PIR-230 Series

User Manual

Warranty

All products manufactured by ICP DAS are under warranty regarding defective materials for a period of one year, beginning from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for any damage resulting from the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, or for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Trademarks

Names are used for identification purposes only and may be registered trademarks of their respective companies.

Contact Us

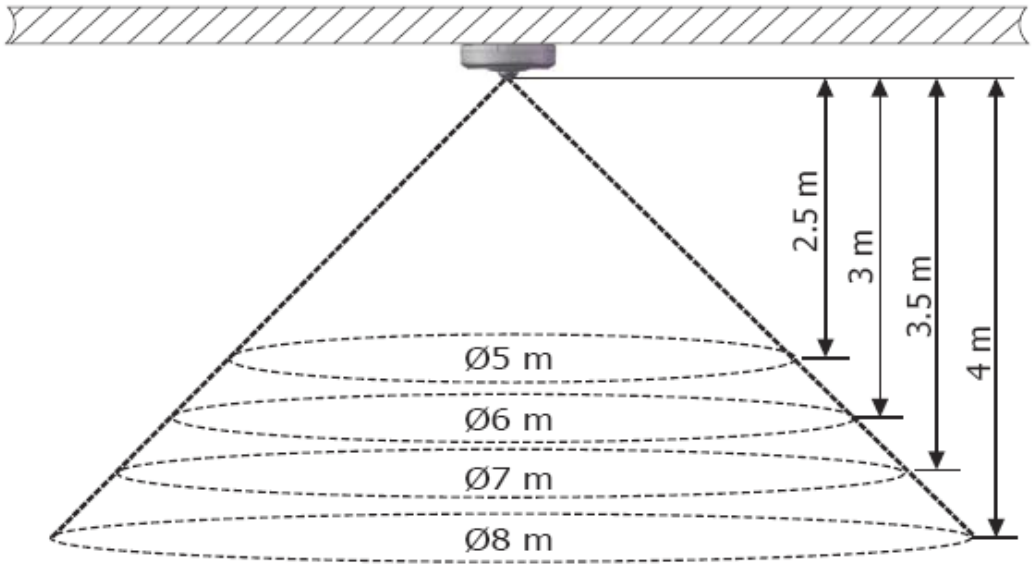
If you have any problems, please feel free to contact us by email at:
service@icpdas.com.

Table of Contents

Introduction

The PIR-230 series module includes a 1-channel passive infrared (PIR) sensor that is able to detect infrared waves generated by human within a range of approximately 8 meters in diameter with a 360° coverage area. The PIR-230 series is used for indoor motion detection, and can be configured to automatically switch on a light if motion is detected.

1-channel temperature sensor is provided for measuring room temperature, it can also be used as a heat detector to activate the fire alarm system.



The PIR-230 series contains RS-485, Ethernet and PoE communication interfaces, the most common communication interfaces in industrial network. With additional Wi-Fi interface, the PIR-230-WF series provides a WLAN connection which makes an easy way to incorporate wireless connectivity into monitoring and control systems.

Compatible with IEEE 802.11b/g/n standards
The PIR-230-WF modules are complied with IEEE 802.11b/g/n standard from 2.4~2.5 GHz. It can be used to provide up to 11 Mbps for IEEE 802.11b and 54 Mbps for IEEE 802.11g to connect to your wireless LAN.

2. Hardware Information

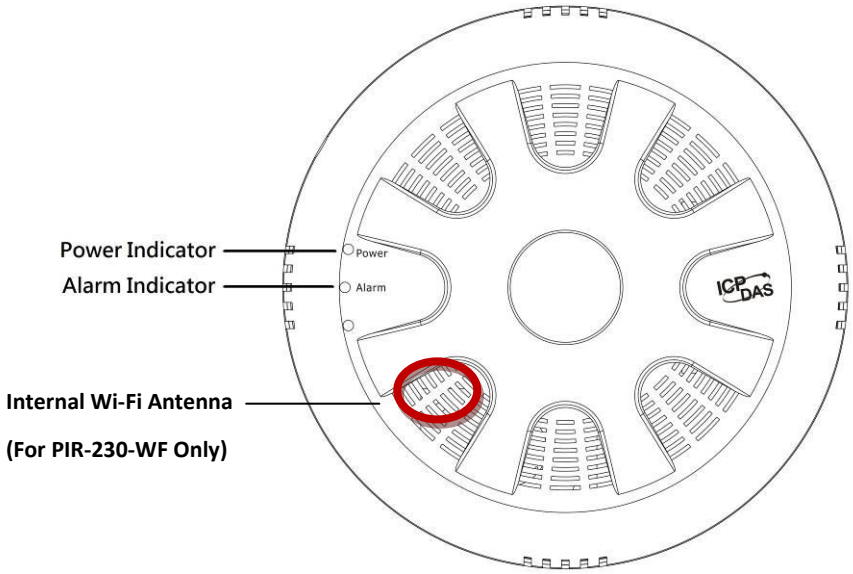
2.1. Specifications

Model		PIR-230-E	PIR-230-BLE	PIR-230-WF
PIR				
Time-delay	Hardware	8-step Switch-selectable (seconds): 6, 16, 33, 66, 131, 262, 524, 1049		
	Software	16-step (seconds): 2, 4, 6, 8, 16, 33, 49, 66, 131, 262, 393, 524, 1049, 2097, 3146, 4194		
LUX Control Level	Hardware	2 (Dawn and dust)		
	Software	5 levels adjustable		
Detection Range		Distance: 4 meters Max.		
Detection Field of View		360º; Diameter 8 meters Max.		
Temperature				
Measurement Range		-40 ~ +120°C		
Fire Alarm		65°C (Programmable)		
Resolution		0.01°C		
Accuracy		± 0.5°C		
Relative Humidity				
Range		0 to 100% RH		
Resolution		0.01% RH		
Accuracy		± 5% RH		
Digital Input				
Channel		2		
Type		Wet		
On Voltage Level		70 VAC to 240 VAC		
Off Voltage Level		20 VAC Max.		
Input Impedance		68 KΩ, 1 W		
Relay Output				
Channel		1		
Type		Power Relay, Form C		
Max. Load Current		NO: 10 A @ 250 VAC NC: 6 A @ 250 VAC		
Load Wattage		Incandescent Bulb: 1500 W Max.; Fluorescent Lamp 300 W Max.		

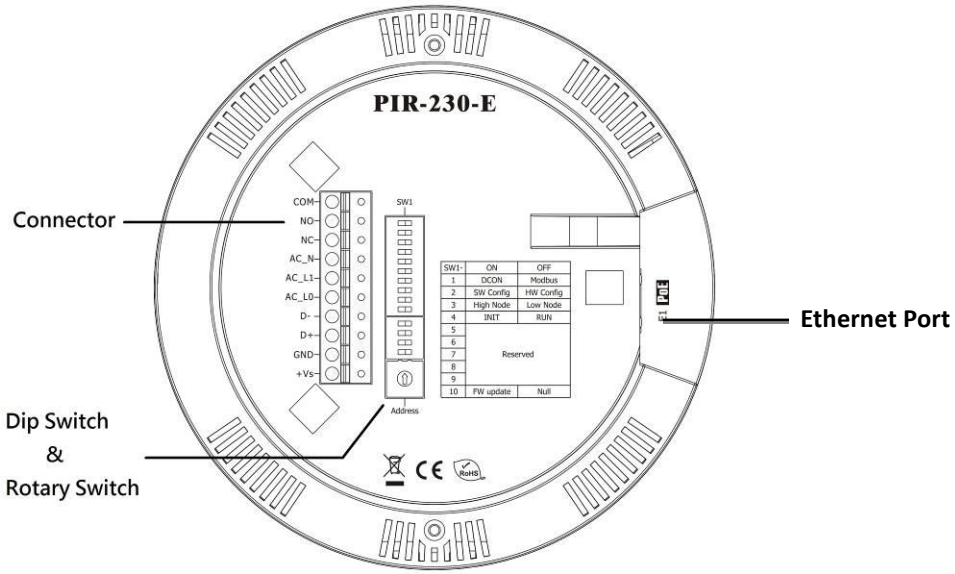
Model	PIR-230-E	PIR-230-BLE	PIR-230-WF
Communication			
Node Address	Hardware: 160 to 191 / Software:1 to 255		
Protocol	DCON, Modbus RTU, Modbus TCP, MQTT		
Wireless interface	-	Bluetooth	Wi-Fi
Standard Supported	-	BT 4.0	IEEE 802.11 b/g/n
Wireless Mode	-	Slave	Infrastructure/ Limited AP
Wireless Security	-	AES 128	WEP, WPA, WPA2
Transmission Range	-	20 M(LOS)	50 M(LOS)
LED Display			
System LED Indictor	1 LED as Power/Communication Indicator		
I/O LED Indicator	1 LED as Alarm Indicator		
EMS Protection			
ESD (IEC 61000-4-2)	±4 kV Contact for each Terminal, ±8 kV Air for Random Point		
EFT (IEC 61000-4-4)	±4 kV for Power Line		
Power Requirements			
Reverse Polarity Protection	Yes		
Powered from Terminal Block	+10 to +48 VDC		
Powered from PoE	Yes, IEEE 802.3af, Class1		
Consumption	2 W	2.5 W	3 W
Mechanical			
Installation	Ceiling mounting		
Protection Class	IP20		
Dimensions (D x H)	Ø 150 mm x 53 mm		
Environment			
Operating Temperature	0 to +75°C		
Storage Temperature	-30 to +80°C		
Humidity	10 to 90% RH, Non-condensing		

2.2. Appearance

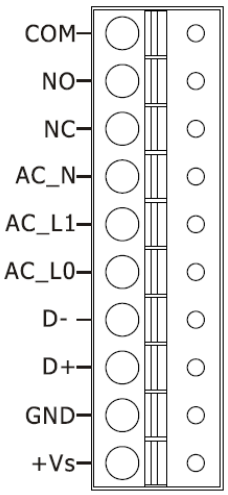
Front



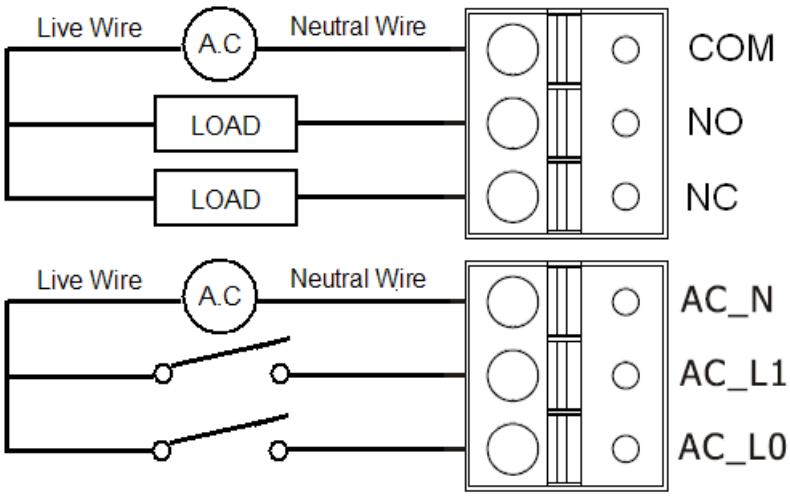
Rear



2.3. Pin Assignments

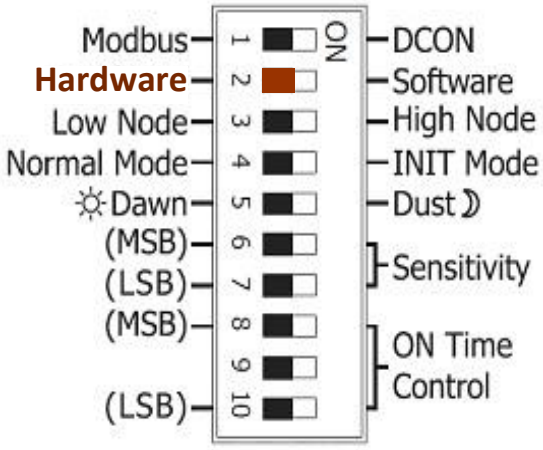
PIR-230-E	Pin	Description
	**COM**	Relay's Common Contact
	N.O	Relay's Normally Open Contact
	N.C	Relay's Normally Closed Contact
	AC_N	Naught Wire of Alternating Current
	AC_L1	Live Wire1 of Alternating Current
	AC_L0	Live Wire0 of Alternating Current
	D-	RS-485 Serial Communication Interface
	D+	
	GND	Ground
	+VS	Power Input (+10 to +48 VDC)

2.4. Wiring Connections



2.5. Hardware Configuration

DIP switches located on the rear side of the PIR-230 series module allow for configuration options. The switches are numbered 1 through 10 and can be set to ON or OFF. All the configuration will only take effect when the DIP[2] is set to OFF(Hardware) position. Following is more information on the DIP switch settings.



DIP [1]	**Protocol:** Used to specify the communication protocol to be used by the module ON: DCON OFF: Modbus RTU (default)
DIP [2]	**Configuration:** Used to specify the configuration settings for the module ON: Configure the module using DCON/Modbus commands (Software) OFF: Configure the module via DIP switch (Hardware, default)
DIP [3]	**Address:** Used to specify the module address when DIP [2] is set to OFF ON: Use rotary switch positions 0 to F for node addresses 176 to 191 OFF: Use rotary switch positions 0 to F for node addresses 160 to 175 (default)
DIP [4]	**Mode:** Used to specify the operating mode ON: Operating in INIT mode OFF: Operating in Normal mode (default)
DIP [5]	**PIR Operation:** Used to specify the luminance level at which the sensor will activate the light when movement is detected.

<table>
<tr><td></td><td>ON: The PIR sensor will only detect motion when the luminance level is between 0 and 200
OFF: The PIR sensor will detect motion continuously (default)</td></tr>
<tr><td>DIP [6:7]</td><td>Sensitivity:
Used to specify sensitivity of the PIR sensor, and adjust the detection range of the sensor
<table>
<tr><th>DIP 6</th><th>DIP 7</th><th>Sensitivity</th></tr>
<tr><td>OFF</td><td>OFF</td><td>Maximum (default)</td></tr>
<tr><td>OFF</td><td>ON</td><td>High</td></tr>
<tr><td>ON</td><td>OFF</td><td>Low</td></tr>
<tr><td>ON</td><td>ON</td><td>Minimum</td></tr>
</table></td></tr>
<tr><td>DIP [8:10]</td><td>ON Time Control:
Used to specify the ON time for the relay after the PIR sensor has been triggered.
<table>
<tr><th>DIP 8</th><th>DIP 9</th><th>DIP 10</th><th>ON Time</th></tr>
<tr><td>OFF</td><td>OFF</td><td>OFF</td><td>6 seconds (default)</td></tr>
<tr><td>OFF</td><td>OFF</td><td>ON</td><td>16 seconds</td></tr>
<tr><td>OFF</td><td>ON</td><td>OFF</td><td>33 seconds</td></tr>
<tr><td>OFF</td><td>ON</td><td>ON</td><td>66 seconds</td></tr>
<tr><td>ON</td><td>OFF</td><td>OFF</td><td>131 seconds</td></tr>
<tr><td>ON</td><td>OFF</td><td>ON</td><td>262 seconds</td></tr>
<tr><td>ON</td><td>ON</td><td>OFF</td><td>524 seconds</td></tr>
<tr><td>ON</td><td>ON</td><td>ON</td><td>1049 seconds</td></tr>
</table></td></tr>
</table>

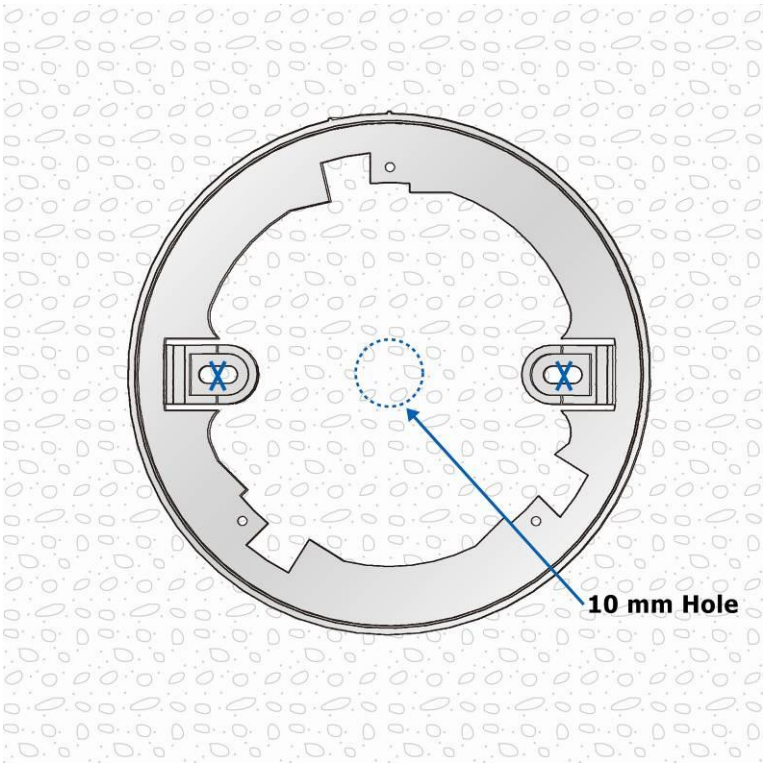
2.6. Hardware Installation

Installation Tips

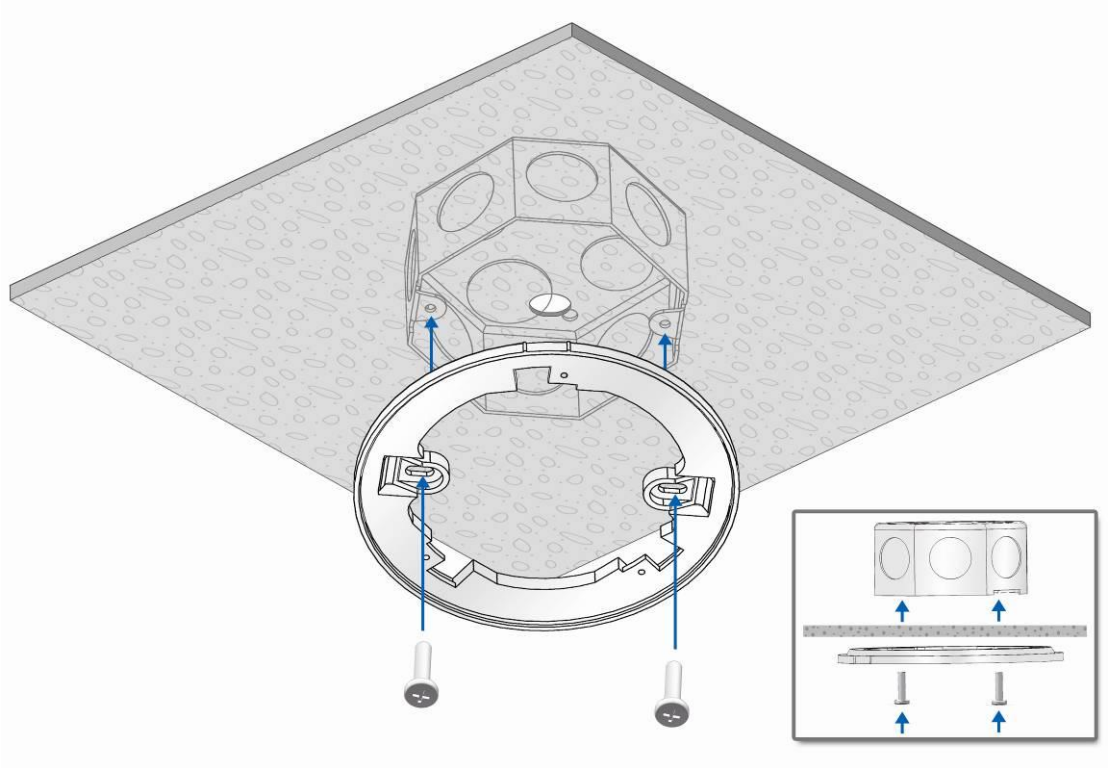
- Avoid installing the PIR-230 in areas where it will face direct or reflected sunlight.
- Avoid installing the PIR-230 in areas where the environmental temperature may change rapidly.
- Ensure that the PIR-230 is located at least one meter away from the nearest fluorescent light so as to avoid interference.
- Ensure that there are no obstructions in the field of view.

Installation Instructions

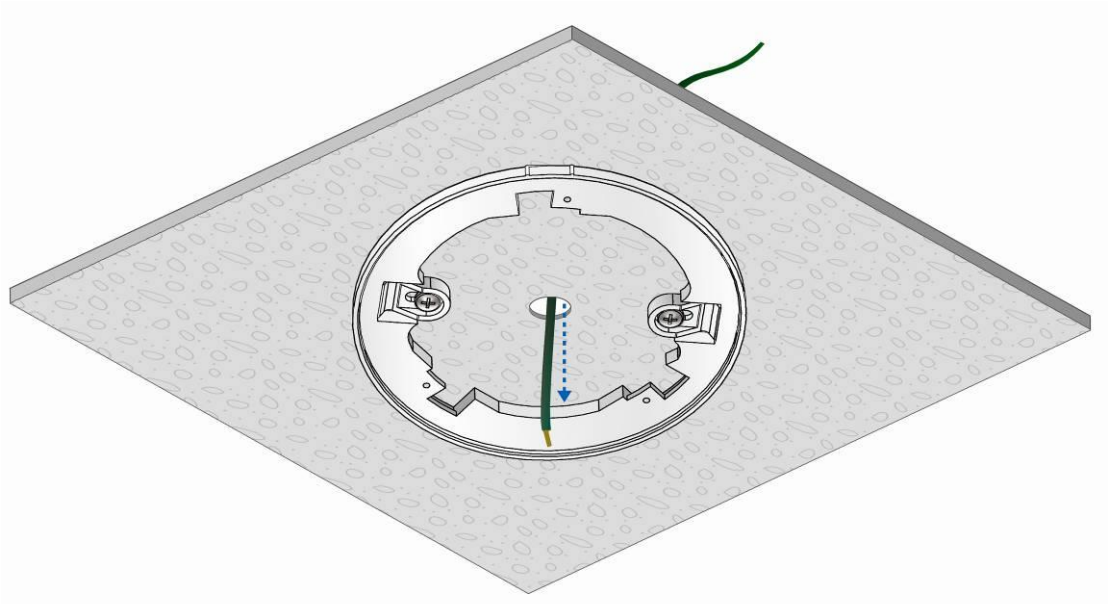
1. Position the Mounting Plate in the desired location. Mark the positions of the two screw holes and a 10 mm hole, as indicated below.



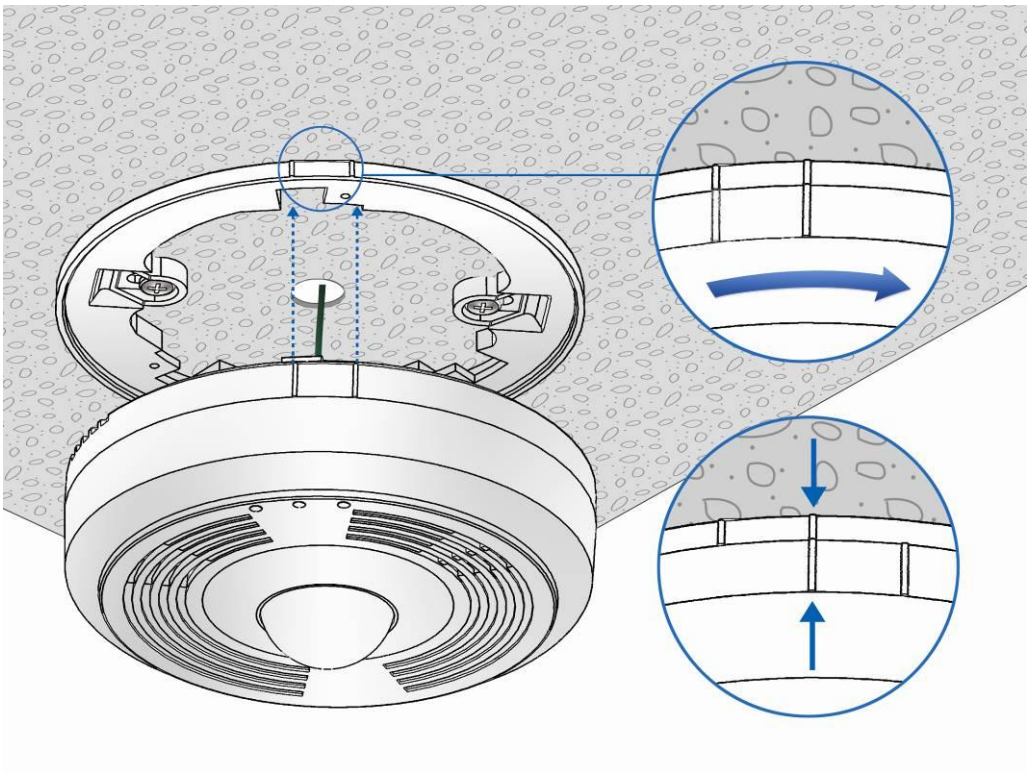
2. Secure the Mounting Plate to the ceiling using the M4x12 drywall screws and the optional octagonal box.



3. Feed the wires through the wiring hole.

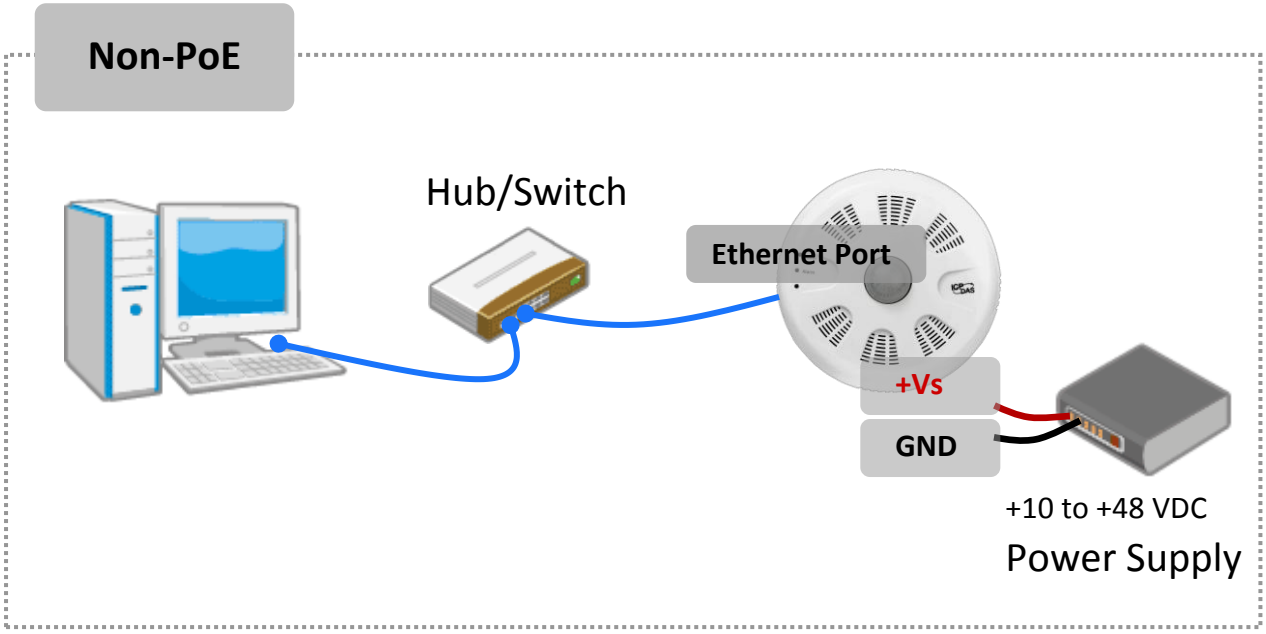


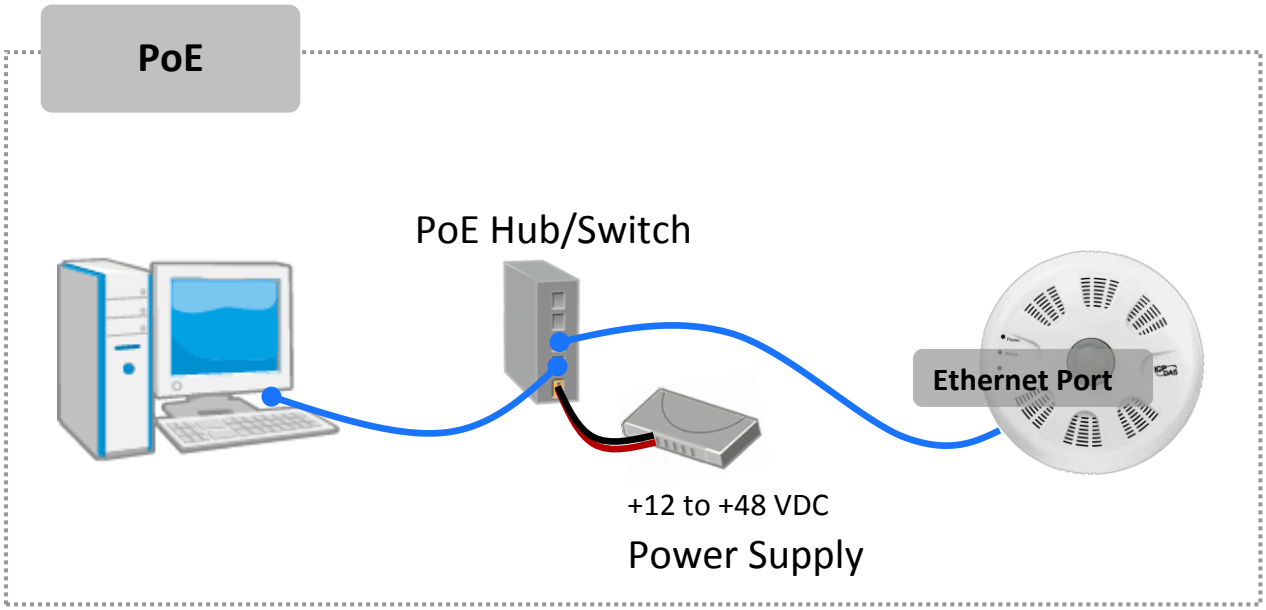
4. Connect all the wires to the appropriate locations on the connector.
5. Align the marks on the PIR-230 with the marks on the mounting Plate.
6. Rotate the PIR-230 clockwise until it locks into place.



3. Configuration via Web Browser

3.1 Connecting the Power and the Host PC





For connecting with PC via Wi-Fi

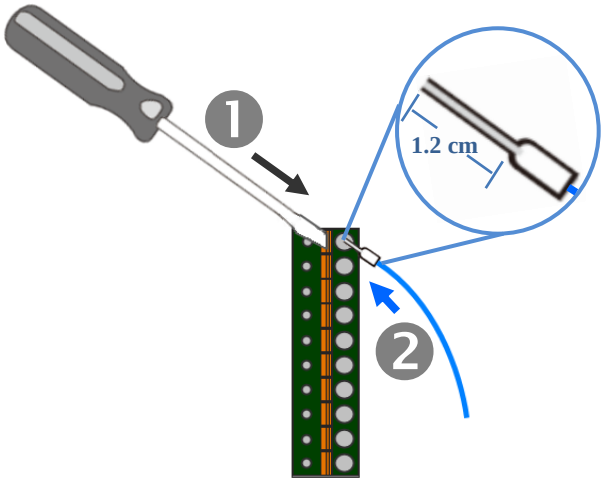
The PIR-230-WF logger can connect to the PC through Wi-Fi with power input requirement of +12 ~ +48 V_{DC}.

The PIR-230-WF device can be configured as station mode, such that the PC/Laptop can be connected through Wi-Fi AP.

The PIR-230-WF device can be configured as AP mode, such that the PC/Laptop can be connected through Wi-Fi directly. Only one device is allowed to be connected to the PIR-230-WF module in AP mode.

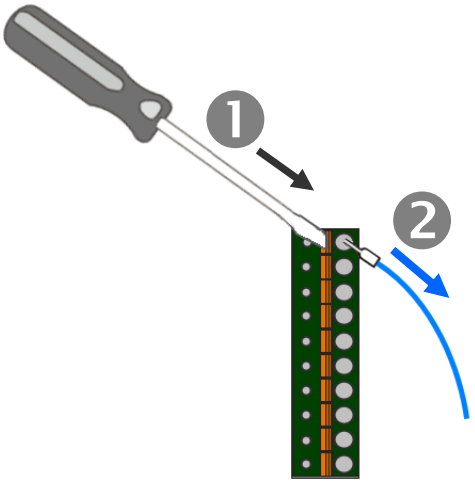
A tip for connecting the wire to the connector

1. Use the blade of the flat-head screwdriver to push down the wire clamp.
2. While holding the screwdriver in place, insert the wire into the terminal block.
3. Release the screwdriver.



A tip for removing the wire from the connector

1. Use the blade of the flat-head screwdriver to push down the wire clamp.
2. While holding the screwdriver in place, remove the wire from the terminal block.
3. Release the screwdriver.



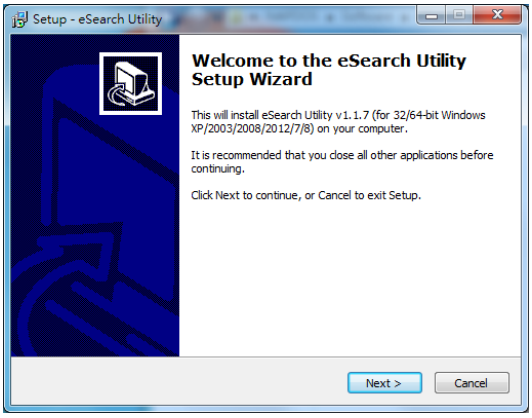
3.2. Network Configuration

Step 1: Get the eSearch Utility

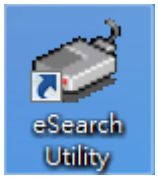
Download the eSearch Utility from

http://ftp.icpdas.com/pub/cd/iiot/utility/esearch/

Step 2: Install the eSearch utility

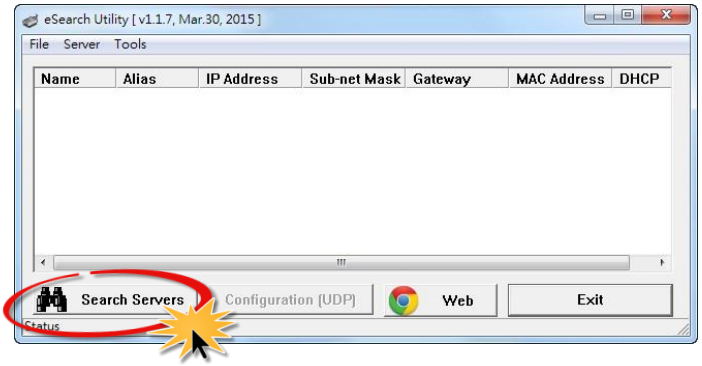


After the installation has been completed, a new short cut for the eSearch Utility will be displayed on your desktop.



Step 3: Search the PIR-230 series module on the Ethernet

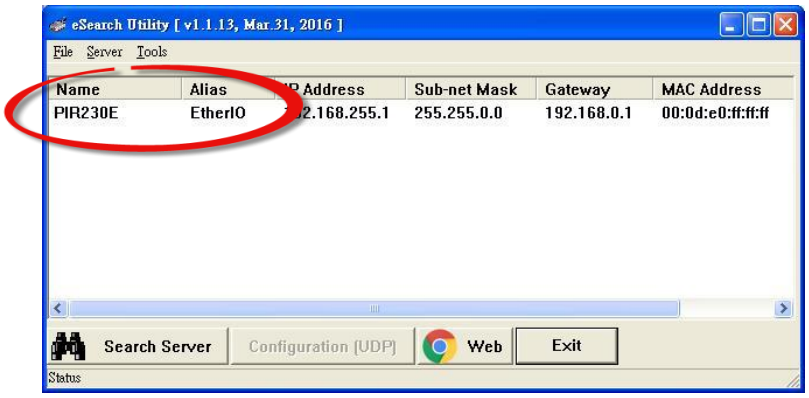
Launch eSearch Utility and click the “Search Servers” button to search for the PIR-230 module



Step 4: Double-click the name of the module to open the “Configure Server (UDP)” dialog box

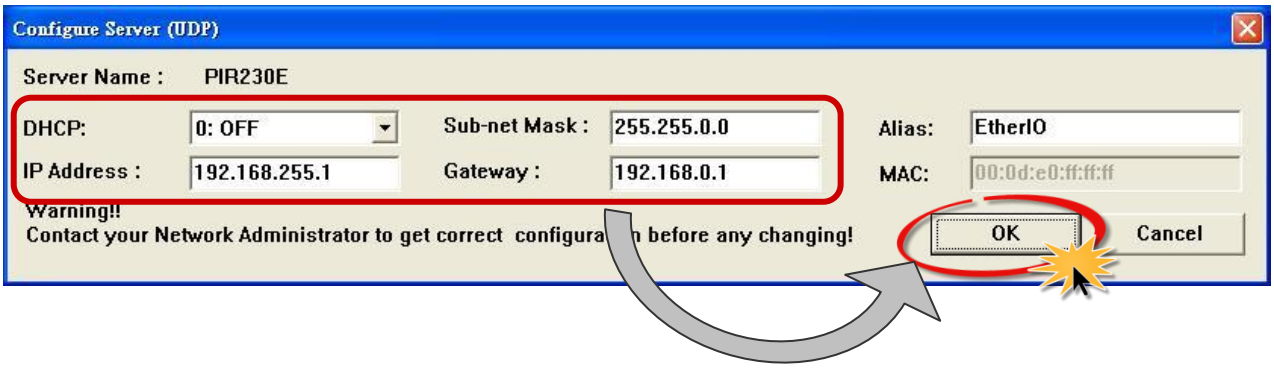
Factory Default Settings:

IP	192.168.255.1
Gateway	192.168.0.1
Mask	255.255.0.0

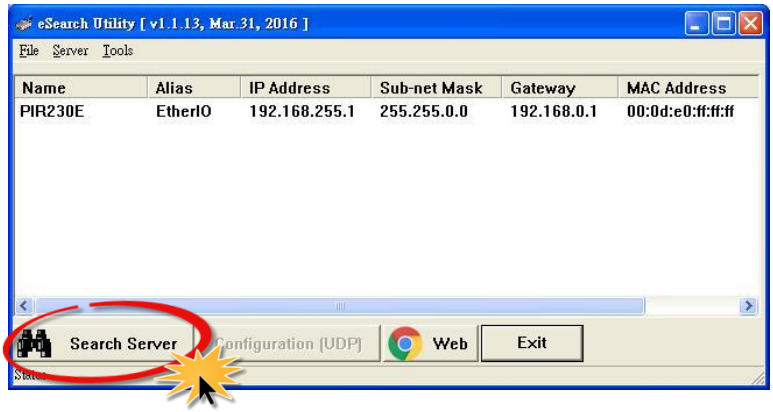


Step 5: Assign a new IP address

Enter valid **IP Address, Subnet Mask** and **Gateway** for your network, and then click the **"OK"** button. The new settings for the PIR-230 module will take effect within 2 seconds. If the correct network configuration information is unknown, contact the Network Administrator to obtain the relevant details.



Step 6: Wait for 2 seconds and then click the "Search Servers" button again to ensure that the PIR-230 module is operating correctly using the new configuration



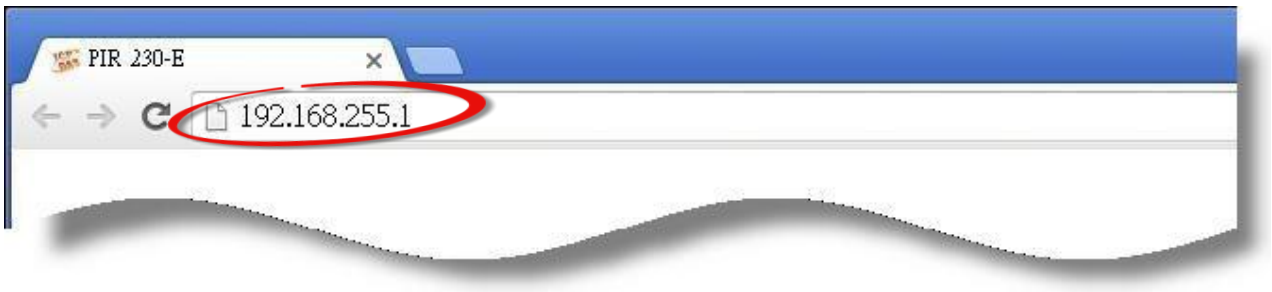
3.3. Logging into the PIR-230

Step 1: Open a new browser windows

Open a standard web browser. For example, Mozilla Firefox, Google Chrome and Internet Explorer are reliable and popular internet browsers that can be used to configure the PIR-230 module.

If you intend to use Internet Explorer, ensure that the cache to functions is disabled in order to avoid *browser access errors. Detailed information how to do this can be found in "FAQ_General_001: How to avoid a browser access error that causes a blank page to be displayed when using Internet Explorer".*

Step 2: Enter the new IP address for the PIR-230 and press the Enter key



Step 3: Enter the password to login to the web interface

Enter the password in the login password field (default is "**Admin**"), and then click the **"Submit"** button to enter the configuration web page.

3.4. Home

The first page displayed is Home, it shows the main ***Status & Configuration*** page.

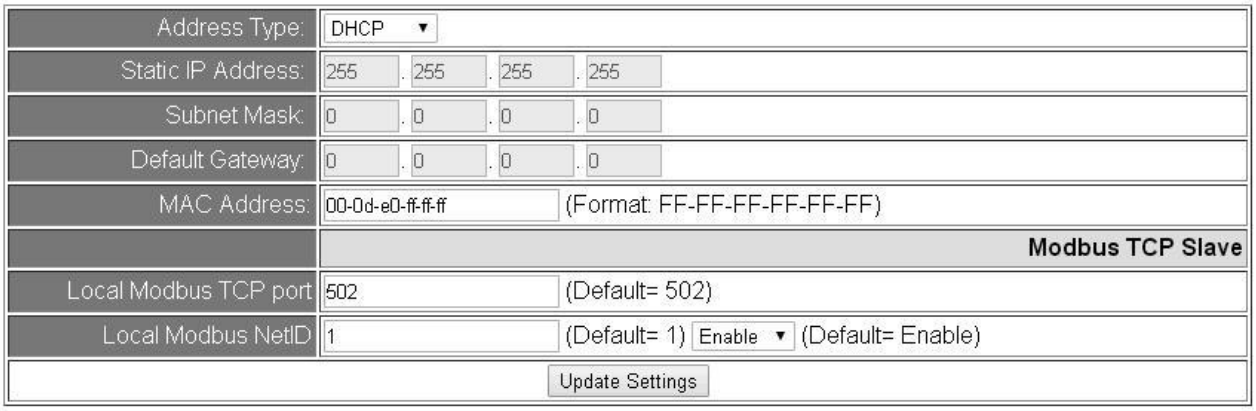
This section provides basic information related to the PIR-230 series module including the Model Name, Firmware version, IP Address, Initial Switch position, Alias Name, MAC Address, and the TCP Port and System Timeout values. If the firmware for the PIR-230 module is updated, you can check the version information here.

Model Name:	PIR230E	Alias Name:	EtherIO
Firmware Version:	A0.1 [Dec.29, 2015]	MAC Address:	00-0d-e0-ff-ff-ff
IP Address:	192.168.255.1	Initial Switch:	OFF
TCP Port Timeout (Socket Watchdog, Seconds):	180	System Timeout (Network Watchdog, Seconds)	0

3.5. Network

Clicking the ***Network*** tab to go to the page allowing you to verify the current settings, configure the IP Address and general parameters, and restore the default settings for the PIR-230 module, each of which will be described in more detail below.

3.5.1. IP Address Configuration



The following table provides an overview of the parameters contained in the ***IP Address Configuration*** section:

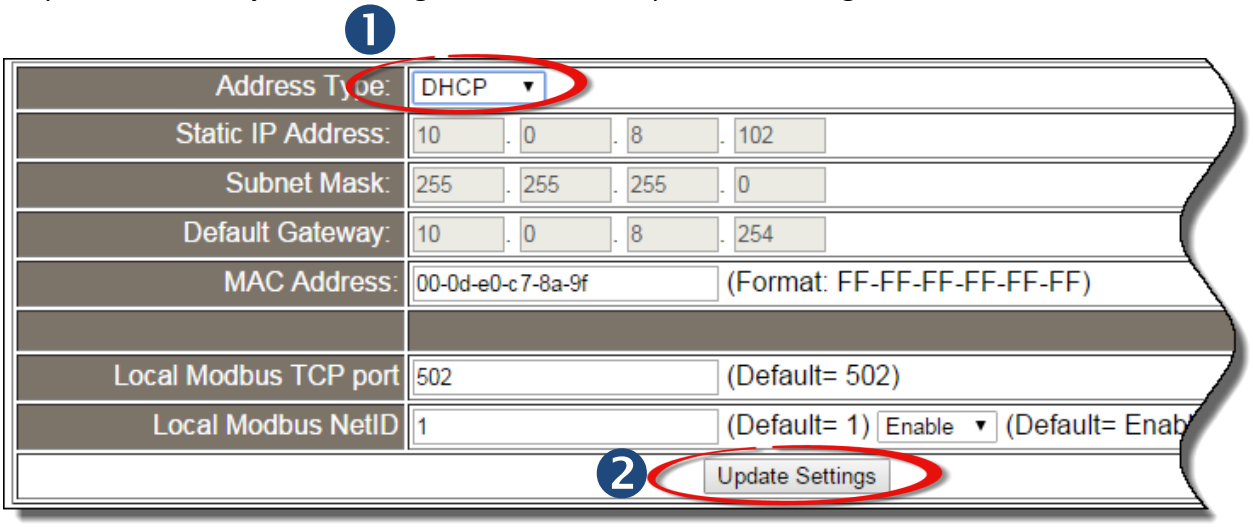
Item	Description
Address Type	**Static IP:** If there is no DHCP server installed in your network, you can configure the network settings manually. Refer to Section ***"Manual Configuration"*** below for more details.
	DHCP: Dynamic Host Configuration Protocol (DHCP) is a network application protocol that automatically assigns an IP address to each device. Refer to Section ***"DHCP Configuration"*** below for more details.
Static IP Address	Each PIR-230 module connected to the network must have its own unique IP address. This parameter is used to assign a specific IP address if there is no DHCP server on the network.
Subnet Mask	This parameter is used to assign the subnet mask for the PIR-230 module. The subnet mask indicates which portion of the IP address is used to identify the local network or subnet.
Default Gateway	This parameter is used to assign the IP Address of the Gateway to the PIR-230 module. A Gateway (or router) is a device that is used to connect an individual network to one or more additional networks.
MAC Address	This parameter is used to set the User-defined MAC address, which must be in the format FF-FF-FF-FF-FF-FF.
Modbus TCP Slave	
Local Modbus TCP port	This parameter is used to set the local port for Modbus communication. The default value is 502.
Local Modbus NetID	This parameter is used to set the Network ID for Modbus communication. The default value is 1.
	Enable option: the NetID will be checked when the PIR-230 module receives a Modbus command for identifying if to respond to this command. Disable option: the NetID will not be checked when the PIR-230 module receives a Modbus command. The PIR-230 module will respond to every command it receives.
Update Settings	Click this button to save the revised settings to the PIR-230 module.

DHCP Configuration

DHCP configuration is very easy to perform. If a DHCP server is connected to you network, network addresses will be dynamically configured after the following setting:

Step 1: Select "***DHCP***" from the ***Address Type*** drop-down menu

Step 2: Click the "***Update Settings***" button to complete the configuration



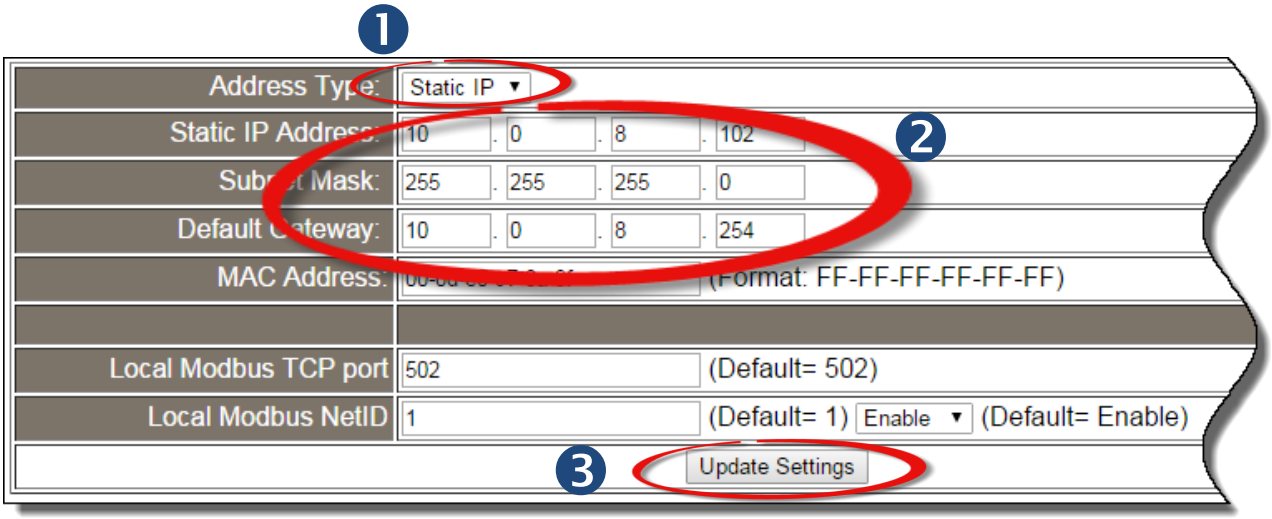
Manual Configuration

When using manual configuration, the network settings should be assigned as follows:

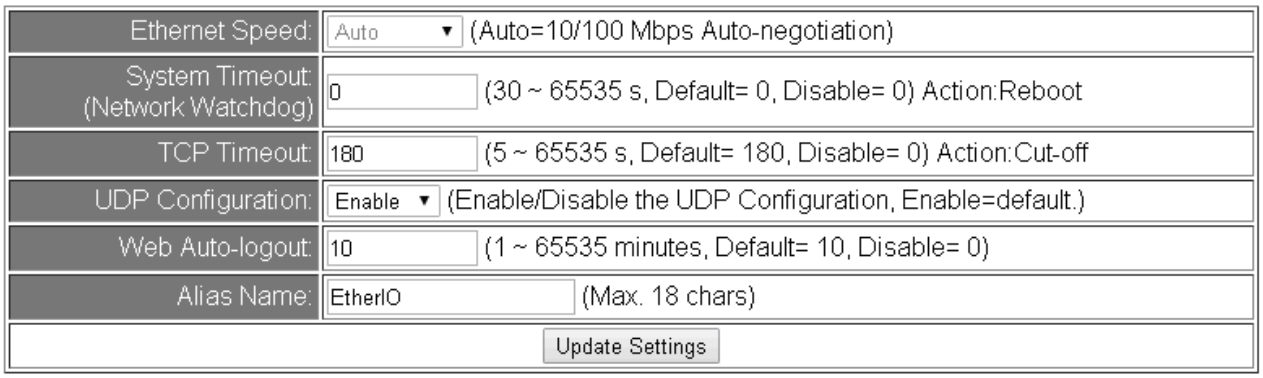
Step 1: Select "***Static IP***" from the ***Address Type*** drop-down menu

Step 2: Enter the relevant details in the respective network settings fields.

Step 3: Click the "***Update Settings***" button to complete the configuration



3.5.2. General Settings



The following table provides an overview of the parameters contained in the ***General Settings*** section:

Item	Description
Ethernet Speed	This parameter is used to set the Ethernet speed. The default value is Auto (Auto = 10/100 Mbps Auto-negotiation).
System Timeout (Network Watchdog)	This parameter is used to configure the system timeout value. If there is no activity on the network for a certain period of time, the system will be rebooted based on the configured system timeout value.
TCP Timeout (Seconds)	This parameter is used to configure the TCP timeout value. If Modbus TCP communication is idle for a certain period of time, the system will cut off the connection.
UDP Configuration	This parameter is used to enable or disable UDP configuration function.
Web Auto-logout	This parameter is used to configure the automatic logout value. If there is no activity on the web server for a certain period of time, the current user account will automatically logged out.
Alias Name	This parameter is used to assign an alias name for each PIR-230 module to assist with easy identification.
Update Settings	Click this button to save the revised settings to the PIR-230 module.

3.5.3. Restore Factory Defaults

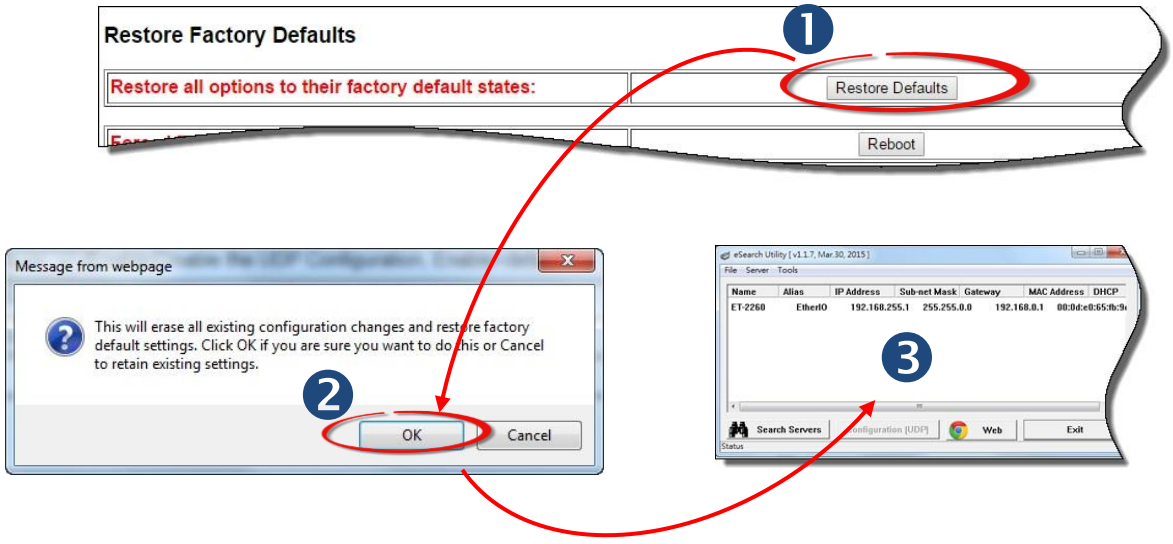
After performing the following operation, items will be restored to factory default settings as below:

Factory Default Settings	
IP Address	192.168.255.1
Gateway Address	192.168.0.1
Subnet Mask	255.255.0.0

Step 1: Click the **"*Restore Defaults*"** button to reset the configuration.

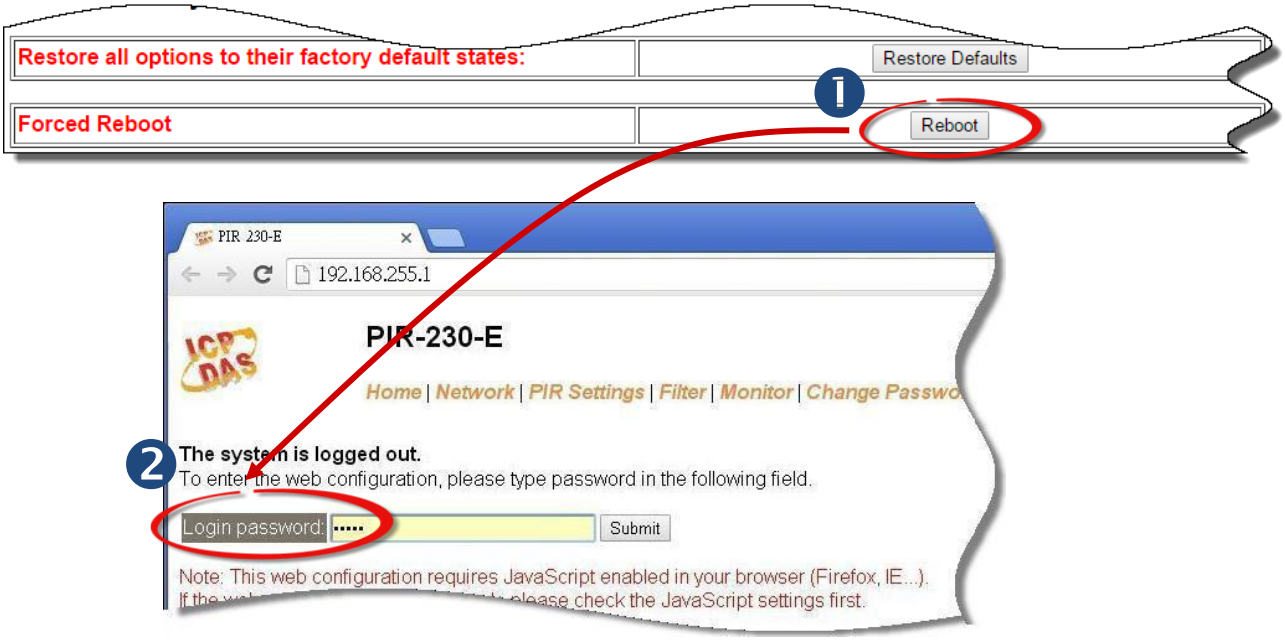
Step 2: Click the **"*OK*"** button in the message dialog box.

Step 3: Refer to step 3 and step 4 in Section "*3.2. Network Configuration*", to check whether the settings are restored to factory defaults.



3.5.4. Forced Reboot

The **Forced Reboot** function can be used to force the PIR-230 module to reboot or to remotely reboot the device. After the PIR-230 module has rebooted, the original login screen will be displayed and your Login Password will be requested.



3.5.5. Firmware Update

Click the Update button and then select the firmware file to update the firmware.

Firmware Update

If the remote firmware update is failed, then the traditional firmware update (on-site) is required to make the module working again. Step 1: Refer to firmware update manaul first. Step 2: Run eSearch Utility to prepare and wait for update. Step 3: Click the [**Update**] button to **reboot** the module and start update. Step 4: Configure the module again.	Update

The firmware can be obtained from web site:

http://ftp.icpdas.com/pub/cd/pir/pir-230-e/firmware/

3.6. PIR Settings

Clicking the **PIR Settings** tab to go to the **PIR Settings** page where you can configure the PIR sensor settings, temperature alarm settings and relay output, which will be described in more detail below.

3.6.1. Read I/O Status

DO Value	0x0
DI Value	0x0
PIR Status	Inactive

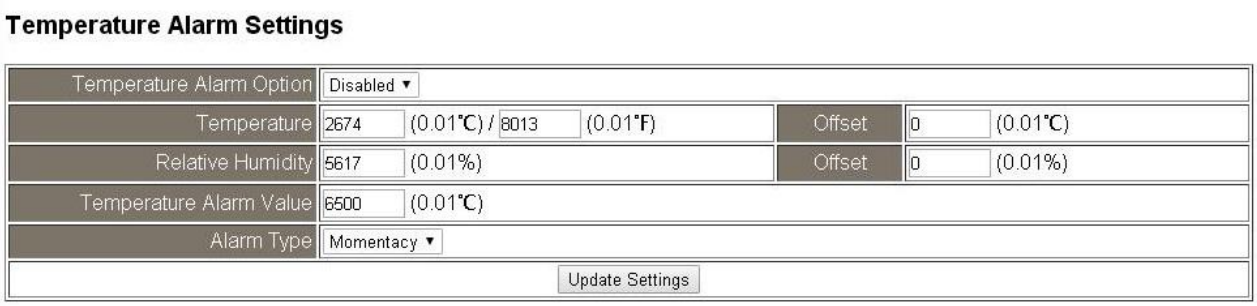
3.6.2. PIR Settings

Configuration	Hardware	
	Software Settings	Hardware Settings
Luminance Value	Disabled (Lum)	Disable
PIR Output On Time	6 (Seconds)	6 (Seconds)
Sensitivity value	0 (0 ~ 9)	Maximum
Active Delay Time (ms)	0 (0000 ~ 3000 ms)	
Buzzer Operation	Inactive	
Sensor Output	Disabled	DO ON OFF
Update Settings		

The following table provides an overview of the parameters contained in the ***PIR Settings*** section:

Item	Description
Configuration	The configuration for PIR-230 module can refer to either hardware or software, and is optional via the DIP[2] switch on the rear side of the module. This field is read only; it shows the current setting on the module.
Luminance Value	The luminance value for enable the PIR sensor. The PIR sensor will be enabled when the luminance value is lower than the
PIR Output ON Time	The delay time of sensor status after PIR sensor is triggered. (Refer to Section 5.15. $AALC8C0 for details)
Sensitivity Value	The lower value denotes a higher sensitivity. (0~9)
Active Delay Time (ms)	The delay time of PIR active after power on. (0~3000ms)
Buzzer Operation	Buzzer is active or inactive when the PIR sensor is triggered.
Sensor Output	Digital output is enabled or disabled when the PIR sensor is triggered.

3.6.3. Temperature Alarm Settings



The following table provides an overview of the parameters contained in the **Temperature Alarm** section:

Item		Description	Attribute
Temperature Alarm Option		Enable/Disable the temperature high alarm function	R/W
Temperature		The value of Centigrade(unit 0.01℃) and Fahrenheit (unit 0.01℉)	R
	Offset	The temperature offset value sets all the measured temperature plus a constant offset to close to room temperature.	R/W
Relative Humidity		The value of relative humidity (unit 0.01%)	R
	Offset	The humidity offset value sets all the measured humidity plus a constant offset to close to room humidity.	R/W
Temperature Alarm Value		The value of temperature high alarm (unit 0.01℃)	R/W
Alarm Type		Momentary alarm / Latch alarm	R/W

3.7 Filter

Clicking the **Filter** tab to go to the **Filter Settings** page where you can configure the IP Filter for the PIR-230 module, which will be described in more detail below.

3.7.1. Filter Settings

The ***Filter Settings*** page is used to query or edit the IP Filter List for the PIR-230 module. The IP filter list restricts the access of incoming packets based on the IP header. If one or more IP addresses are saved to the IP Filter table, only Clients whose IP address is specified in the IP Filter List will be able to access the PIR-230 module.

Filter Settings:

Available IP List	IP Address
IP1:	10.0.8.20
IP2:	0.0.0.0
IP3:	0.0.0.0
IP4:	0.0.0.0
IP5:	0.0.0.0

Add . . . To The List
Delete IP#
Delete ALL
Save to Flash
submit

The following table provides an overview of the parameters contained in the IP Address Configuration section:

Item	Description
Add “IP” to the List	This parameter is used to add an IP address to the IP filter List.
Delete IP # “number”	This parameter is used to delete IP# address from the IP filter List.
Delete All	This parameter is used to delete all IP address current contained in the IP filter List.
Save to Flash	This parameter is used to save the updated IP filter List to the flash memory. Check the checkbox before clicking the Submit button of you wish to store the most recent list.
Submit	Click this button to save the revised settings to PIR-230 module.

3.8. Monitor

After clicking the ***Monitor*** tab, the Current Connection Status page will be displayed showing detailed information regarding the current status of the serial port connection settings for the PIR-230 module.

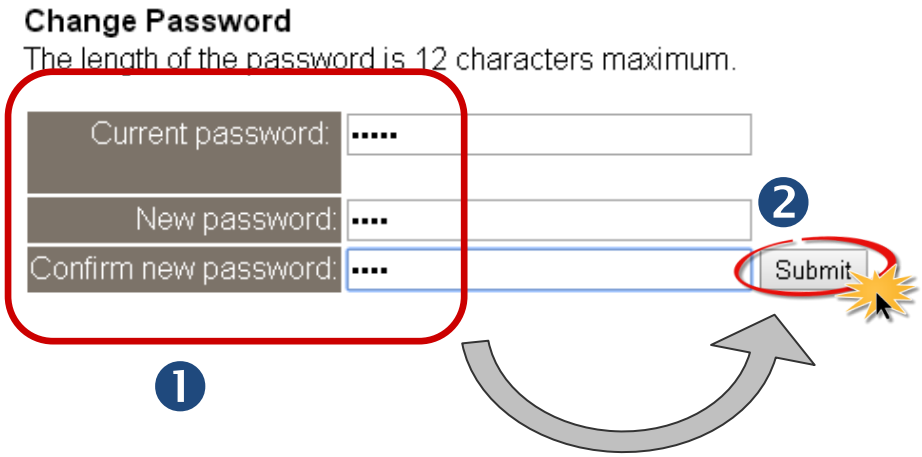
Current Connection Status:

Server Mode	Server
Connected IP1:	0.0.0.0
IP2:	0.0.0.0
IP3:	0.0.0.0
IP4:	0.0.0.0
IP5:	0.0.0.0
IP6:	0.0.0.0
Available Connections:	32

3.9. Change Password

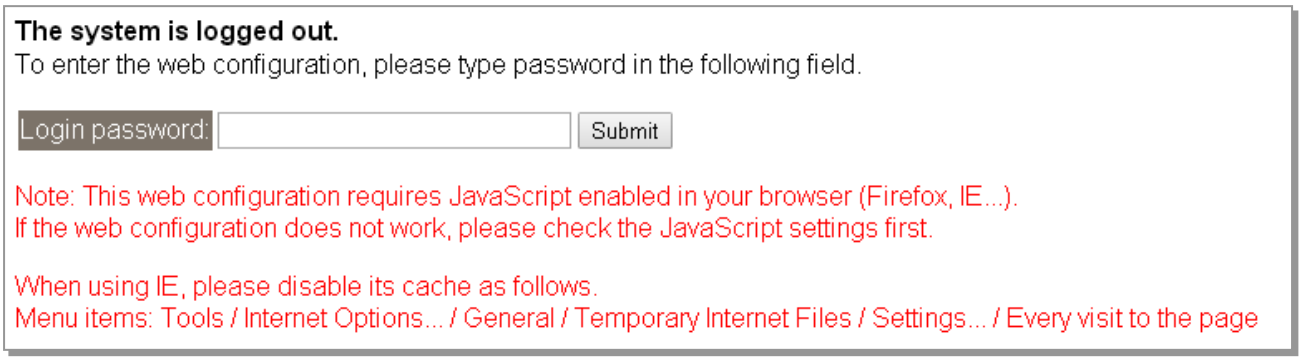
To change the p default password:

Step 1: Go to the ***Change Password*** page by clicking the ***Change Password*** tab.
Step 2: Enter the old password in the textbox next to **"Current password"**. (Default: **Admin**)
Step 3: Enter a new password in the textbox next to **"New password"**.
Step 4: Re-enter the new password in textbox next to **"Confirm new password"**.
Step 5: Click the "**Submit**" button to update the password.



3.10. Logout

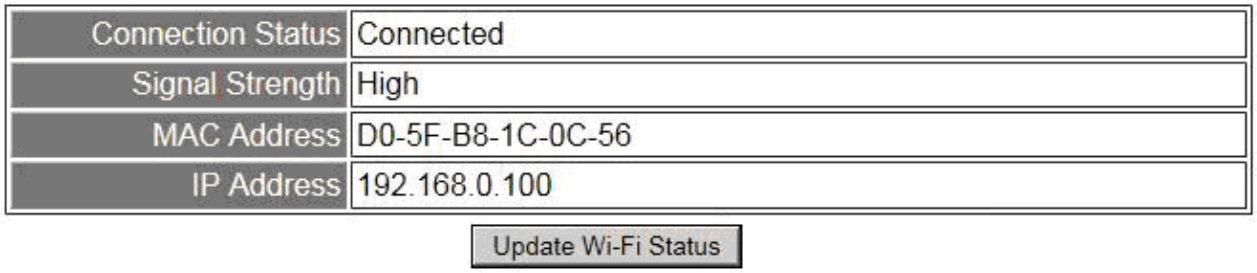
Clicking the ***Logout*** tab will immediately log you out from the system and return you to the login page.



3.11. Wi-Fi (for PIR-230-WF only)

For PIR-230-WF module, the Wi-Fi related parameters can be set via the Wi-Fi page. This page including Wi-Fi Status and Wi-Fi Settings, each of which will be described in more detail below.

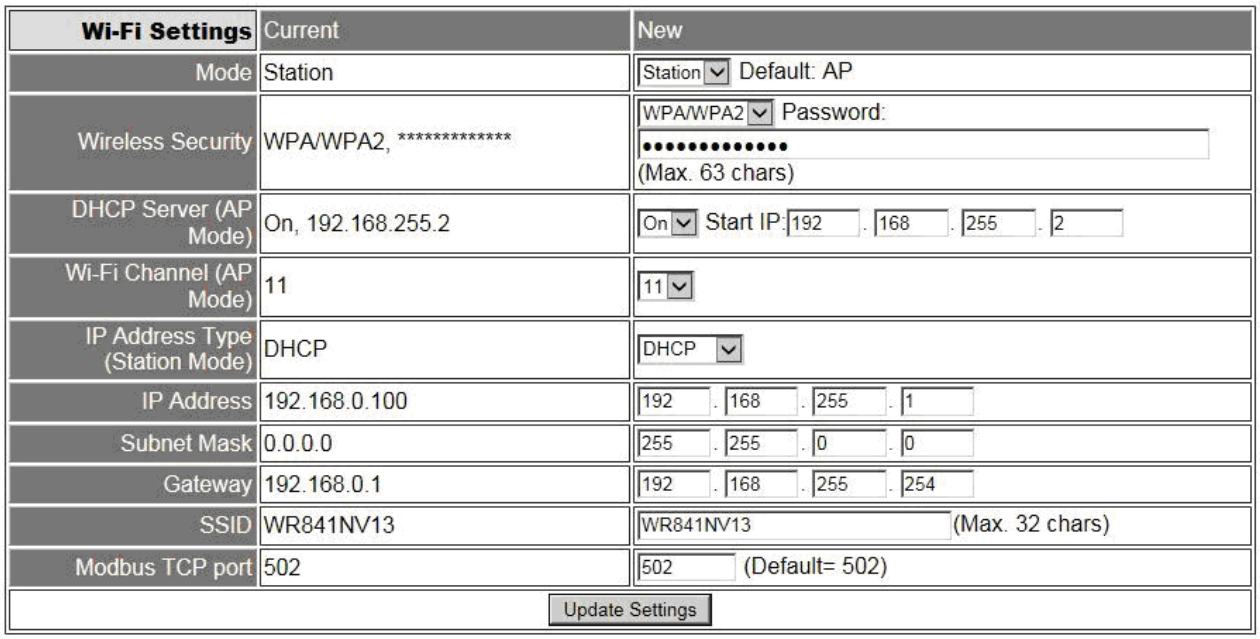
3.11.1. Wi-Fi Status



The following table provides an overview of the parameters contained in the Wi-Fi Status section:

Item	Description
Connection Status	The Wi-Fi connection status of the PIR-230-WF device.
Signal Strength	The Wi-Fi signal strength of the PIR-230-WF device in station mode. It can be High, Medium, Low, or Not Connected.
MAC Address	The MAC address of the Wi-Fi interface of the PIR-230-WF device.
IP Address	The IP address of the Wi-Fi interface of the PIR-230-WF device.
Update Wi-Fi Status	Click this button to update the Wi-Fi status of the PIR-230-WF device.

3.11.2. Wi-Fi Settings



The column of Current shows the current Wi-Fi settings. You can change the settings by changing the column of New. The following table provides an overview of the parameters contained in the Wi-Fi Settings section:

Item	Description
Mode	This parameter is used to specify the Wi-Fi mode of the PIR-230-WF device. It can be station or AP. For AP mode, only one device can be connected.
Wireless Security	This parameter is used to specify which security protocol is used to secure wireless computer network. It can be open, WEP, or WPA/WPA2. It is recommended to use WPA/WPA2 if possible.
DHCP Server (AP Mode)	This parameter is used to specify whether to turn on the DHCP server function. It is only available to the AP mode.
Wi-Fi Channel (AP Mode)	This parameter is used to specify which channel is used for Wi-Fi transmission. It can be 1 to 11. It is only available to the AP mode.
IP Address Type (Station Mode)	This parameter is only available to the station mode and it can be Static IP or DHCP. If DHCP is supported by the AP you would like to connect, then DHCP should be selected. Otherwise, select Static IP and the following three parameters IP Address, Subnet Mask and Gateway should be set, too.

IP Address	Each PIR-230-WF device connected to the Wi-Fi network must have its own unique IP address. This parameter is used to assign a specific IP address.
Subnet Mask	This parameter is used to assign the subnet mask for the PIR-230-WF device. The subnet mask indicates which portion of the IP address is used to identify the local network or subnet.
Gateway	This parameter is used to assign the IP address of the gateway to be used by the PIR-230-WF device. A gateway (or router) is a device that is used to connect an individual network to one or more additional networks.
SSID	This parameter is used to specify the Service Set Identifier. For station mode, specify the SSID of the AP you would like to connect. For AP mode, the SSID will be used by the device to be connected.
Modbus TCP Port	This parameter is used to set the local port of the Wi-Fi interface to be used by the Modbus slave device. The default value is 502.
Update Settings	Click this button to save the revised settings to the PIR-230-WF device.

The following table provides an overview of the factory default Wi-Fi settings:

Factory Default Wi-Fi Settings	
Mode	AP
Wireless Security	WPA/WPA2, "00000000"
DHCP Server (AP Mode)	DHCP Server on, start IP: 192.168.255.2
Wi-Fi Channel (AP Mode)	11
IP Address	192.168.255.1
Gateway Address	192.168.255.254
Subnet Mask	255.255.0.0
SSID	PIR-230-WF
Modbus TCP Port	502

4. Configuration via Wi-Fi

The factory default settings for Wi-Fi communication of the PIR-230-WF are as follows.

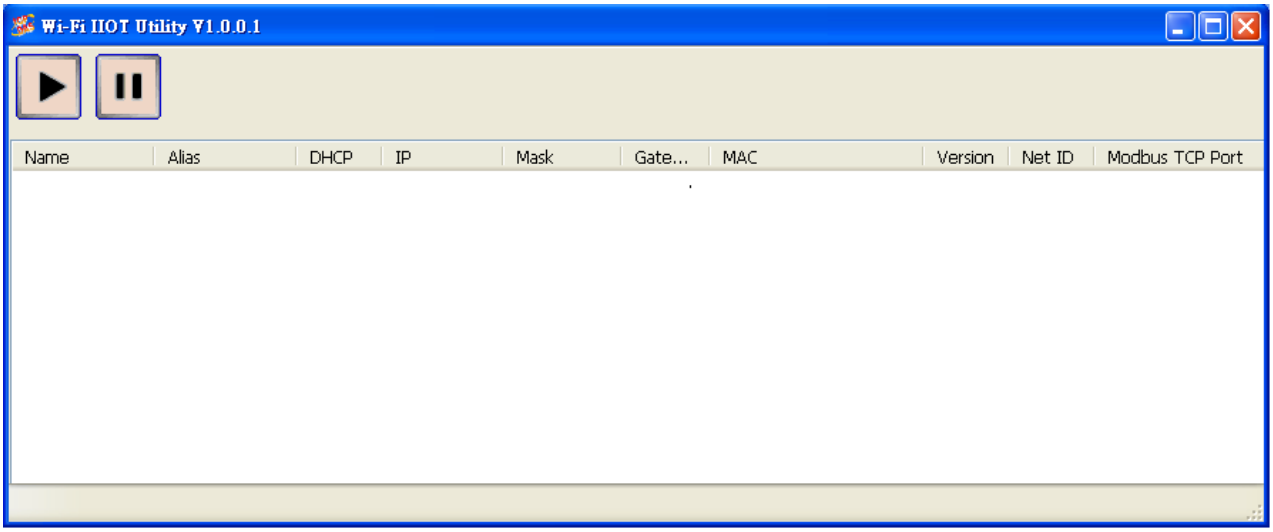
- Mode: AP
- Wireless Security: WPA/WPA2, "00000000"
- DHCP Server (AP Mode): DHCP Server on, start IP: 192.168.255.2
- Wi-Fi Channel (AP Mode): 11
- IP Address: 192.168.255.1
- Gateway Address: 192.168.255.254
- Subnet Mask: 255.255.0.0
- SSID: PIR-230-WF
- Modbus TCP Port: 502

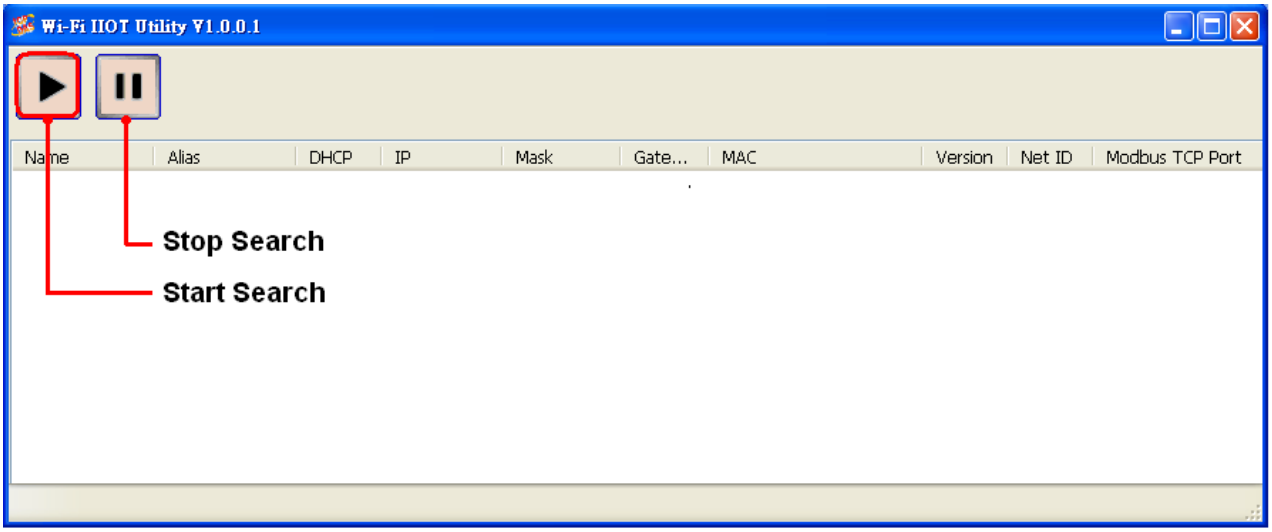
The Wi-Fi IIOT Utility is provided to configure and test the PIR-230-WF module through the Wi-Fi interface.

4.1. Building the Wi-Fi Connection

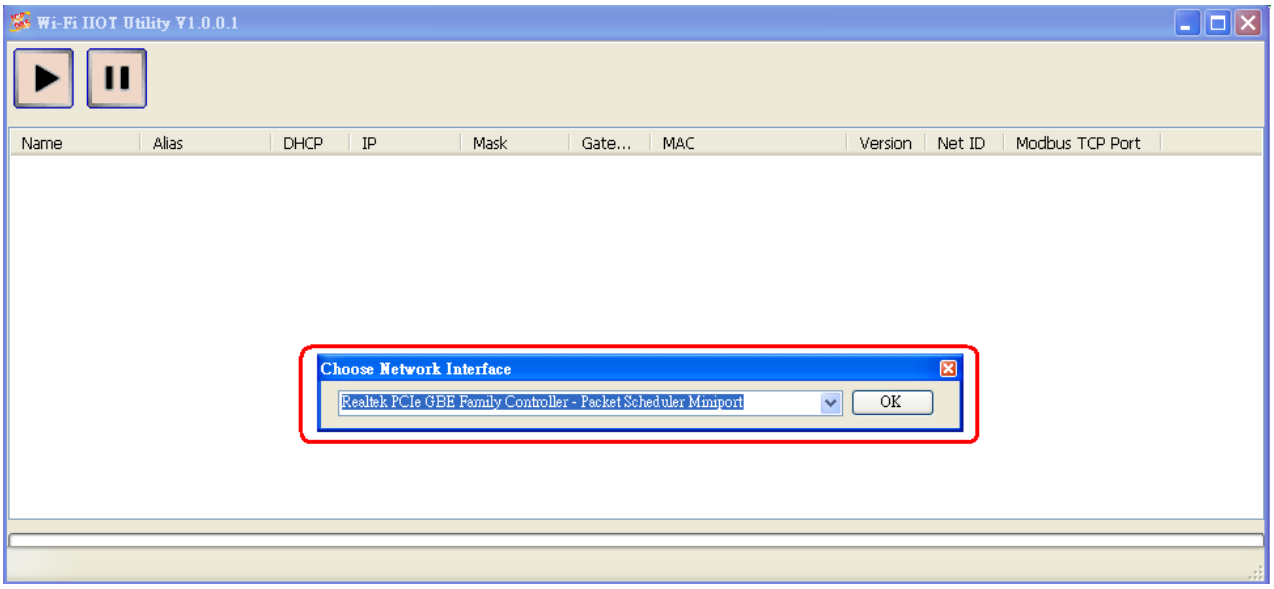
1. Install Wi-Fi IIOT Utility
   The installation file location of the Wi-Fi IIOT Utility is at:
   http://ftp.icpdas.com/pub/cd/iiot/utility/

2. Search and Find the Module
   Click on the search button to find the modules via the Wi-Fi interface.

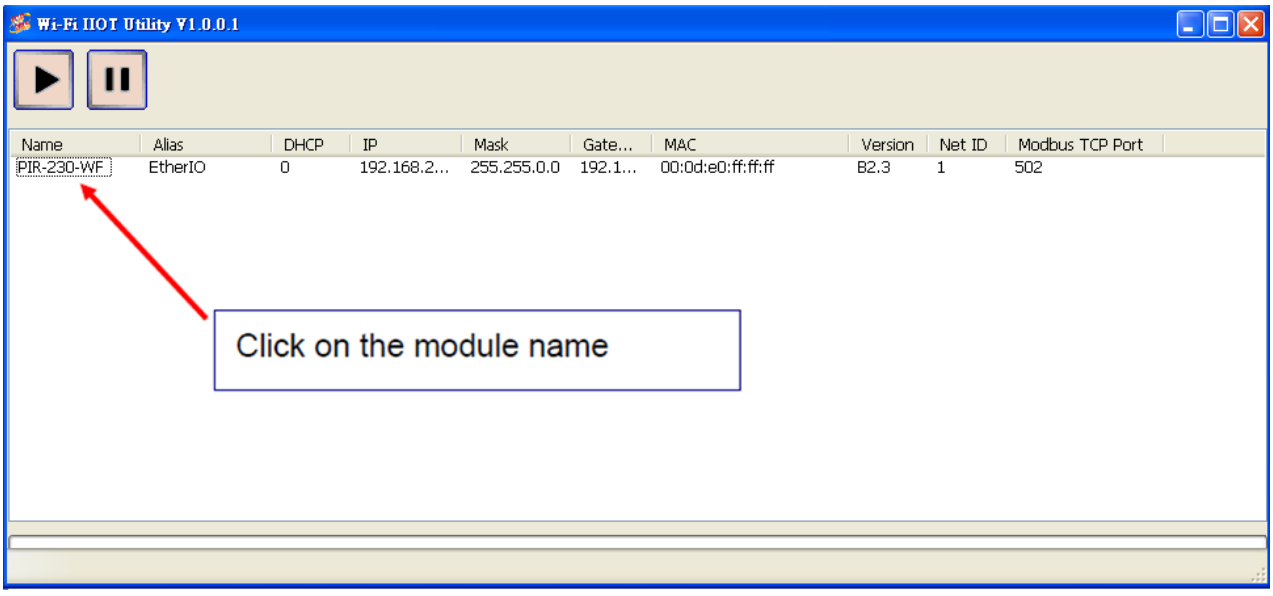




3. Select the Wi-Fi network interface and click on the **OK** button.

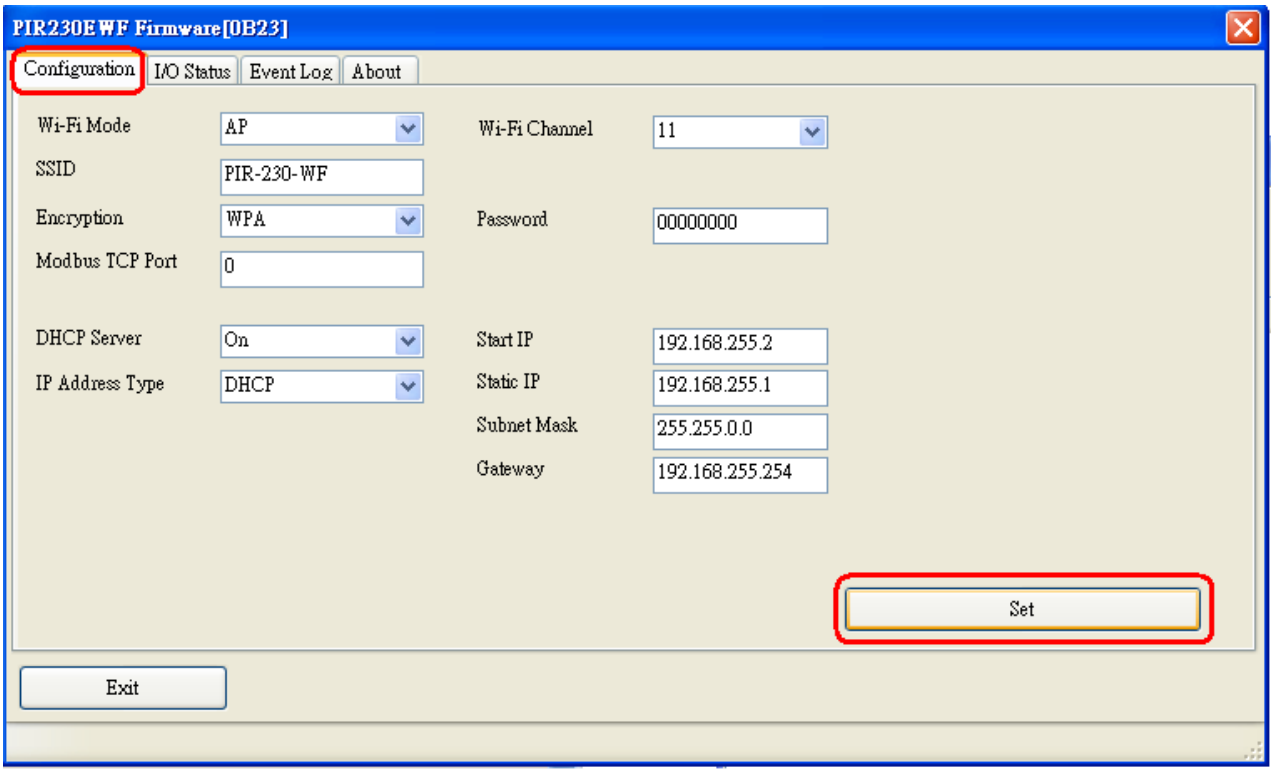


4. When the module is found, click on the module name to enter the configuration form.



4.2. Configuring the Wi-Fi Settings

In the Configuration form, you can change the Wi-Fi related settings. Click on the Set Module Configurations button to save the changes to the module.

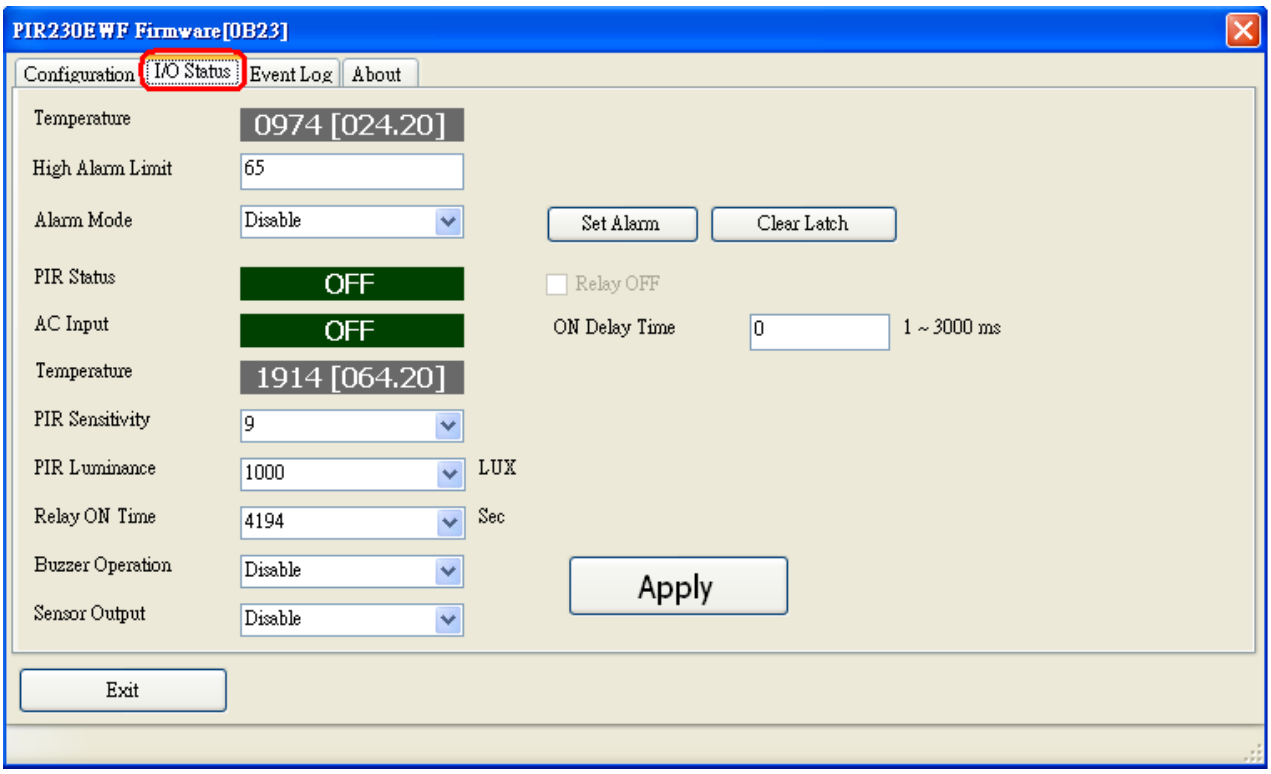


The followings show the detailed description of each setting.

Item	Description
WiFi Mode	This parameter is used to specify the Wi-Fi mode of the SL device. It can be Station or AP. For AP mode, only one device can be connected.
Encryption	This parameter is used to specify which security protocol is used to secure wireless computer network. It can be open, WEP, or WPA. It is recommended to use WPA if possible.
DHCP Server	This parameter is used to specify whether to turn on the DHCP server function. It is only available to the AP mode.
WiFi Channel	This parameter is used to specify which channel is used for Wi-Fi transmission. It can be 1 to 11. It is only available to the AP mode.
IP Address Type	This parameter is only available to the station mode and it can be Static or DHCP. If DHCP is supported by the AP you would like to connect, then DHCP should be selected. Otherwise, select Static and the following three parameters Static IP, Subnet Mask and Gateway should be set, too.
Static IP	Each SL device connected to the Wi-Fi network must have its own unique IP address. This parameter is used to assign a specific IP address.

Subnet Mask	This parameter is used to assign the subnet mask for the SL device. The subnet mask indicates which portion of the IP address is used to identify the local network or subnet.
Gateway	This parameter is used to assign the IP address of the gateway to be used by the SL device. A gateway (or router) is a device that is used to connect an individual network to one or more additional networks.
SSID	This parameter is used to specify the Service Set Identifier. For station mode, specify the SSID of the AP you would like to connect. For AP mode, the SSID will be used by the device to be connected.
Modbus TCP Port	This parameter is used to set the local port of the Wi-Fi interface to be used by the Modbus slave device. The default value is 502.

In the I/O Status form, you can configure the PIR sensor settings, temperature alarm settings and relay output.



The followings show the detailed description of each setting.

Item	Description
Temperature	This parameter is sensor readings temperature
High Alarm Limit	Sets the High alarm limit conditions for Temperature (unit 0.01℃)

Alarm Mode	- Disabled: Disables alarm function. - Momentary: If a measurement value higher than the High Alarm Limit or lower than the Low Alarm Limit, the alarm occurs until the measurement value is within a range from Low Alarm Limit to High Alarm Limit. - Latched: If a measurement value higher than the High Alarm Limit or lower than the Low Alarm Limit, the alarm occurs.
Relative Humidity	This parameter is sensor readings relative humidity
PIR Sensitivity	The lower value denotes a higher sensitivity. (0~9)
PIR Luminance	The luminance value for enable the PIR sensor. The PIR sensor will be enabled when the luminance value is lower than the
Relay ON Time	The delay time of sensor status after PIR sensor is triggered. (Refer to Section 5.15. $AALC8C0 for details)
Buzzer Operation	Buzzer is active or inactive when the PIR sensor is triggered.
Sensor Output	Digital output is enabled or disabled when the PIR sensor is triggered.

5. DCON Protocol

5.1. Overview

All communication with the PIR-230 module consists of commands generated by the Host and responses transmitted by the PIR-230 module. Each module has a unique ID number that is used for addressing purposes and is stored in non-volatile memory. The module ID number is set to 01 by default and can be changed by sending a user command. All commands to the modules contain the ID number as the address, meaning that only the addressed module will respond.

Command Format:

Delimiter Character	Module Address	Command	Checksum	CR

Response Format:

Delimiter Character	Module Address	Data	Checksum	CR

CR = End of command character, carriage return (0x0D), used to end a frame.

Note 1: All characters should be in upper case.
Note 2: The DCON Utility Pro can be downloaded from:
http://ftp.icpdas.com/pub/cd/iiot/utility/

DCON Command Sets

General Command Sets			
Command	**Response**	**Description**	**Section**
%AANNTTCCFF	!AA	Sets the Configuration of the Module	5.2
$AA2	!AANNTTCCFF	Reads the Configuration of the Module	5.7
$AA5	!AAS	Reads the Reset Status of the Module	5.8
$AAF	!AA(Data)	Reads the Firmware Version of the Module	5.9
$AAM	!AA(Data)	Reads the Name of the Module	5.20
$AAP	!AASC	Reads the Communication Protocol currently used by the module	5.21
$AAPN	!AA	Sets the Communication Protocol to be used by the module	5.22
~AARD	!AA(Data)	Reads the current Response Delay Time	5.23
~AARDVV	!AA	Sets the Response Delay Time for the Module	5.24

PIR Input/Relay Output Status Command Sets			
Command	**Response**	**Description**	**Section**
@AADI	!(Data)	Reads the current status of the digital input /output	5.25
@AADONN	>	Sets the digital output	5.26
~AAD	!AAVV	Reads the current active state of the digital output	5.27
~AADVV	!AA	Sets the active state of the digital output	5.28

PIR Argument Command Sets			
Command	**Response**	**Description**	**Section**
$AALC2C0NNNN	!AA	Sets the PIR output delay time after power on	5.10
$AALC3	!AANNNN	Reads the PIR output delay time after power on	5.11
$AALC5C0NN	!AA	Sets the value of the luminance level for the PIR sensor	5.12
$AALC6C0	!AANN	Reads the current value of the luminance level	5.13

		for the PIR sensor	
$AALC7C0NN	!AA	Sets the PIR output ON time when the PIR sensor is triggered	5.14
$AALC8C0	!AANN	Reads the current PIR output ON time when the PIR Sensor is triggered	5.15
$AALC9C0NN	!AA	Sets the PIR output configuration	5.16
$AALCAC0	!AANN	Reads the current PIR output configuration	5.17
$AALCBC0NN	!AA	Sets the sensitivity value for the PIR sensor	5.18
$AALCCC0	!AANN	Read s the sensitivity value for PIR sensor	5.19

High Alarm Command Sets			
Command	**Response**	**Description**	**Section**
#AA	>(Data)	Read the value of temperature and humidity	5.3
#AA0	>(Data)	Read the value of Centigrade	5.4
#AA1	>(Data)	Read the value of Fahrenheit	5.5
#AA2	>(Data)	Read the value of relative humidity	5.6
@AACHC0	!AA	Clears the status of the high alarm	5.29
@AADA	!AA	Disables the high alarm function	5.30
@AAEAT	!AA	Enables the high alarm function	5.31
@AAHI(Data)	!AA	Sets the high alarm condition value	5.32
@AAHO	!AA(Data)	Reads the value of humidity offset	5.33
@AAHO(Data)	!AA	Sets the value of humidity offset	5.34
@AARH	!AA(Data)	Reads the current value of the high alarm	5.35
@AARAO	!AAHH00	Reads the currently activated alarm	5.36
@AATO	!AA(Data)	Reads the value of temperature offset	5.37
@AATO(Data)	!AA	Sets the value of temperature offset	5.38

5.2. %AANNTTCCFF

Description:

This command is used to set the configuration of a specified module.

Syntax:

%AANNTTCCFF[CHKSUM](CR)

%	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
NN	The new address of the module in hexadecimal format (00 to FF)
TT	The Type Code, which should be set to 40 for DIO modules
CC	The new Baud Rate and data format settings. See the following tables for detailed information.
FF	The new Checksum setting. See the following tables for detailed information.

Note: The DIP switch #4(Init) must be set to ON position before sending this command. It needs be set back to OFF after finishing the configuration settings. See Section 1.5 for more details.

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid. If an attempt is made to change the Baud Rate or Checksum setting without putting the DIP switch position #4 in the ON position, the module will return a response indicating that the command was invalid. See Section 1.5 for more details.
AA	The address of the responding module in hexadecimal format (00 to FF)

Note: There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Configuration Data Format

Baud Rate Settings (CC)

7	6	5	4	3	2	1	0
Parity		Baud Rate Code					

Baud Rate Code (Bits 0 to 5)

Code	03	04	05	06	07	08	09	0A
Baud Rate	1200	2400	4800	9600	19200	38400	57600	115200

Parity (Bits 6 and 7)

Code	00	01	10	11
Parity	N,8,1	N,8,2	E,8,1	O,8,1

Data Format Settings (FF)

7	6	5	4	3	2	1	0
Reserved	CS	Reserved					

Key	Description
CS	Checksum Settings 0: Disabled 1: Enabled

Note: All Reserved bits should be zero.

Examples:

Command: %0102400600 Response: !02

Changes the address of module 01 to 02. The module returns a response indicating that the command was valid and includes the new address of the module.

Command: %0101200A00 Response: ?01

Attempts to change the Baud Rate of module 01 to 115200 bps, but the module returns a response indicating that the command was invalid. It maybe because that the DIP switch #4 is not in the ON position. See Section 1.5 for more details.

Command: %0101200A00 Response: !01

Changes the Baud Rate of module 01 to 115200 bps with putting the DIP switch #4 in the ON position. The module returns a response indicating that the command was valid.

Command: $012 Response: !01400600

Reads the configuration of module 01 and returns a response indicating that the command was valid. The response showing that the Type Code is 40, the Baud Rate is 9600 bps, the Checksum is Disabled.

Related Commands:

Section 5.7. $AA2

Related Topics:

Section 2.5. Hardware Configuration

Notes:

Changes to the address settings take effect immediately after a valid command is received. Changes to the Baud Rate and Checksum settings take effect on the next power-on reset.

5.3. #AA

Description:

This command is used to read the value of temperature and relative humidity for a special module.

Syntax:

#AA[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)

Response:

Valid Command: **>(Data) (CR)**

Invalid Command: **?AA[CHKSUM](CR)**

>	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The value of the Centigrade, Fahrenheit and relative humidity

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #01 Response: >+030.00+086.00+050.00

Reads the temperature and humidity of module 01 and returns a response indicating that the command was valid, and temperature is +30℃/+86℉ and the relative humidity is +50%.

Related Commands:

Section 5.4. #AA0, Section 5.5. #AA1, Section 5.6. #AA2

5.4. #AA0

Description:

This command is used to read the Centigrade of a specified module.

Syntax:

#AA0[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
0	The command to read the value of Centigrade

Response:

Valid Command: **#AA0[CHKSUM](CR)**

Invalid Command: **>(Data)[CHKSUM](CR)**

>	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The value of the Centigrade

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #010 Response: >+030.00

Reads the temperature of module 01 and returns a response indicating that the command was valid, and the temperature is +30℃.

Related Commands:

Section 5.3. #AA, Section 5.5. #AA1, Section 5.6. #AA2

5.5. #AA1

Description:

This command is used to read the Fahrenheit of a specified module.

Syntax:

#AA1[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
1	The command to read the Fahrenheit.

Response:

Valid Command: **>(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

>	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The value of the Fahrenheit

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #011 Response: >+086.00

Reads the temperature of module 01 and returns a response indicating that the command was valid, and the temperature is +86°F.

Related Commands:

Section 5.3. #AA, Section 5.4. #AA0, Section 5.6. #AA2

5.6. #AA2

Description:

This command is used to read the relative humidity of a specified module.

Syntax:

#AA2[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
2	The command to read the relative humidity.

Response:

Valid Command: **>(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

>	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The value of the relative humidity

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #012 Response: >+050.00

Reads the relative humidity of module 01 and returns a response indicating that the command was valid, and temperature is +50%.

Related Commands:

Section 5.3. #AA, Section 5.4. #AA0, Section 5.5. #AA1

5.7. $AA2

Description:

This command is used to read the current configuration of a specified module.

Syntax:

$AA2[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
2	The command to read the module configuration

Response:

Valid Command: **!AATTCCFF[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
TT	The Type Code for the module, which should be 40 for DIO modules
CC	The Baud Rate for the module. See Section 2.1 for details.
FF	The Checksum status. See Section 2.1 for details.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: %0101000600 Response: !01

Changes the Baud Rate of module 01 to 9600 bps, the DIP switch #4 is in the ON position and disable the checksum function. The module returns a response indicating that the command was valid.

Command: $012 Response: !01000600

Reads the configuration of module 01 and returns a response indicating that the command was valid, and showing that the Baud Rate is 9600 bps and the Checksum is Disabled.

Related Commands:

Section 5.2. %AANNTTCCFF

Related Topics:

Section 2.5. Hardware Configuration

5.8. $AA5

Description:

This command is used to read the current reset status for a specified module.

Syntax:

$AA5[CHKSUM](CR)

$ Delimiter character

AA The address of the module to be configured in hexadecimal format (00 to FF)

5 The command to read the reset status of the module

Response:

Valid Command: **!AAS[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

AA The address of the responding module in hexadecimal format (00 to FF)

S The reset status of the module:

0: This is **NOT** the first time the command has been sent since the module was powered on, which denotes that there has been no module reset since the last $AA5 command was sent.

1: This is the first time the $AA5 command has been sent since the module was powered on.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $015 Response: !011

Reads the reset status for module 01 and returns a response indicating that the command was valid, and that it is the first time the $AA5 command has been sent since the module was powered on.

Command: $015 Response: !010

Reads the reset status for module 01 and returns a response indicating that the command was valid, and that there has been no module reset since the last $AA5 command was sent.

Related Commands:

None

5.9. $AAF

Description:

This command is used to read the current firmware version of a specified module.

Syntax:

$AAF[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
F	The command to read the current firmware version

Response:

Valid Command: **!AA(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A string indicating the current firmware version of the module

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01F Response: !01A1.0

Reads the current firmware version of module 01, and returns a response indicating that the command was valid, and that the firmware is version A1.0.

Related Commands:

None

5.10. $AALC2C0NNNN

Description:

This command is used to set the PIR output delay time on a specified module after power on.

Syntax:

$AALC3C0NNNN[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LC2	The command to set the PIR output delay time after power on
C	The command to set the PIR output channel
0	Specifies the PIR output channel to be set, zero based. Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.
NNNN	A four-digit hexadecimal value representing the PIR output delay time in milliseconds. The maximum delay time is 0x0BB8 (3000 milliseconds).

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LC2C003E8 Response: !01

Sets the PIR output delay time of module 01 to 0x03E8 (1000 milliseconds) after power on and the module returns a response indicating that the command was valid. The PIR output will be active for 1000 milliseconds after powered on.

Command: $01LC3 Response: !010BB8

Reads the PIR output delay time of module 01 after power on and returns a response indicating that the command was valid, with a value of 0BB8 meaning that the PIR output delay time is 3000 milliseconds. The PIR output will be active for 3000 milliseconds after power on.

Command: $01LC2C00BB9 Response: ?01

Attempts to set the PIR output delay time of module 01 to 0x0BB9 (3001 milliseconds) after power on, but the module returns a response indicating that the command was invalid because the value for the PIR output delay time was not within the valid range.

Related Commands:

Section 5.11 $AALC3

5.11. $AALC3

Description:

This command is used to read the PIR output delay time on a specified module after power on.

Syntax:

$AALC3C0NNNN[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LC3	The command to read the PIR output delay time after power on

Response:

Valid Command: **!AANNNN[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
NNNN	A four-digit hexadecimal value representing the PIR output delay time in milliseconds after power on. The maximum delay time is 0x0BB8 (3000 milliseconds).

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LC2C003E8 Response: !01

Sets the PIR output delay time of module 01 to 0x03E8 (1000 milliseconds) after power on and the module returns a response indicating that the command was valid. The PIR output will be active for 1000 milliseconds after powered on.

Command: $01LC3 Response: !0103E8

Reads the PIR output delay time of module 01 after power on and returns a response indicating that the command was valid, with a value of 0BB8 meaning that the PIR output delay time is 1000 milliseconds. The PIR output will be active for 1000 milliseconds after power on.

Related Commands:

Section 5.10 $AALC2C0NNNN

5.12. $AALC5C0NN

Description:

This command is used to set the Luminance limit value for the PIR sensor on a specific channel of a specified module.

Syntax:

$AALC5C0N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LC5	The command to set the Luminance limit value for the PIR Sensor
C	The command to set the PIR output channel
0	Specifies the PIR output channel to be set, zero based. Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.
NN	The command to set the Luminance Value for the PIR sensor, where: 00: Disabled 01: 100 Lux 02: 200 Lux 03: 500 Lux 04: 1000 Lux The valid range is 0 to 4. This value will be stored inside.

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LC5C00 Response: !01

Sets the Luminance value for the PIR sensor on channel 0 of module 01 to a value of 0 meaning that the Luminance value is disabled, and the module returns a response indicating that the command was valid.

Command: $01LC5C05 Response: ?01

Attempts to set the Luminance value for the PIR sensor on channel 0 of module 01 to a value of 5, but the module returns a response indicating that the command was invalid because the value is not within the valid range.

Command: $01LC6C0 Response: !010

Reads the Luminance value for the PIR sensor on channel 0 of module 01 and returns a response indicating that the command was valid, with a value of 1, meaning that the Luminance Value is disabled.

Related Commands:

Section 5.13 $AALC6C0

5.13. $AALC6C0

Description:

This command is used to read the Luminance value for the PIR sensor on a specific channel of a specified module.

Syntax:

$AALC6C0[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LC6	The command to read the Luminance value for the PIR sensor
C	The command to read the PIR channel
0	Specifies the PIR channel to be read, zero based. Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.

Response:

Valid Command: **!AAN[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

Ignored Command: **!AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
NN	The Luminance value for the PIR sensor, which is stored inside. The valid range is 0 to 4, where: 0: Disabled 1: 100 Lux 2: 200 Lux 3: 500 Lux 4: 1000 Lux

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LC5C00 Response: !01

Sets the Luminance value for the PIR sensor on channel 0 of module 01 to a value of 0 meaning that the Luminance value is disabled, and the module returns a response indicating that the command was valid.

Command: $01LC5C05 Response: ?01

Attempts to set the Luminance value for the PIR sensor on channel 0 of module 01 to a value of 5, but the module returns a response indicating that the command was invalid because the value is not within the valid range.

Command: $01LC6C0 Response: !010

Reads the Luminance value for the PIR sensor on channel 0 of module 01 and returns a response indicating that the command was valid, with a value of 1, meaning that the Luminance Value is disabled.

Related Commands:

Section 5.12 $AALC5C0N

5.14. $AALC7C0NN

Description:

This command is used to set the PIR output ON time on a specific module when PIR sensor is triggered.

Syntax:

$AALC70NN[CHKSUM](CR)

$ Delimiter character

AA The address of the module to be configured in hexadecimal format (00 to FF)

LC7 The command to set the PIR output ON time when the PIR sensor is triggered

C The command to set the PIR output channel

0 Specifies the PIR output channel to be set, zero based.
Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.

NN The command to set the PIR output ON time for when the PIR sensor is triggered in hexadecimal format. This value will be stored inside, and the valid range is 00 to 0F.

NN	Seconds	NN	Seconds	NN	Seconds	NN	Seconds
00	2	01	4	02	6	03	8
04	16	05	33	06	49	07	66
08	131	09	262	0A	393	0B	524
0C	1049	0D	2097	0E	3146	0F	4194

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

AA The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LC7C000 Response: !01

Sets the PIR output ON time value for channel 0 of module 01 to 00, meaning that the output ON time will be 2 seconds, and the module returns a response indicating that the command was valid.

Command: $01LC8C0 Response: !0100

Reads the PIR output ON time value for channel 0 of module 01, and the module returns a response indicating that the command was valid, with a value of 00 meaning that the output ON time will be 2 seconds.

Command: $01LC7C100 Response: ?01

Attempts to set the PIR output ON time value for channel 1 of module 01 to 00, but the module returns a response indicating that the command was invalid because channel 1 does not exist on the PIR-230 module.

Related Commands:

Section 5.15 $AALC8C0

5.15. $AALC8C0

Description:

This command is used to read the current PIR output ON time when the PIR sensor is triggered on a specific module.

Syntax:

$AALC1[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LC8	The command to read the PIR output ON time when the PIR sensor is triggered
C	The command to read the PIR output channel
0	Specifies the PIR output channel to be read, zero based. Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.

Response:

Valid Command: **!AANN[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
NN	The PIR output ON time value when the PIR sensor is triggered in hexadecimal format. This value will be stored inside, and the valid range is 00 to 0F.

NN	Seconds	NN	Seconds	NN	Seconds	NN	Seconds
00	2	01	4	02	6	03	8
04	16	05	33	06	49	07	66
08	131	09	262	0A	393	0B	524
0C	1049	0D	2097	0E	3146	0F	4194

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LC7C000 Response: !01

Sets the PIR output ON time value for channel 0 of module 01 to 00, meaning that the output ON time will be 2 seconds, and the module returns a response indicating that the command was valid.

Command: $01LC8C0 Response: !0100

Reads the PIR output ON time value for channel 0 of module 01, and the module returns a response indicating that the command was valid, with a value of 00 meaning that the output ON time will be 2 seconds.

Command: $01LC7C100 Response: ?01

Attempts to set the PIR output ON time value for channel 1 of module 01 to 00, but the module returns a response indicating that the command was invalid because channel 1 does not exist on the PIR-230-E module.

Related Commands:

Section 5.14 $AALC7C0NN

5.16. $AALC9C0NN

Description:

This command is used to set the PIR output configuration for a specific channel on a specified module.

Syntax:

$AALC9C0N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LC9	The command to set the output configuration
C	The command to set the PIR output channel
0	Specifies the PIR output channel to be set, zero based. Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.
NN	The command to set the PIR output configuration, where: Bit 0: Enable/Disable the Buzzer be activated Bit 1: Enable/Disable the relay output be activated This value will be stored inside.

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LC9C003 Response: !01

Sets the PIR output configuration of module 01 and the module returns a response indicating that the command was valid and the relay output and buzzer will be active when the PIR sensor is triggered.

Command: $01LCAC0 Response: !0103

Reads the PIR output configuration of module 01 and the module returns a response indicating that the command was valid and the relay output and buzzer will be active when the PIR sensor is triggered.

Command: $01LC9C103 Response: ?01

Attempts to read the PIR output configuration for channel 1 of module 01, but the module returns a response indicating that the command was invalid because channel 1 does not exist on the PIR-230 module.

Related Commands:

Section 5.17 $AALCAC0

5.17. $AALCAC0

Description:

This command is used to read the PIR output configuration for a specific channel on a specified module.

Syntax:

$AALC9C0N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LCA	The command to read the PIR output configuration
C	The command to read the PIR output channel
0	Specifies the PIR output channel to be read, zero based. Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.

Response:

Valid Command: **!AANN[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
NN	The command to set the output configuration, where: Bit 0: Enable/Disable the Buzzer be activated Bit 1: Enable/Disable the relay output be activated This value will be stored inside.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LC9C003 Response: !01

Sets the PIR output configuration of module 01 and the module returns a response indicating that the command was valid and the relay output and buzzer will be active when the PIR sensor is triggered.

Command: $01LCAC0 Response: !0103

Reads the PIR output configuration of module 01 and the module returns a response indicating that the command was valid and the relay output and buzzer will be active when the PIR sensor is triggered.

Command: $01LC9C103 Response: ?01

Attempts to read the PIR output configuration for channel 1 of module 01, but the module returns a response indicating that the command was invalid because channel 1 does not exist on the PIR-230 module.

Related Commands:

Section 5.16 $AALC9C0NN

5.18. $AALCBC0NN

Description:

This command is used to set the sensitivity value for the PIR sensor on a specific channel of a specified module.

Syntax:

$AALC9C0N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LCB	The command to set the sensitivity value for the PIR sensor
C	The command to set the PIR output channel
0	Specifies the PIR output channel to be set, zero based. Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.
NN	The command to set the sensitivity value for the PIR sensor. The valid range is 00 to 09, where a lower value denotes a higher sensitivity, and this value will be stored inside.

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LCBC001 Response: !01

Sets the sensitivity value for the PIR sensor on channel 0 of module 01 to 1, and the module returns a response indicating that the command was valid.

Command: $01LCCC0 Response: !0101

Reads the sensitivity value for the PIR sensor on channel 0 of module 01 and the module returns a response indicating that the command was valid, with a value of 1.

Command: $01LCBC101 Response: ?01

Attempts to set the sensitivity value for the PIR sensor on channel 1 of module 01 to 1, but the module returns a response indicating that the command was invalid because channel 1 does not exist on the PIR-230 module.

Related Commands:

Section 5.19 $AALCCC0

5.19. $AALCCC0

Description:

This command is used to read the current sensitivity value for the PIR sensor on a specific channel of a specified module.

Syntax:

$AALC9C0N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
LCC	The command to read the sensitivity value for the PIR sensor
C	The command to read the PIR output channel
0	Specifies the PIR output channel to be read, zero based. Note that as there is only one PIR sensor channel on the PIR-230 module, the only valid value is 0.

Response:

Valid Command: **!AANN[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
NN	The sensitivity value for the PIR sensor, which is stored inside. The valid range is 00 to 09, where a lower value denotes a higher sensitivity.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01LCBC001 Response: !01

Sets the sensitivity value for the PIR sensor on channel 0 of module 01 to 1, and the module returns a response indicating that the command was valid.

Command: $01LCCC0 Response: !0101

Reads the sensitivity value for the PIR sensor on channel 0 of module 01 and the module returns a response indicating that the command was valid, with a value of 1.

Command: $01LCBC101 Response: ?01

Attempts to set the sensitivity value for the PIR sensor on channel 1 of module 01 to 1, but the module returns a response indicating that the command was invalid because channel 1 does not exist on the PIR-230 module.

Related Commands:

Section 5.18 $AALCBC0N

5.20. $AAM

Description:

This command is used to read the name of a specified module.

Syntax:

$AAM[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
M	The command to read the name of the module

Response:

Valid Command: **!AA(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A string indicating the name of the module

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $02M Response: !02PIR230E

Reads the name of module 02 and returns a response indicating that the command was valid, and that the name of the module is “PIR230E”.

Related Commands:

None

5.21. $AAP

Description:

This command is used to read the current communication protocol information configured for a specified module.

Syntax:

$AAP[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
P	The command to read the current communication protocol information

Response:

Valid Command: **!AASC[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
S	The communication protocol(s) supported by the module: 0: Only the DCON protocol is supported 1: Both the DCON and Modbus RTU protocols are supported
C	The communication protocol currently saved inside that will be used at the next power-on reset: 0: The communication protocol currently saved inside is DCON 1: The communication protocol currently saved inside is Modbus RTU

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01P1 Response: !01

Sets the communication protocol to be used for module 01 to Modbus RTU and the module returns a response indicating that the command was valid.

Command: $01P Response: !0110

Reads the current communication protocol information configured for module 01, and returns a response indicating that the command was valid, with a value of 10, which denotes that the module supports both the DCON and Modbus RTU protocols and that the DCON protocol will be used at the next power-on reset.

Related Commands:

Section 5.22. $AAPN

Related Topics:

Section 2.5. Hardware Configuration

5.22. $AAPN

Description:

This command is used to set the communication protocol to be used by a specified module.

Syntax:

$AAPN[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
P	The command to set the communication protocol
N	The communication protocol to be used: 0: DCON Protocol 1: Modbus RTU Protocol

Note that the INIT DIP Switch (DIP 4) must be set to the ON position before using this command. See Section 1.5 for more details. The new protocol information will be saved inside and will become effective after the next power-on reset.

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01P1 Response: ?01

Attempts to set the communication protocol to be used by module 01 to Modbus RTU, but the module returns a response indicating that the command was invalid because the INIT DIP Switch (DIP 4) has not been set to the ON position. See Section 1.5 for more details.

Command: $01P1 Response: !01

Sets the communication protocol to be used for module 01 to Modbus RTU and the module returns a response indicating that the command was valid.

Command: $01P Response: !0110

Reads the current communication protocol information configured for module 01 returns a response indicating that the command was valid, with a value of 10, which denotes that the module supports both the DCON and Modbus RTU protocols and that the DCON protocol will be used at the next power-on reset.

Response: ?01

Command: $01P1

Attempts to set the current communication protocol into Modbus RTU for module 01, but the module returns a response indicating that the command was invalid because the INIT DIP Switch (DIP 4) hasn't been set to the ON position. See Section 1.5 for more details.

Related Commands:

Section 5.21. $AAP

Related Topics:

Section 2.5. Hardware Configuration

5.23. ~AARD

Description:

This command is used to read the response delay time for a specified module.

Syntax:

~AARD[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RD	The command to read the response delay time

Response:

Valid Command: **!AA(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A two-digit hexadecimal value representing the response delay time. The valid range is 00 to 1E in 1ms intervals.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~03RD1E Response: !03

Sets the response delay time for module 03 to 1E (30 ms), and returns a response indicating that the command was valid.

Command: ~03RD Response: !031E

Reads the response delay time for module 03 and returns a response indicating that the command was valid, with a value of 1E (30 ms).

Related Commands:

Section 5.24 ~AARDVV

5.24. ~AARDVV

Description:

This command is used to set the response delay time for a specified module.

Syntax:

~AARDVV[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RD	The command to set the response delay time.
VV	A two-digit hexadecimal value representing the response delay time in milliseconds. The valid range is 00 to 1E in 1ms intervals.

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~03RD1E Response: !03

Sets the response delay time for module 03 to 1E (30 ms), and returns a response indicating that the command was valid.

Command: ~03RD Response: !031E

Reads the response delay time for module 03 and returns a response indicating that the command was valid, with a value of 1E (30 ms).

Command: ~03RD1F Response: ?03

Attempts to set the response delay time for module 03 to 1F (31 ms), but the module returns a response indicating that the command was invalid because the value specified for the response delay time was not within the valid range.

Related Commands:

Section 5.23. ~AARD

5.25. @AADI

Description:

This command is used to read digital input status and the current status of the PIR relay output channel and PIR active status on a specified module.

Syntax:

@AADI[CHKSUM](CR)

$ Delimiter character

AA The address of the module to be configured in hexadecimal format (00 to FF)

DI The command to read the digital input status and the current status of the PIR relay output channel and PIR active status

Response:

Valid Command: **!AASOOII[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

AA The address of the responding module in hexadecimal format (00 to FF)

S High temperature alarm enable status
0=alarm disable
1=momentary alarm enabled
2=latch alarm enabled.

OO The status of the PIR relay output channel represented by a two-digit hexadecimal value. 00: relay output is inactive; 01: relay output is active.

II The status represented by a two-digit hexadecimal value.
Bit 0: The status of the PIR trigger
Bit 1: The status of the digital input

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01DI Response: !100101

Reads the status of the Relay Output channel for module 01 and returns a response indicating that the command was valid, and that both the PIR Relay Output channel and the status of the PIR are active.

Related Commands:

Section 5.26 @AADONN

5.26. @AADONN

Description:

This command is used to set the relay output to on/off on a specified module.

Syntax:

@AADI[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
DO	The command to set the relay output
NN	The status of the relay output channel represented by a two-digit hexadecimal value. 00: Set the relay output to off 01: Set the relay output to on

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01DO01 Response: ?01

Attempts to set the relay output channel 0 to on for module 01 and returns a response indicating that the command was invalid, because the relay output is set to link to PIR sensor.

Command: @01LC9C000 Response: !01

Disable the linking of relay output and buzzer to PIR sensor of module 01, and the module returns a response indicating that the command was valid.

Command: @01DO01 Response: !01

Sets the relay output channel 0 to on for module 01, and the module returns a response indicating that the command was valid.

Command: @01DI Response: !0100100

Reads the status of the relay output channel for module 01 and returns a response indicating that the command was valid, and that both the relay output channel and the status of the PIR are active.

Related Commands:

Section 5.16. @AALC9C0NN, 4.25. @AADI

Notes:

Enable the relay output when PIR sensor is triggered. Relay output cannot use this command to set.

5.27. ~AAD

Description:

This command is used to read whether the relay output signal for a specified module is active or inactive.

Syntax:

~AAD [CHKSUM](CR)

~ Delimiter character

AA The address of the module to be read in hexadecimal format (00 to FF)

D The command to read whether the relay output signal is active or inactive

Response:

Valid Command: **!AAVV[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

AA The address of the responding module in hexadecimal format (00 to FF)

VV A two-digit hexadecimal value representing the status of the relay output signal. See below for details.

7	6	5	4	3	2	1	0
Reserved						OAS	Reserved

Key	Description
OAS	Specifies the status of the PIR Relay Output signal 0: An output value of 0 indicates that the relay is inactive An output value of 1 indicates that the relay is active 1: An output value of 0 indicates that the relay is active An output value of 1 indicates that the relay is inactive

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~02D02 Response: !02

Sets the relay output signal for module 02 to 02, which denotes that the relay output channel is in inactive mode, and returns a response indicating that the command was valid.

Command: ~02D Response: !0202

Reads the status of the relay output signal for module 02 and returns a response indicating that the command was valid, with a value of 02, which denotes that the relay output channel is in inactive mode.

Related Commands:

Section 5.28. ~AADVV

5.28. ~AADVV

Description:

This command is used to set the relay output signal for a specified module to active or inactive.

Syntax:

~AADVV[CHKSUM](CR)

~ Delimiter character

AA The address of the module to be read in hexadecimal format (00 to FF)

D The command to set the relay output to active or inactive

VV A two-digit hexadecimal value representing the status of the relay output signal. See below for details.

7	6	5	4	3	2	1	0
Reserved						OAS	Reserved

Key	Description
OAS	Specifies the status of the PIR Relay Output signal 0: An output value of 0 indicates that the relay is inactive An output value of 1 indicates that the relay is active 1: An output value of 0 indicates that the relay is active An output value of 1 indicates that the relay is inactive

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

AA The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~02D02 Response: !02

Sets the relay output signal for module 02 to 02, which denotes that the relay output channel is in inactive mode, and returns a response indicating that the command was valid.

Command: ~02D Response: !0202

Reads the status of the relay output signal for module 02 and returns a response indicating that the command was valid, with a value of 02, which denotes that the relay output channel is in inactive mode.

Command: ~02D07 Response: ?02

Attempts to set the relay output signal for module 02 to 07, but returns a response indicating that the command was invalid because the output value was not within the valid range.

Related Commands:

Section 5.27. ~AAD

5.29. @AACHC0

Description:

This command is used to clear the status of a latched high alarm for a specified module.

Syntax:

@AACHC0[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
CHC0	The command to clear the status of the latched high alarm.

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @03CHC0 Response: !03

Clears the status of the latched high alarm for module 03, and returns a response indicating that the command was valid.

Related Commands:

Section 5.31. @AAEAT, Section 5.32. @AAHI(Data), Section 5.30. @AADA, Section 5.35. @AARH, Section 5.36. @AARAO

5.30. @AADA

Description:

This command is used to disable the high alarm function for a specified module.

Syntax:

@AAHI(Data)[CHKSUM](CR)

@ Delimiter character

AA The address of the module to be read in hexadecimal format (00 to FF)

DA The command to disable the high alarm function

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

AA The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command:@01EAM Response:!01

Enables the high alarm function for module 01 and sets the alarm type to momentary, and returns a response indicating that the command was valid.

Command:@03DA Response:!03

Disables the high alarm function for module 03, and returns a response indicating that the command was valid.

Related Commands:

Section 5.31. @AAEAT, Section 5.32. @AAHI(Data), Section 5.29. @AACHC0, Section 5.35. @AARH, Section 5.36. @AARAO

5.31. @AAEAT

Description:

This command is used to enable the high alarm function for a specified module and set the alarm type.

Syntax:

@AAEAT[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
EA	The command to enable the high alarm function
T	The Alarm Type: M: Momentary Alarm L: Latch Alarm

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command:@01EAM Response:!01

Enables the high alarm function for module 01 and sets the alarm type to momentary, and returns a response indicating that the command was valid.

Command:@03DA Response:!03

Disables the high alarm function for module 03, and returns a response indicating that the command was valid.

Related Commands:

Section 5.32. @AAHI(Data), Section 5.30. @AADA, Section 5.29. @AACHC0, Section 5.35. @AARH, Section 5.36. @AARAO

5.32. @AAHI(Data)

Description:

This command is used to set the high alarm limits for a specified module.

Syntax:

@AAHI(Data)[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
HI	The command to set the high alarm limits
(Data)	A signed value representing the high alarm limits in degrees Celsius in the format xxx.xx. The valid range is +000.00 to +999.99 degrees Celsius.

Response:

Valid Command: **!AA[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01HI+086.00 Response:!01

Sets the high alarm limits for module 01 to +86.00 degrees Celsius, and returns a response indicating that the command was valid.

Command: @01RH Response: !03+086.00

Reads the high alarm limits for module 01, and returns a response indicating that the command was valid, with a value of +086.00, which denotes that the high alarm limits is +86.0 degrees Celsius.

Command: @01HI+1000.00 Response:?01

Attempts to set the high alarm limits for module 01 to +1000.00 degrees Celsius, but returns a response indicating that the command was invalid because the specified value was not within the valid range.

Related Commands:

Section 5.31. @AAEAT, Section 5.30. @AADA, Section 5.29. @AACHC0, Section 5.35. @AARH, Section 5.36. @AARAO

5.33. @AAHO

Description:

This command is used to read the offset of the relative humidity for a specified module.

Syntax:

@AAHO [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
HO	The command to read the offset of relative humidity

Response:

Valid Command: **!AA(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A two-digit hexadecimal value to represent the offset of relative humidity

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01HO+001.00 Response:!01

Sets the offset +1% of relative humidity, and returns a response indicating that the command was valid.

Command: @01HO Response: !01+001.00

Reads the offset of relative humidity for module 01, and returns a response indicating that the command was valid, with a value of +001.00, which denotes that the offset value is +1%.

Related Commands:

Section 5.34. @AAHO(Data)

5.34. @AAHO(Data)

Description:

This command is used to set the offset of the relative humidity for a specified module.

Syntax:

@AAHO(Data) [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
HO	The command to set the offset of relative humidity
(Data)	A signed value representing the offset of relative humidity in the format xxx.xx. The valid range is -100.00% to +100.00%.

Response:

Valid Command: **!AA [CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01HO+001.00 Response:!01

Sets the offset +1% of relative humidity, and returns a response indicating that the command was valid.

Command: @01HO Response: !01+001.00

Reads the offset of relative humidity for module 01, and returns a response indicating that the command was valid, with a value of +001.00, which denotes that the offset value is +1%.

Related Commands:

Section 5.33. @AAHO

5.35. @AARH

Description:

This command is used to read the current high alarm limits for a specified module.

Syntax:

@AARH[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RH	The command to read the current high alarm limits.

Response:

Valid Command: **!AA(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A signed value representing the high alarm limits in degrees Celsius in the format xxx.xx. The valid range is +000.00 to +999.99.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @03HI+090.50 Response: !03

Sets the high alarm limits for module 03 to +90.5 degrees Celsius, and returns a response indicating that the command was valid.

Command: @03RH Response: !03+090.50

Reads the high alarm limits for module 03, and returns a response indicating that the command was valid, with a value of +090.50, which denotes that the high alarm limits is +90.5℃.

Related Commands:

Section 5.31. @AAEAT, Section 5.32. @AAHI(Data), Section 5.30. @AADA, Section 5.29. @AACHC0, Section 5.36. @AARAO

5.36. @AARAO

Description:

This command is used to read the currently activated alarm for a specified module.

Syntax:

@AARAO [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RAO	The command to read the currently activated alarm.

Response:

Valid Command: **!AA(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
HH	A two-digit hexadecimal value to represent the currently activated high alarms 00: High alarm is not active. 01: High alarm is active.
00	Reserved.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @03RAO Response:!030000

Reads the currently activated high alarm, and returns a response indicating that the command was valid.

Command: @03RH Response: !03+090.50

Reads the High Alarm limits for module 03, and returns a response indicating that the command was valid, with a value of +090.50, which denotes that the High Alarm limits is +90.5 degrees Celsius.

Related Commands:

Section 5.31. @AAEAT, Section 5.32. @AAHI(Data), Section 5.30. @AADA, Section 5.29. @AACHC0

5.37. @AATO

Description:

This command is used to read the offset of the temperature for a specified module.

Syntax:

@AAHO [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
TO	The command to read the offset of temperature

Response:

Valid Command: **!AA(Data)[CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A two-digit hexadecimal value to represent the offset of Centigrade

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01TO+001.00 Response:!01

Sets the offset +1℃, and returns a response indicating that the command was valid.

Command: @01TO Response: !01+001.00

Reads the offset of temperature for module 01, and returns a response indicating that the command was valid, with a value of +001.00, which denotes that the offset value is +1℃.

Related Commands:

Section 5.38. @AATO(Data)

5.38. @AATO(Data)

Description:

This command is used to set the offset of the temperature for a specified module.

Syntax:

@AAHO(Data) [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
TO	The command to set the offset of temperature
(Data)	A signed value representing the offset of Celsius in the format xxx.xx. The valid range is -100.00℃ to +100.00℃.

Response:

Valid Command: **!AA [CHKSUM](CR)**

Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01TO+001.00 Response:!01

Sets the offset +1℃, and returns a response indicating that the command was valid.

Command: @01TO Response: !01+001.00

Reads the offset of temperature for module 01, and returns a response indicating that the command was valid, with a value of +001.00, which denotes that the offset value is +1℃.

Related Commands:

Section 5.37. @AATO

6. Modbus RTU Protocol

The Modbus protocol was originally developed for Modicon controllers by Modicon Inc. Detailed information related to the Modbus RTU protocol can be found at: http://www2.schneider-electric.com/sites/corporate/en/products-services/automation-control/automation-control.page.

You can also visit http://www.modbus.org for more valuable information.

The PIR-230 module supports the Modbus RTU protocol, with communication Baud Rates ranging from 1200 bps to 115200 bps. The parity, data bits and stop bits are fixed as no parity, 8 data bits and 1 stop bit. The following Modbus functions are supported.

Function Code	Description	Section
0x01	Reads the Coils	3.1
0x02	Reads the Discrete Inputs	3.2
0x03	Reads Multiple Registers	3.3
0x04	Reads Multiple Input Registers	3.4
0x05	Writes a Single Coil	3.5
0x06	Writes a Single Register	3.6
0x0F	Writes Multiple Coils	3.7
0x10	Writes Multiple Registers	3.8

Error Response

If the function specified in the message is not supported, then the module responds as below.

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	Function Code + 0x80
02	Exception Code	1	01

Note:

1. **If a CRC mismatch occurs, the module will not respond.**
2. **The address mapping for the Modbus protocol is 0-Based.**

6.1. Modbus Address Mapping

General Commands

<table>
<tr><th>Address</th><th>Description</th><th>Attribute</th></tr>
<tr><td>00257</td><td>Reads/sets the Communication Protocol
0: DCON
1: Modbus RTU</td><td>R/W</td></tr>
<tr><td>10273</td><td>Reads the Reset Status:
0: This is NOT the first time the module has been read since being powered on
1: This is the first time the module has been read since being powered on</td><td>R</td></tr>
<tr><td>30485</td><td>Reads/sets the Module address. The valid range is 1 to 247.</td><td>R/W</td></tr>
<tr><td>30486</td><td>Reads/sets the Baud Rate and the Data Format:
Bits 5:0 (Baud Rate)
<table><tr><td>Code</td><td>0x03</td><td>0x04</td><td>0x05</td><td>0x06</td></tr><tr><td>Baud</td><td>1200</td><td>2400</td><td>4800</td><td>9600</td></tr><tr><td>Code</td><td>0x07</td><td>0x08</td><td>0x09</td><td>0x0A</td></tr><tr><td>Baud</td><td>19200</td><td>38400</td><td>57600</td><td>115200</td></tr></table>
Baud Rate, valid range: 0x03 to 0x0A
Bits 7:6 (Data Format)
00: no parity, 1 stop bit
01: no parity, 2 stop bits
10: even parity, 1 stop bit
11: odd parity, 1 stop bit</td><td>R/W</td></tr>
<tr><td>30488</td><td>Reads/sets the Response Delay Time in milliseconds. The valid range is 0 to 30 ms (00 to 1E in 1 ms intervals).</td><td>R/W</td></tr>
<tr><td>40481-40482</td><td>Reads the Firmware Version</td><td>R</td></tr>
<tr><td>40483-40484</td><td>Reads the Name of the Module</td><td>R</td></tr>
</table>

PIR-related Commands

Address	Description	Attribute
00001	Relay Output. (If enable the relay output when PIR sensor is triggered, read only) 1: Active 0: Inactive	R/W
00262	Enables/Disables the high temperature alarm. 0: Disabled 1: Enabled	R/W
00263	Reads/Sets the high temperature alarm type. 0: Momentary Alarm 1: Latch Alarm	R/W
00274	The status of Buzzer 0: Inactive 1: Active	R/W
00277	Enables/Disables the relay output is active when PIR sensor is triggered. 0: Disabled 1: Enabled	R/W
00278	Enables/Disables the buzzer output when PIR sensor is triggered. 0: Disabled 1: Enabled	R/W
00305	Reads/Sets the high alarm status, write 1 to clear latched high alarm1	R/W
10033	Reads the current status of PIR trigger	R
10034	Reads the status of digital input	R
30001	Reads the temperature of Centigrade (unit 0.01℃)	R
30002	Reads the temperature of Fahrenheit (unit 0.01℉)	R
30003	Reads the real humidity (unit 0.01%)	R
40226	Reads/Sets the high alarm limits	R/W
40450	Reads/Sets relative humidity offset (unit 0.01%)	R/W
40451	Reads/Sets temperature offset (unit 0.01℃)	R/W

<table>
<tr>
<td>40513</td>
<td>Reads/Sets the PIR output On time when the PIR sensor is triggered in hexadecimal format. The valid range is 00 to 0F.
<table>
<tr><th>Hex</th><th>Secs</th><th>Hex</th><th>Secs</th><th>Hex</th><th>Secs</th><th>Hex</th><th>Secs</th></tr>
<tr><td>00</td><td>2</td><td>01</td><td>4</td><td>02</td><td>6</td><td>03</td><td>8</td></tr>
<tr><td>04</td><td>16</td><td>05</td><td>33</td><td>06</td><td>49</td><td>07</td><td>66</td></tr>
<tr><td>08</td><td>131</td><td>09</td><td>262</td><td>0A</td><td>393</td><td>0B</td><td>524</td></tr>
<tr><td>0C</td><td>1049</td><td>0D</td><td>2097</td><td>0E</td><td>3146</td><td>0F</td><td>4194</td></tr>
</table>
</td>
<td>R/W</td>
</tr>
<tr>
<td>40514</td>
<td>Reads/Sets the luminance limit value for the PIR sensor. The valid range is 0 to 4, where.
0: Disabled
1: 100 Lux
2: 200 Lux
3: 500 Lux
4: 1000 Lux</td>
<td>R/W</td>
</tr>
<tr>
<td>30516</td>
<td>Reads/Sets the active delay time for the PIR Output in milliseconds after power on. The valid range is 0 to 0xBB8 (0 to 3000 milliseconds).</td>
<td>R/W</td>
</tr>
<tr>
<td>30517</td>
<td>Reads/Sets sensitivity value for the PIR Sensor. The valid range is 0 to 9, where a lower value denotes a higher sensitivity.</td>
<td>R/W</td>
</tr>
</table>

Modbus RTU Function Description:

(0xxxx): 0x05, 0x0F Function Code

(1xxxx): 0x01 Function Code

(3xxxx): 0x06, 0x10 Function Code

(4xxxx): 0x03 Function Code

Wi-Fi Related Modbus Address Mappings (Base 1)

Address	Description	Attribute
40642	This parameter is used to specify the Wi-Fi mode of the DL-300-WF device. It can be 0 for station mode or 2 for AP mode. For AP mode, only one device can be connected.	R/W
40643	This parameter is used to specify which security protocol is used to secure wireless computer network. It can be 0 for open, 1 for WEP, or 2 for WPA/WPA2. It is recommended to use WPA/WPA2 if possible.	R/W
40644 ~ 40650	WEP password Byte 0: password length Byte 1 ~ 13: password	R/W
40651 ~ 40682	WPA/WPA2 password Byte 0: password length Byte 1 ~ 63: password	R/W
40683	This parameter is used to specify whether to turn on the DHCP server function. It can be 0 for turning off and 1 for turning on. It is only available to the AP mode.	R/W
40684 ~ 40685	This parameter is used to specify the start IP address of the allocated IP by the DHCP server when the DHCP server function is turned on. It is only available to the AP mode.	R/W
40687	IP address type in station mode, 0 for static type, 1 for DHCP This parameter is only available to the station mode and it can be 0 for Static IP or 1for DHCP. If DHCP is supported by the AP you would like to connect, then DHCP should be selected. Otherwise, select Static IP and the following three parameters IP Address, Subnet Mask and Gateway should be set, too.	R/W
40688 ~ 40689	Each SL device connected to the Wi-Fi network must have its own unique IP address. This parameter is used to assign a specific IP address.	R/W
40690 ~ 40691	This parameter is used to assign the subnet mask for the DL-300-WF device. The subnet mask indicates which portion of the IP address is used to identify the local network or subnet.	R/W

Address	Description	Attribute
40692 ~ 40693	This parameter is used to assign the IP address of the gateway to be used by the SL device. A gateway (or router) is a device that is used to connect an individual network to one or more additional networks.	R/W
40694 ~ 40709	This parameter is used to specify the Service Set Identifier, SSID. For station mode, specify the SSID of the AP you would like to connect. For AP mode, the SSID will be used by the device to be connected.	R/W
40710	This parameter is used to specify which channel is used for Wi-Fi transmission. It can be 1 to 11. It is only available to the AP mode.	R/W
40711	This parameter is used to set the local port of the Wi-Fi interface to be used by the Modbus slave device. The default value is 502.	R/W
40715	Write 1 to let the new Wi-Fi settings take effect.	W
40716 ~ 40718	Wi-Fi module MAC address	R
40719	Firmware version of the Wi-Fi module	R
40720	Wi-Fi module status High byte 0: not configured 1: not connected 2: connected 3: reconnecting Low byte 0: not connected 1: high signal strength 2: medium signal strength 3: low signal strength	R

6.2. Function 01 (0x01) Read Coils

This function code is used to read the values at addresses 0xxxx and 1xxxx.

Request

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x01
02 - 03	Starting Address	2	Refer to the Modbus Address Mapping Table for details.
04 - 05	Number of Addresses Requested	2	0x0001 to 0x0001 + *N

*N = Number of addresses requested

Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x01
02	Byte Count	1	*N
03	Value from the Requested Address	*N	

*N = (Number of addresses requested / 8)

Error Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x81
02	Exception Code	1	Refer to the Modbus standard for more details.

6.3. Function 02 (0x02) Read Discrete Input

This function code is used to read the value at address 1xxxx.

Request

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x02
02 - 03	Starting Address	2	0x0020 to 0x003F
04 - 05	Number of Addresses Requested	2	0x0001 to 0x0001 + *N

*N = Number of addresses requested

Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x02
02	Byte Count	1	*N
03	Value from the Requested Address	*N	

*N = (Number of addresses requested / 8)

Error Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x82
02	Exception Code	1	Refer to the Modbus standard for more details.

6.4. Function 03 (0x03) Read Multiple Registers

This function code is used to read the values at addresses 3xxxx and 4xxxx.

Request

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x03
02 - 03	Starting Address	2	Refer to the Modbus Address Mapping Table for details.
04 - 05	Number of Addresses Requested	2	0x0001 to 0x0001 + *N

*N = Number of addresses requested

Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x03
02	Byte Count	1	*N x 2
03 -	Value from the Requested Address	*N x 2	.

*N = Number of addresses requested

Error Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x83
02	Exception Code	1	Refer to the Modbus standard for more details.

6.5. Function 04 (0x04) Read Multiple Input Registers

This function code is used to read the values at address 4xxxx.

Request

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x04
02 - 03	Starting Address	2	Refer to the Modbus Address Mapping Table for details.
04 - 05	Number of Addresses Requested	2	0x0001 to 0x0001 + *N

*N = Number of addresses requested

Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x04
02	Byte Count	1	*N x 2
03 -	Value from the Requested Address	*N x 2	

*N = Number of addressee requested

Error Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x84
02	Exception Code	1	Refer to the Modbus standard for more details.

6.6. Function 05 (0x05) Write Single Coil

This function code is used to write a value to address 0xxxx.

Request

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x05
02 - 03	Starting Address	2	Refer to the Modbus Address Mapping Table for details.
04 - 05	Value to be written	2	A value of 0xFF00 will set the output to ON. A value of 0x0000 will set it to OFF. All other values are invalid and will not affect the coil.

Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x05
02 - 03	Requested Address	2	The value is the same as bytes 02 and 03 of the Request
04 - 05	Value from the Requested Address	2	The value is the same as bytes 04 and 05 of the Request

Error Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x85
02	Exception Code	1	Refer to the Modbus standard for more details.

6.7. Function 06 (0x06) Write Single Register

This function code is used to write a value to address 3xxxx.

Request

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x06
02 - 03	Starting Address	2	Refer to the Modbus Address Mapping Table for details.
04 - 05	The value to be written	2	

Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x06
02 - 03	Requested Address	2	The value is the same as bytes 02 and 03 of the Request
04 - 05	Value from the Requested Address	2	The value is the same as bytes 04 and 05 of the Request

Error Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x86
02	Exception Code	1	Refer to the Modbus standard for more details.

6.8. Function 15 (0x0F) Write Multiple Coils

This function code is used to write multiple values.

Request

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x0F
02 - 03	Starting Address	2	Refer to the Modbus Address Mapping Table for details.
04 - 05	Number of Addresses Requested	2	0x0001 to 0x0001 + *N
06	Byte Count	1	*N/8
07	The values to be written	1	A bit corresponds to a channel. If the bit is 1, it denotes that the channel that was set is ON. If the bit is 0, it denotes that the channel that was set is OFF.

*N = Number of addresses requested

Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x0F
02 - 03	Starting Address	2	The value is the same as bytes 02 and 03 of the Request
04 - 05	Value from the Requested Address	2	The value is the same as bytes 04 and 05 of the Request

Error Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x8F
02	Exception Code	1	Refer to the Modbus standard for more details.

6.9. Function 16 (0x10) Write Multiple Registers

This function code is used to write multiple values.

Request

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x10
02 - 03	Starting Address	2	Refer to the Modbus Address Mapping Table for details.
04 - 05	Number of Addresses Requested	2	0x0001 to 0x0001 + *N
06	Byte Count	1	*N x 2
07	The values to be written	*N x 2	

*N = Number of addresses requested

Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x10
02 - 03	Starting Address	2	The value is the same as bytes 02 and 03 of the Request
04 - 05	Number of Addresses Requested	2	The value is the same as bytes 04 and 05 of the Request

Error Response

Byte	Description	Length (in Bytes)	Value
00	Address	1	1 to 247
01	Function Code	1	0x90
02	Exception Code	1	Refer to the Modbus standard for more details.

Appendix: FAQ

A. How to update the firmware via Ethernet

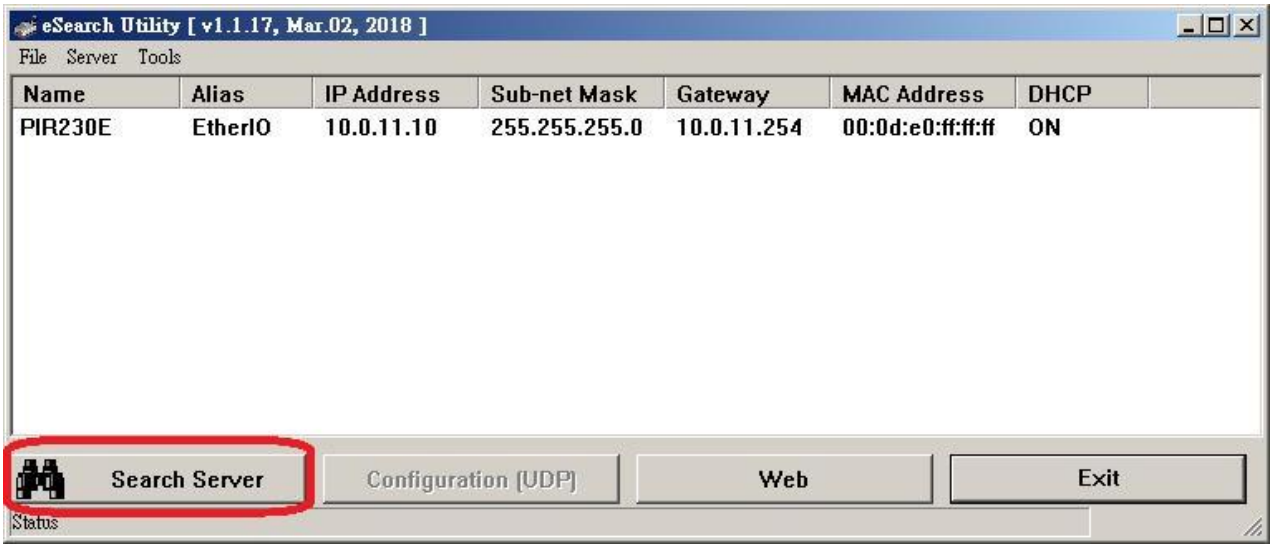
If the module is not functioning correctly (e.g. there is no response to a search request, or if the system LED is continuously displayed as either OFF or ON), download a new image of the firmware from the ICPDAS web site and then update the firmware.

The firmware of the PIR-230-E module is located at:
http://ftp.icpdas.com/pub/cd/pir/pir-230-e/firmware/

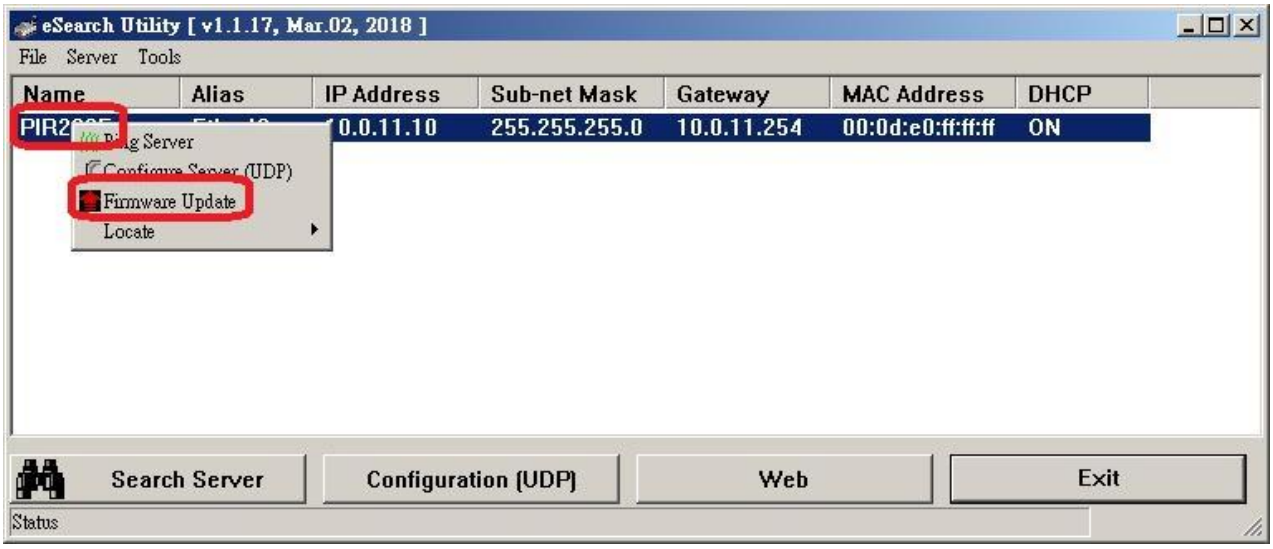
To update the firmware for your PIR-230-E module, connect the module and PC in the same sub-network. Please note that there should be only one network card in the PC.

Download and install the eSearch utility.
http://ftp.icpdas.com/pub/cd/iiot/utility/esearch/

Run the eSearch utility. Click on the **Search Server** button and it should find the PIR-230-E module.

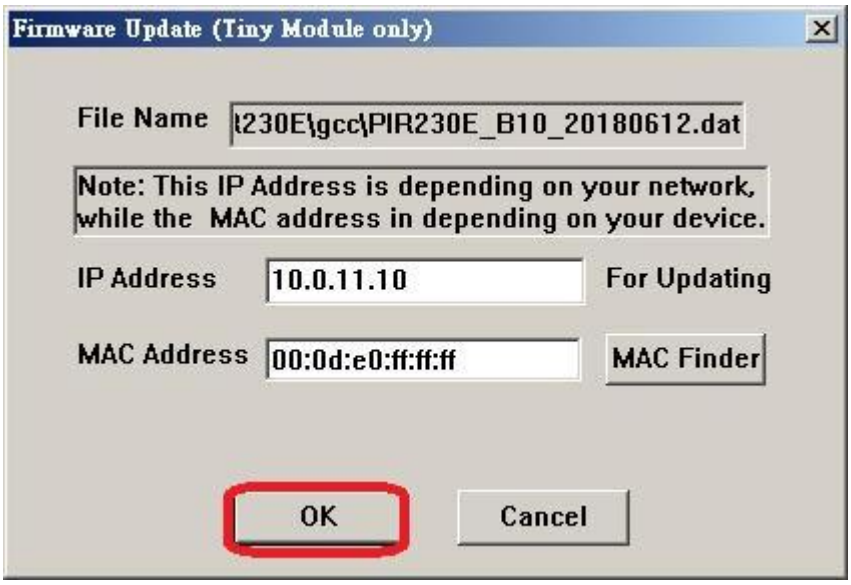


Right click on the **PIR230E** module name then select **Firmware Update**.

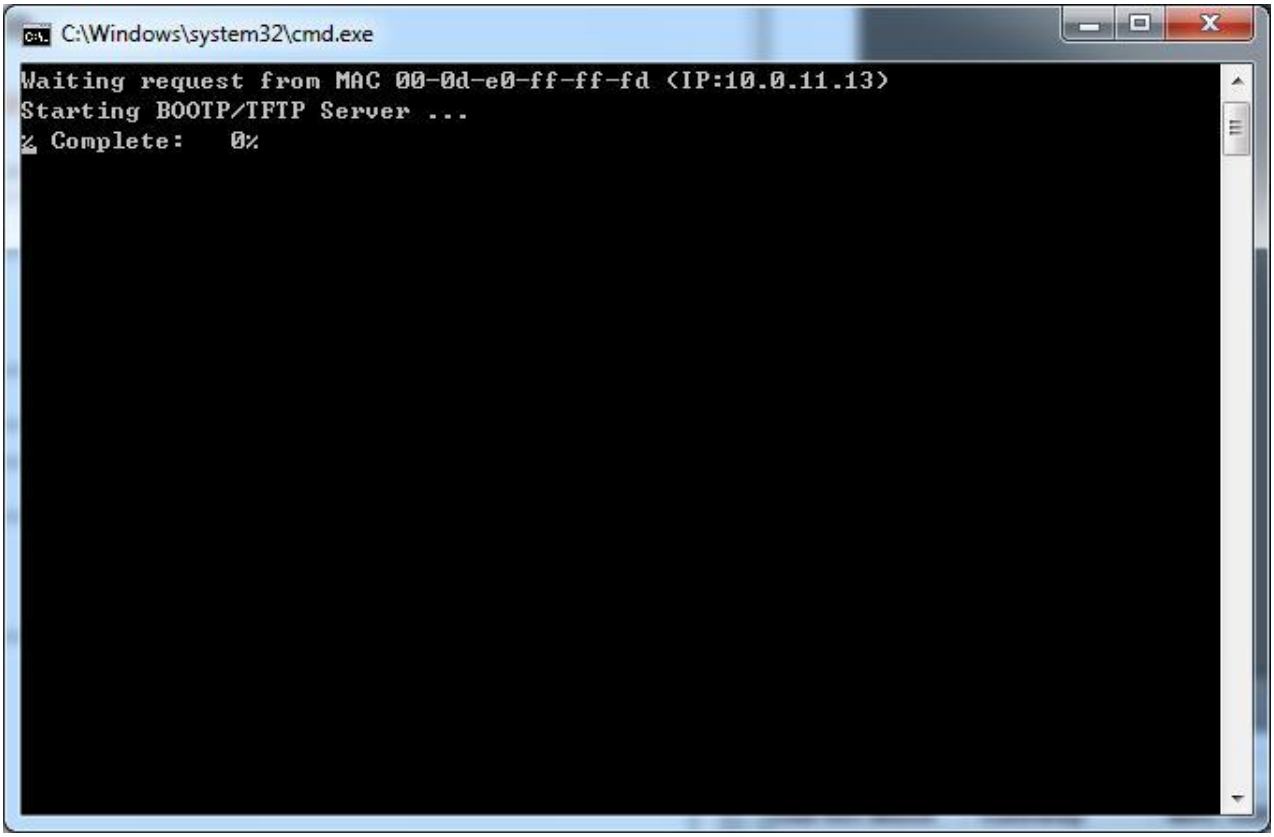


Select the firmware file and click on the **Open** button.

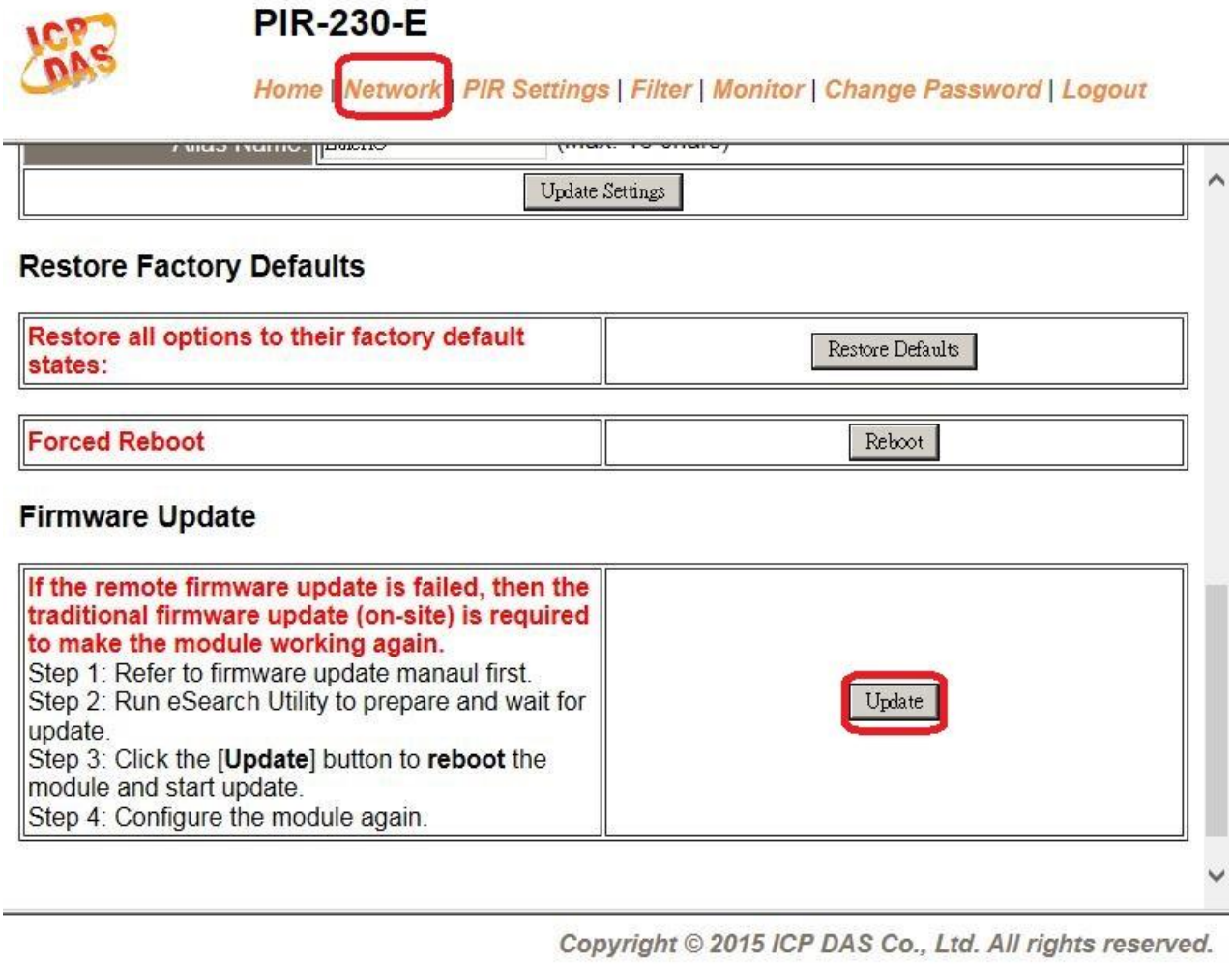
Make sure the IP address and MAC address are correct. Click on the **OK** button.



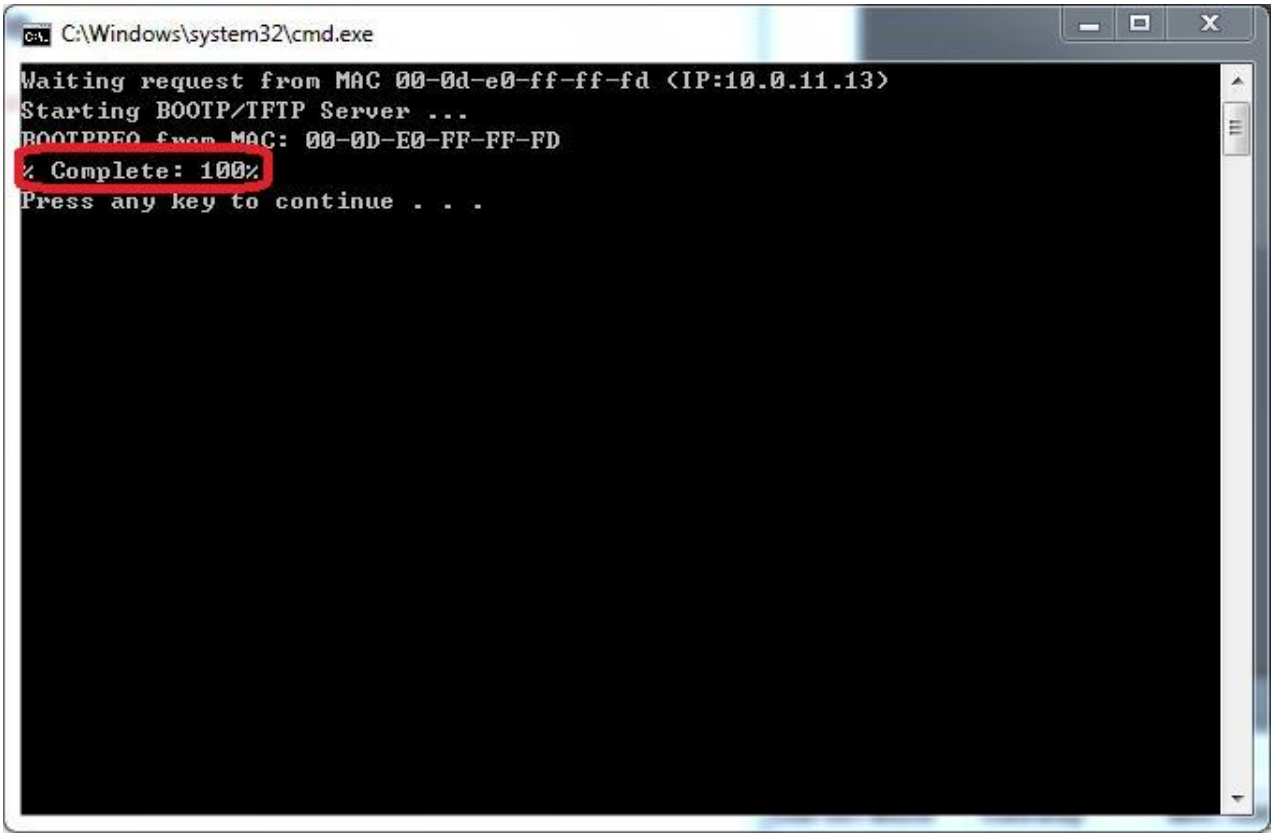
A command prompt window will be displayed to show the progress.



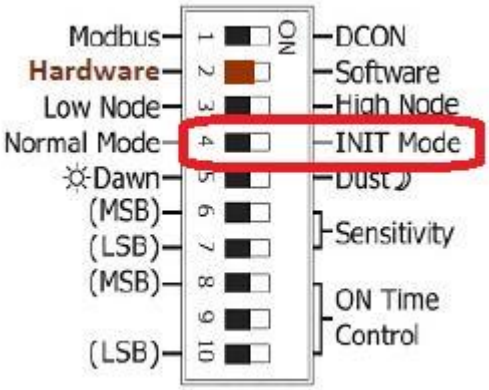
Log in the PIR-230-E web page. Click on the **Network** tab then click on the **Update** button.



When it shows “% Complete: 100%”, the update is finished. You can close the command prompt window.

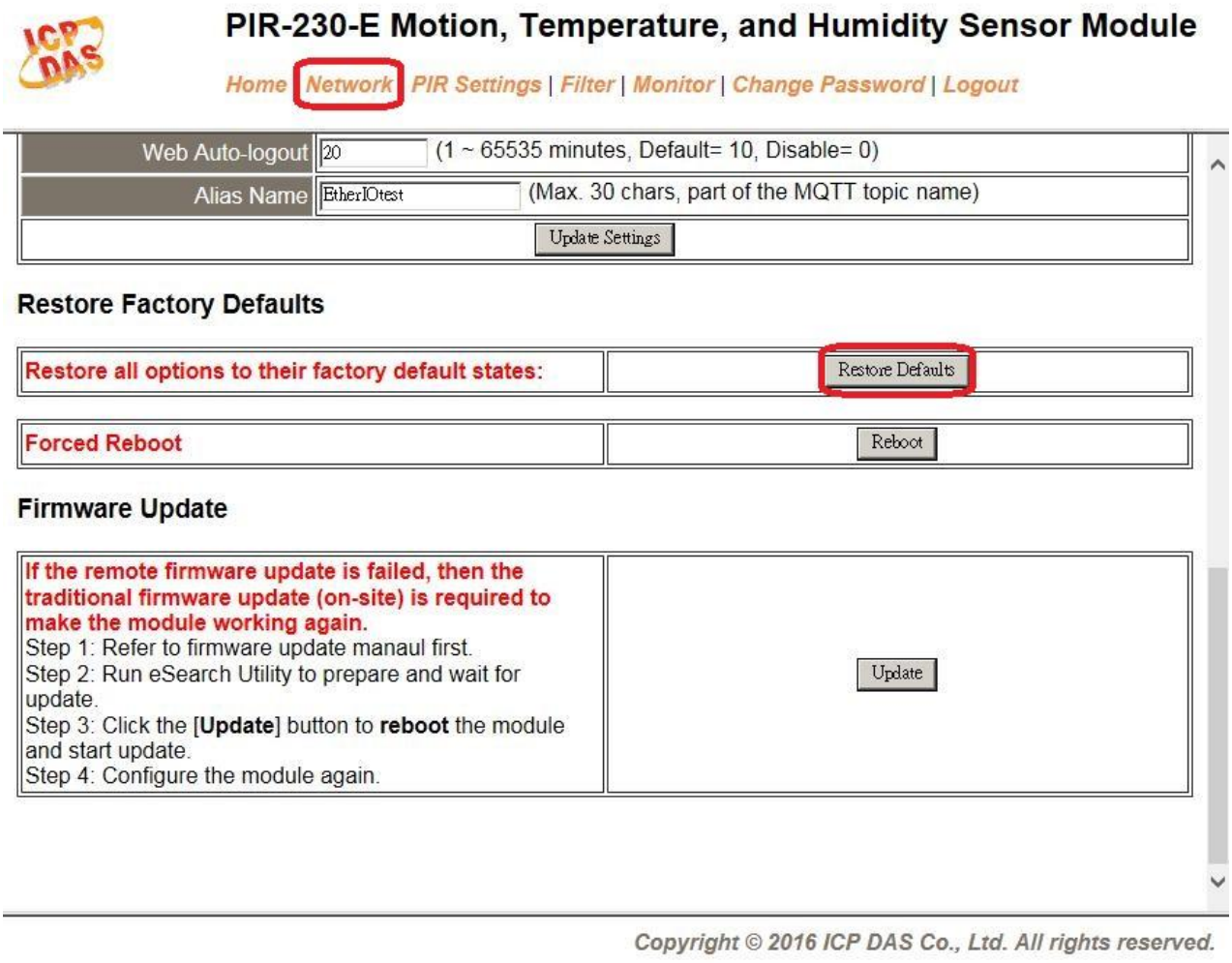


Power off the PIR-230-E module. Turn the INIT switch to ON position, then power on the PIR-230-E module.



Run the eSearch utility to configure the network settings as shown in Section 3.2.

Log in the PIR-230-E web page. Click on the **Network** tab then click on the **Restore Defaults** button.



Turn the INIT switch to OFF position.

Run the eSearch utility to configure the network settings as shown in Section 3.2.

Log in the PIR-230-E web page to configure other settings

.