   Security   Boot   Save & Exit

Boot Configuration
Setup Prompt Timeout                  1
Bootup NumLock State                  [On]

Quiet Boot                            [Disabled]
HDD BootSector Write                  [Normal]
                                                      → ←Select Screen
Boot Option Priorities
Boot Option #1                        [IBA GE Slot 00A0]   ↑ ↓ Select Item
Boot option #2                        [UEFI: Build-in EFI…] Enter: Select
                                                      +-  Change Field
                                                      F1: General Help
                                                      F2: Previous Values
Network Device BBS Priorities                         F3: Optimized Default
                                                      F4: Save
                                                      ESC: Exit
```

Setup Prompt Timeout

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

Bootup NumLock State

Select the keyboard NumLock state.

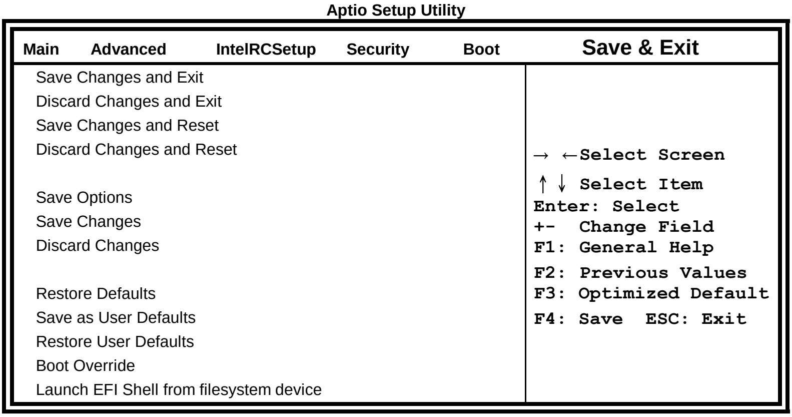
Quiet Boot

Enables/Disables Quiet Boot option.

Boot Option Priorities

Sets the system boot order.

Save & Exit Settings



```
                              Aptio Setup Utility
Main   Advanced   IntelRCSetup   Security   Boot        Save & Exit
  Save Changes and Exit
  Discard Changes and Exit
  Save Changes and Reset
  Discard Changes and Reset                     → ←Select Screen

  Save Options                                  ↑ ↓ Select Item
  Save Changes                                  Enter: Select
  Discard Changes                               +-  Change Field
                                                F1: General Help
                                                F2: Previous Values
  Restore Defaults                              F3: Optimized Default
  Save as User Defaults                         F4: Save  ESC: Exit
  Restore User Defaults
  Boot Override
  Launch EFI Shell from filesystem device
```

Save Changes and Exit

Exit system setup after saving the changes.

Discard Changes and Exit

Exit system setup without saving any changes.

Save Changes and Reset

Reset the system after saving the changes.

Discard Changes and Reset

Reset system setup without saving any changes.

Save Changes

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

Discard Changes

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

Restore Defaults

Restore/Load Defaults values for all the setup options.

Save as User Defaults

Save the changes done so far as User Defaults.

Restore User Defaults

Restore the User Defaults to all the setup options.

Drivers Installation

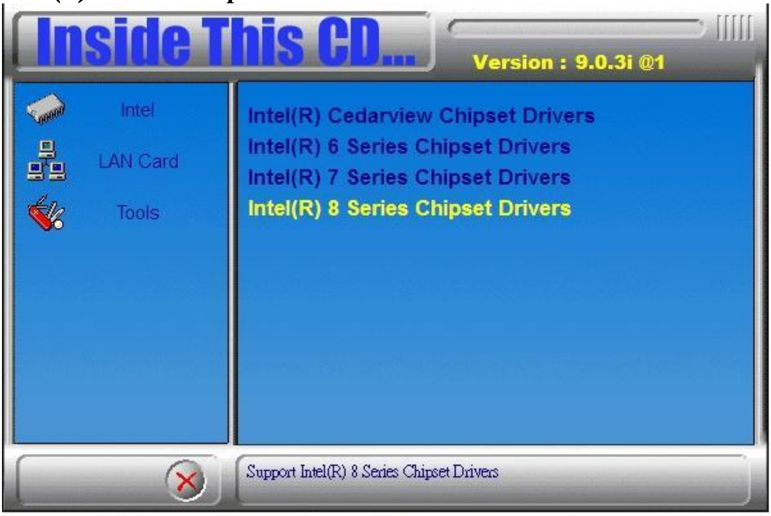
This section describes the installation procedures for software and drivers. The software and drivers are included with the motherboard. If you find the items missing, please contact the vendor where you made the purchase. The contents of this section include the following:

LAN Drivers Installation..45

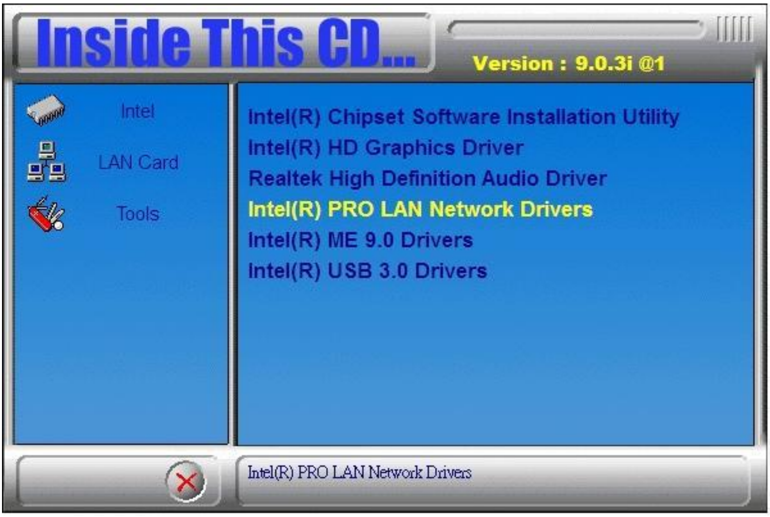
IMPORTANT NOTE:
After installing your Windows operating system, you must install first the Intel Chipset Software Installation Utility before proceeding with the drivers installation.

LAN Drivers Installation

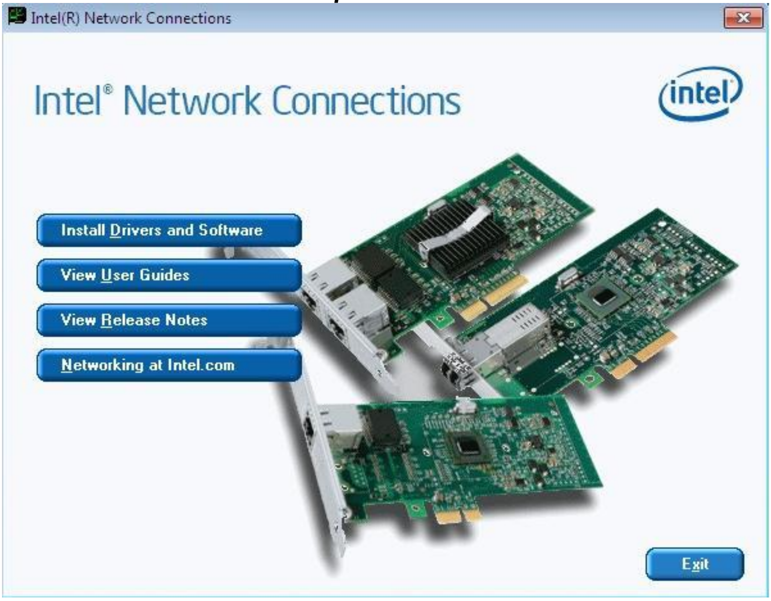
1. Insert the DVD that comes with the board. Click ***Intel*** and then ***Intel(R) 8 Series Chipset Drivers***.


Inside This CD...
Version : 9.0.3i @1
Intel
LAN Card
Tools
Intel(R) Cedarview Chipset Drivers
Intel(R) 6 Series Chipset Drivers
Intel(R) 7 Series Chipset Drivers
Intel(R) 8 Series Chipset Drivers
Support Intel(R) 8 Series Chipset Drivers

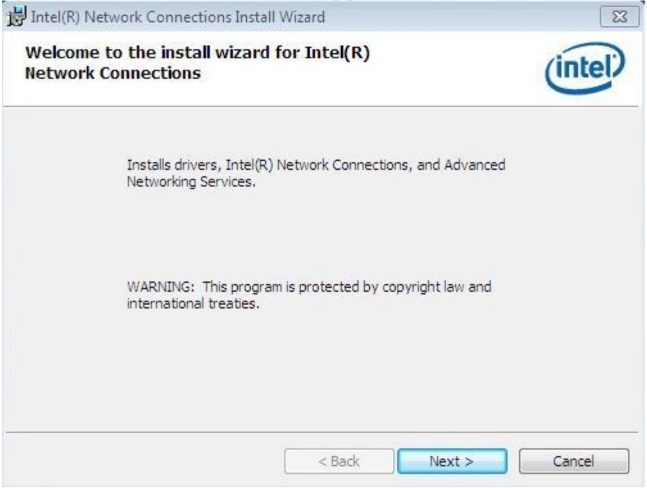
2. Click ***Intel(R) PRO LAN Network Driver.***


Inside This CD...
Version : 9.0.3i @1
Intel
LAN Card
Tools
Intel(R) Chipset Software Installation Utility
Intel(R) HD Graphics Driver
Realtek High Definition Audio Driver
Intel(R) PRO LAN Network Drivers
Intel(R) ME 9.0 Drivers
Intel(R) USB 3.0 Drivers
Intel(R) PRO LAN Network Drivers

3. Click ***Install Drivers and Software.***


Intel(R) Network Connections
Intel® Network Connections
intel
Install Drivers and Software
View User Guides
View Release Notes
Networking at Intel.com
Exit

4. When the Welcome screen appears, click ***Next***.



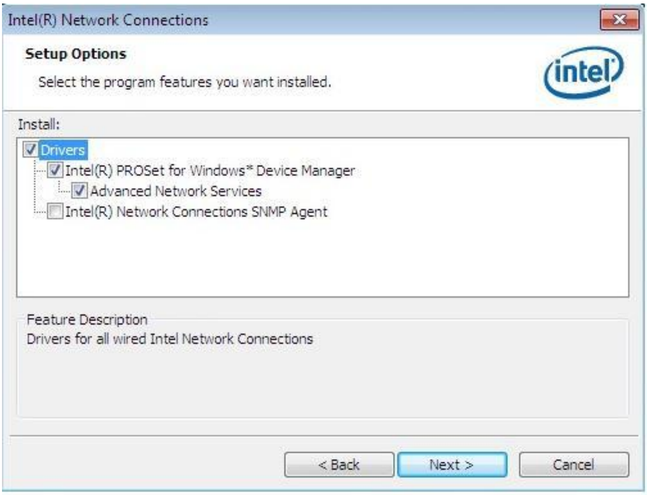
Welcome to the install wizard for Intel(R) Network Connections

Installs drivers, Intel(R) Network Connections, and Advanced Networking Services.

WARNING: This program is protected by copyright law and international treaties.

5. Click ***Next*** to to agree with the license agreement.

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



Intel(R) Network Connections

Setup Options

Select the program features you want installed.

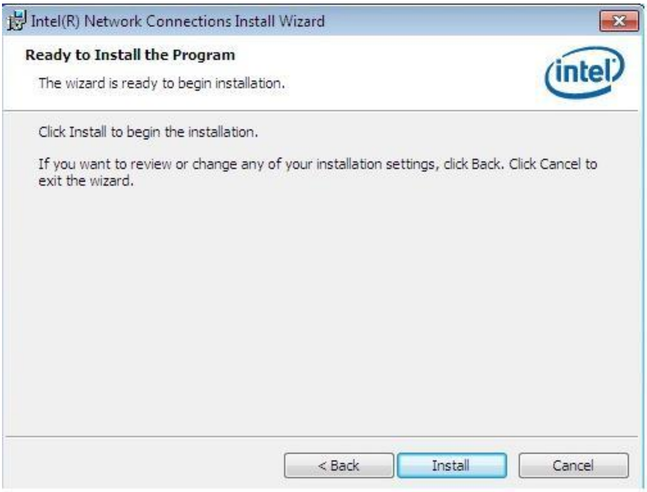
Install:

- ☑ Drivers
  - ☑ Intel(R) PROSet for Windows* Device Manager
    - ☑ Advanced Network Services
  - □ Intel(R) Network Connections SNMP Agent

Feature Description

Drivers for all wired Intel Network Connections

7. The wizard is ready to begin installation. Click ***Install*** to begin the installation.



Intel(R) Network Connections Install Wizard

Ready to Install the Program

The wizard is ready to begin installation.

Click Install to begin the installation.

If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.

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

Appendix

A. I/O Port Address Map

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

Address	Device Description
000h - 01Fh	DMA Controller #1
020h - 03Fh	Interrupt Controller #1
040h - 05Fh	System Timer
060h - 064h	Keyboard Controller
070h - 07Fh	Real Time Clock
080h - 09Fh	DMA Controller #2
0A0h - 0BFh	Interrupt Controller #2
0C0h - 0DFh	DMA Controller #3
0F0h	Clear Math Coprocessor Busy Signal
0F1h	Reset Math Coprocessor
3B0h- 3DFh	Graphics adapter Controller
3F8h - 3FFh	Serial Port #0 (COM1)
2F8h - 2FFh	Serial Port #1 (COM2)
3E8h – 3EFh	Serial Port #2 (COM3)
2E8h – 2EFh	Serial Port #3 (COM4)
3E8h - 3EFh	Serial Port #4 (COM5)
2E8h - 2EFh	Serial Port #5 (COM6)
360h - 36Fh	Network Ports

B. Interrupt Request Lines (IRQ)

Peripheral devices use interrupt request lines to notify CPU for the service required. The following table shows the IRQ used by the devices on board.

Level	Function
IRQ0	System Timer Output
IRQ1	Keyboard
IRQ4	Serial Port #1
IRQ3	Serial Port #2
IRQ5	Serial Port #3
IRQ11	Serial Port #4
IRQ8	Real Time Clock
IRQ14	Primary IDE
IRQ15	Secondary IDE

C. Register of the LAN Bypass Controller

To fulfill the varied requests on LAN Bypass controller, IBASE provide the smart LAN Bypass controller. User can define the Bypass function behavior when the system is power-on, power-off and WDT signal is asserted.

The controller is behind the SMBus controller. The I^{2}C address is listed as below:

	I^{2}C Address (8bit)	Remark
1st Controller	0x68	
2nd Controller	0x6A	Optional

CR 0x22 : System-On Bypass Register

Attribute : Read / Write

Reset default : 0x0000

Bit	LAN Port#	Read / Write	Description
0	Eth1, 2	Read / Write	Enable / Disable LAN Bypass function when the system is power On. 1 = Enable LAN Bypass function 0 = Disable LAN Bypass function

CR 0x24 : System-Off Bypass Register

Attribute : Read / Write

Reset default : 0x0000

Bit	LAN Port#	Read / Write	Description
0	Eth1, 2	Read / Write	Enable / Disable LAN Bypass function when the system is power Off. 1 = Enable LAN Bypass function 0 = Disable LAN Bypass function

CR 0x26 : Watchdog (WDT) WDT_IN# Signal Control Register

Attribute : Read / Write

Reset default : 0x00

Bit	Read / Write	Description
0	Read / Write	**WDT_OUT# Generator** The capacity use the WDT to reset the system 1 = Generate 100ms pulse to reset signal when WDT signal is asserted. 0 = Ignore the WDT signal.
1	Read / Write	**WDT LAN Bypass Enable** The capacity use the WDT to set LAN Bypass function 1 = Enable LAN Bypass function when the WDT signal is asserted. **CR 0x28** and **CR 0x2A** will be available if this bit is set to "1". 0 = Disable WDT LAN Bypass function.
2 ~ 7		Reserved

CR 0x28 : Watchdog (WDT) Bypass Control Register

Attribute : Read / Write

Reset default : 0x0000

Bit	LAN Port#	Read / Write	Description
0	Eth1, 2	Read / Write	Enable / Disable WDT Bypass function for each LAN port. 1 = Follow the setting in "WDT Bypass Register **CR 0x2A**" when the WDT signal is asserted. 0 = Ignore to control the bypass when the WDT is asserted.

CR 0x2A : Watchdog (WDT) Bypass Register

Attribute : Read / Write

Reset default : 0x0000

Bit	LAN Port#	Read / Write	Description
0	Eth1, 2	Read / Write	The function works when the bit in **CR 0x28** is “1”. It controls LAN Bypass function should be Enabled / Disabled when the WDT signal is asserted. If the bit is set to "1" in "WDT Bypass Control Register **CR 0x28**", it will follow below setting: 1 = Enable LAN Bypass function 0 = Disable LAN Bypass function