WEBRelay™ 10+

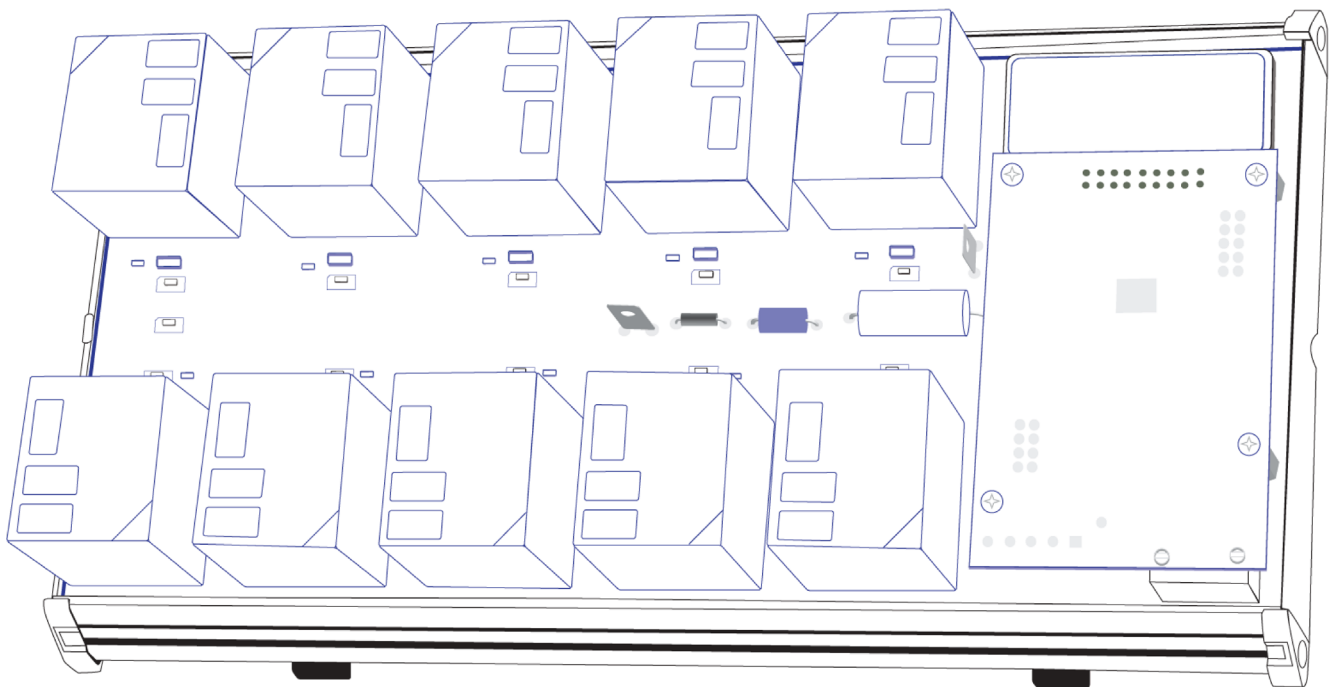## Users Manual

WebRelay-10 Plus (X-WR-10R12-IEP)

CONTROL by WEB™
www.ControlByWeb.com

a division of....
Xytronix Research & Design, Inc.
Nibley, Utah, USA

Table of Contents

Section 1: Introduction

WebRelay-10 Plus™ is a robust, 10-relay module with a built-in web server. It can be controlled and/or monitored over any IP network including private networks, IP-based industrial control networks, and the Internet. Users can operate WebRelay-10 Plus™ using a web browser or a custom application. Computers, PLCs, or automation controllers may control and monitor WebRelay-10 Plus™ without user intervention. This works by sending text commands over the network and reading XML status pages from WebRelay-10 Plus™, or by using Modbus/TCP protocol.
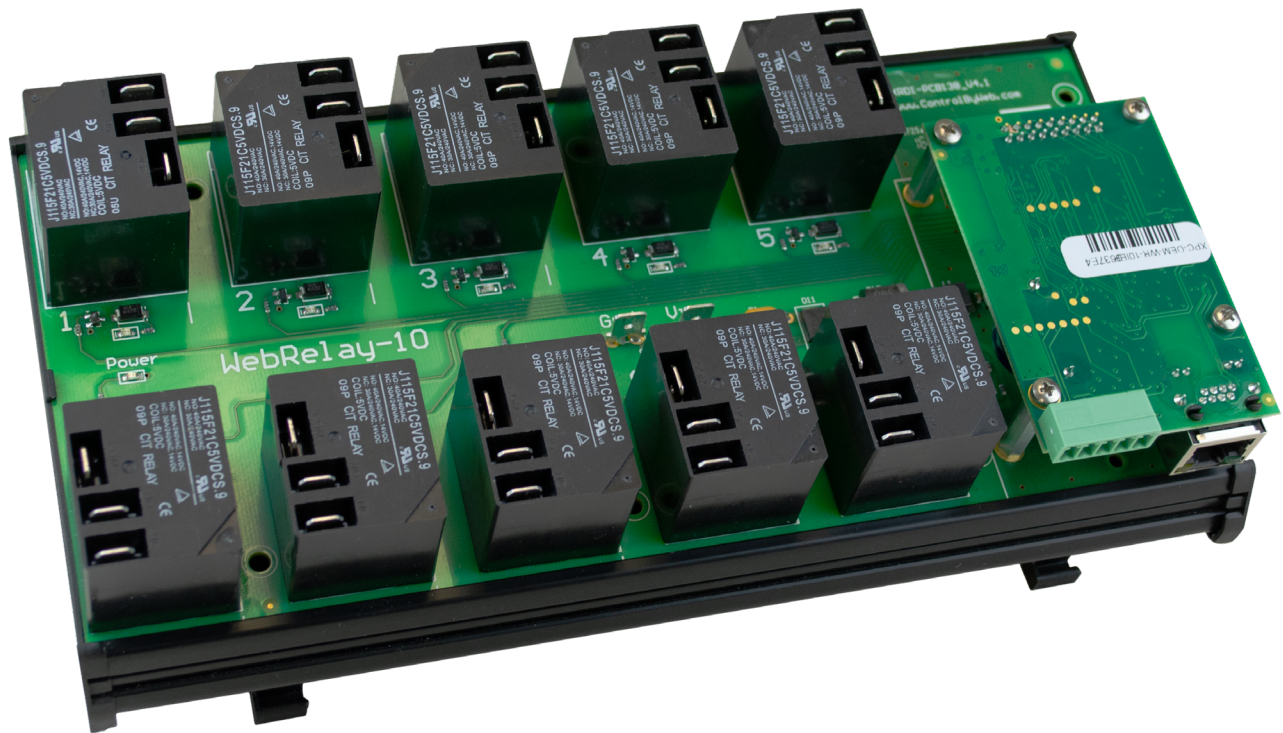
1.1   Features

WebRelay-10 Plus™ includes the following features:

- Ten independent 15-Amp Relays (Form C, SPDT).
- Built-in Web server for browser-based setup and control.
- No special software or device drivers required.
- Customizable email alerts.
- XML/JSON status and **Control Page** make communications with custom computer applications simple.
- Can operate as a Modbus/TCP slave.
- On/Off and Pulse modes.
- Advanced functionality through BASIC scripts.
- Wide power supply range (9-28 VDC).
- DIN-Rail Mountable.
- LED relay state indicators.
- ¼ inch tab-connectors.
- UL/CUL listed relays (E197852).
- Event scheduler with yearly calendar. Schedule up to 100 events.
- Up to sixteen temperature and/or humidity sensors can be added for environmental monitoring.
- Dry-contact sensors or switches can be connected for local control of relays or for monitoring external devices.
- Logging: log relay changes, events, Modbus requests, high/low temperatures, input changes, network traffic, and more.
- Real-time clock, can automatically adjust for daylight savings time, **Sync with NTP server**.
- System log provides detailed diagnostic information.
- Configure manually or with DHCP.

Note: WebRelay-10 Plus™ sends out email messages only. Most wireless carriers provide free services to convert email messages to text messages.

1.2   Applications

WebRelay-10 Plus™ has many applications. It works very well as a stand-alone device that can be controlled using a web browser, or as a convenient way to add I/O to a computer. It can be used in industrial applications to control motors, lights, coils, pumps, valves, etc.

Using the additional features found on WebRelay-10 Plus™, you can monitor temperature and/or humidity, control multiple relays using an input, schedule reoccurring events, log relay state changes, and much more.

1.3 WebRelay-10 Plus™ Models Available

X-WR-10R12-IEP	WebRelay-10 Plus™ includes temperature/humidity monitoring (humidity monitoring requires optional humidity sensor), discrete inputs, real-time clock, event scheduling, capacitor power backup, and logging.

1.3.1 Optional Accessories

Accessory	Description	Part Number
Temperature Sensor	Digital temperature sensor with 12 inch wire leads. Note: Leads may be extended	X-DTS-U
Temperature Sensor (Wall Mount)	Digital temperature sensor housed in vented plastic enclosure	X-DTS-WMX
Temperature/Humidity Sensor (Wall Mount)	Digital temperature and humidity sensor housed in vented plastic enclosure.	X-DTHS-WMX

1.4   Connectors & Indicators
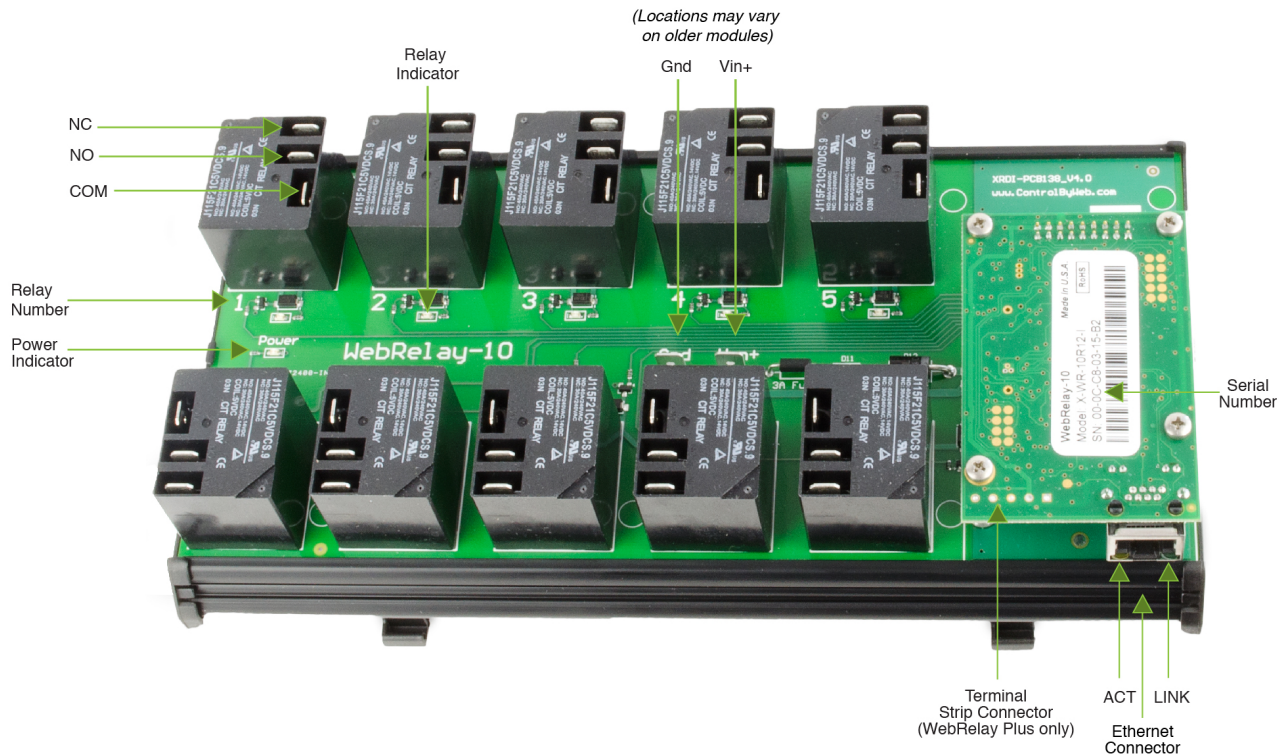
Figure 1.4a - Connections & Indicators

Module Power Connectors
Two 1/4" tab connectors located on the circuit board are used to apply power to the module.

Relay Connectors
Each relay has three ¼" tab connectors for common, normally open, and normally closed contacts. Female mating connectors (not included) are required for connecting user circuits.

Network Connector
The Ethernet connector is located on a small CPU daughter card that is mounted to the main relay board.

I/O Connector
WebRelay-10 Plus™ includes a 5-position removable terminal connector for discrete inputs and temperature and/or humidity sensors.

Relay Indicators
Ten green LEDs indicate when each relay is energized. The LEDs are located next to the corresponding relays. When the relay is energized, the common and normally open contacts are closed. When the relay is not energized, the common and normally closed contacts are closed.

Module Power Indicator
The green power LED indicator is located on the main relay board in between relays 1 and 6.

Ethernet Indicators
The LINK LED is illuminated green when the module is properly connected to an Ethernet network and is ready to communicate. Network communications will only occur if this LED is illuminated. The ACT LED flashes amber when activity is detected on the network.

1.5 Accessing WebRelay-10 Plus™

Standard Access Using a Web Browser

WebRelay-10 Plus™ has a built-in web server that provides simple web pages that can be accessed directly using a standard web browser. This allows users to access the unit with NO SPECIAL SOFTWARE installed on their computer. This is ideal for applications that require a quick, simple solution that does not need to be accessible to more than a few people. This configuration is simple to setup, simple to use, and can be accessed from any computer.

*Note: Network routers may need to be configured to allow access from computers outside of the local network (see **Appendix C: Accessing WebRelay-10 Plus™TM Over The Internet**).*

Section 2: Installation and Setup

Installation consists of mounting WebRelay-10 Plus™, connecting it to an Ethernet network, providing power, configuring via a web browser, and wiring relay contacts to the device that will be controlled.

For WebRelay-10 Plus™ models, inputs and sensors can be wired to the device using the 5-position terminal connector.

Installation Guidelines (Read Before Installing)

- This unit must be installed by qualified personnel.

- This unit must not be installed in unprotected outdoor locations.

- The WebRelay-10 Plus™ must installed inside an NRTL certified industrial control panel or similar NRTL certified fire/electrical enclosure.

- If the WebRelay-10 Plus™ is used in a manner not specified by Xytronix, the protection provided by the equipment may be impaired.

- This unit must not be used for medical, life saving purposes, or for any purpose where its failure could cause serious injury or the loss of life.

- This unit must not be used in any way where its function or failure could cause significant loss or property damage.

- The relay tab terminals are 1/4" x .032". Use UL listed quick-connect connectors suitable for field connections. Use a suitable crimping tool as specified by the connector manufacturer. Use wire rated for 75ºC (min) for connections to the relays.

Security Notes

WebRelay-10 Plus™ does not employ a general purpose computer operating system and does not have features, such as telnet, FTP, SSH, nor uncontrolled open ports. This means it is unlikely for someone to 'break in' to WebRelay-10 Plus™ and access other devices on your local network. The simplicity of WebRelay-10 Plus™ makes it a inherently secure device. Nevertheless, as with any device installed on a network, appropriate security precautions should be observed.

If WebRelay-10 Plus™ is installed on the Internet, it is recommended that passwords be enabled for the **Control Page**. Passwords should be at least 8 characters in length and use a combination of upper and lower case letters and numbers. For additional security, a firewall may be used to limit access to selected IP addresses. Another option may be to set up a Virtual Private Network (VPN) between the network where WebRelay-10 Plus™ resides and the client machine (web browser, another, ControlByWeb™ product, etc.).

Final Installation Notes

This ControlByWeb™ product supports connection to *10 Mbps* and *100 Mbps* networks. Although 100Mbps networks are faster, the amount of data transferred to and from this device is very minimal and little, if any, performance increase will be gained by setting it to *100 Mbps*. There are advantages, however, to operate this device at *10 Mbps*. At *10 Mbps*, less power is required, the unit runs cooler, and the lifetime of the product will be extended.

Any changes to the Ethernet settings will require a removing and re-applying power to WebRelay-10 Plus™.

2.1 Mounting

WebRelay-10 Plus™ can be mounted to a standard (35mm by 7.55mm) DIN-Rail. Attach the module to the DIN-Rail by placing the top hooks on the back of WebRelay-10 Plus™ to the DIN-Rail and then snapping the bottom hooks into place. Remove from the DIN-Rail by depressing the latch-release on the mounting bracket.

See **Appendix L: Mechanical Information** for additional mechanical details.
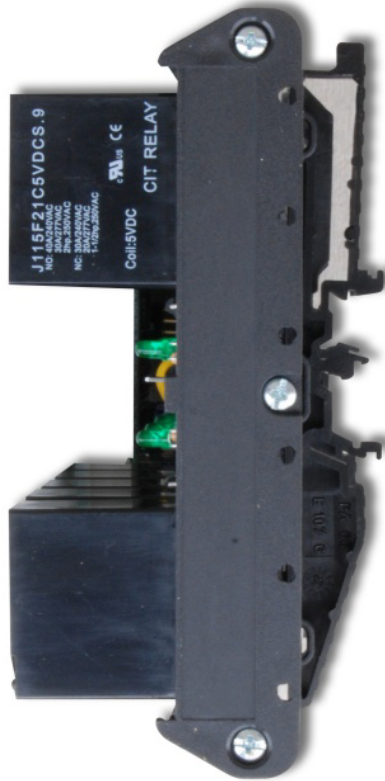
Figure 2.1a - DIN-Rail Mount

2.2   Connections

CAUTION: MAKE SURE POWER IS SHUT OFF BEFORE WIRING!

CAUTION: THIS UNIT SHOULD BE INSTALLED BY A QUALIFIED TECHNICIAN.

MIS-WIRING OR MIS CONFIGURATION COULD CAUSE PERMANENT DAMAGE TO THE WebRelay-10

Plus™ AND/OR THE EQUIPMENT TO WHICH IT IS CONNECTED.

Relay connections are made through the ¼ inch male tab connectors. Mating connectors (¼ inch female tab connectors) can be purchased at electrical supply stores.

It is recommended that the load (device to be controlled) not be connected to WebRelay-10 Plus™ until after WebRelay-10 Plus™ has been configured and tested. By doing this, wiring and configuration mistakes will not cause the attached devices to turn on unexpectedly.

IMPORTANT: WIRES SHOULD BE PROPERLY CRIMPED TO THE CONNECTORS, AND CONNECTORS SHOULD BE PROPERLY ATTACHED TO THE CIRUCIT BOARD OR RELAYS!

2.2.1 Power Supply Connection

WebRelay-10 Plus™ requires power for its internal logic circuits. Connect a 9-28VDC power supply to the Vin+ and Gnd terminals. Note that a regulated power supply is recommended, such as a wall-mount AC-DC adapter. Verify that the adapter is rated for the operating current of WebRelay-10 Plus™ (See **Appendix H: Specifications** for current requirements.)
Multiple WebRelay-10 Plus™ units may be connected to a single power supply by connecting the power supply input terminals in parallel. The power supply must have a high enough current rating to power all units connected. (See **Appendix H: Specifications** for current requirements.)
The power input tab terminals are 1/4" x .032". Use UL listed quick-connect connectors suitable for field connections. Use a suitable crimping tool as specified by the connector manufacturer. Use wire rated for 75ºC (min) for the power connections.

2.2.2 Relay Connection

Direct connection to relay contacts is provided through tab connectors located on the top of the relays. Do not exceed the specified load ratings for relay contacts (see **Appendix H: Specifications** for load ratings). The relay tab terminals are 1/4" x .032". Use UL listed quick-connect connectors suitable for field connections. Use a suitable crimping tool as specified by the connector manufacturer. Use wire rated for 75ºC (min) for connections to the relays.

2.2.3 Network Connection

Connect the Ethernet port to a 10 Base-T or 10/100 Base-T Ethernet connection. This typically connects to an Ethernet hub, switch, or router. For configuration, WebRelay-10 Plus™ may be connected directly to the Ethernet port on a computer using a "crossover" cable. Otherwise, for connection through a hub or router, a standard "straight-through" cable should be used.
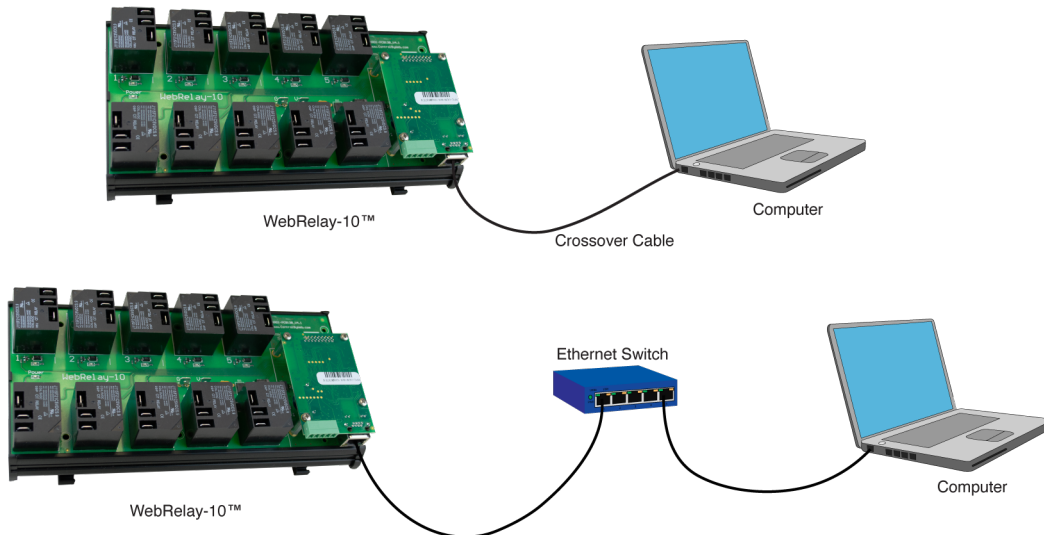
Figure 2.2a - Network Connection

WebRelay-10 Plus™ can be used on a wireless network by connecting through an Ethernet bridge or a wireless router. WebRelay-10 Plus™ is connected directly to the wireless device.
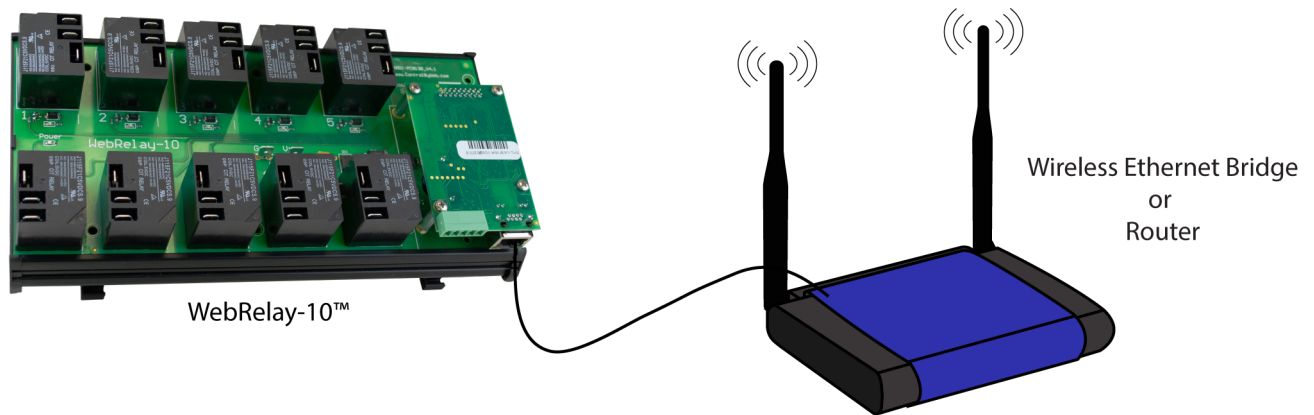
Figure 2.2b - Wireless Connection

Note: The wireless Ethernet bridge or router must be properly configured for the wireless network. Refer to the installation instructions for the wireless device.Terminal Connector.

The removable 5-position terminal connector is used for connecting temperature sensors, humidity sensors, or discrete inputs. The terminals are described in the table below.

IMPORTANT: MAKE SURE WIRES ARE PROPERLY ATTACHED TO THE TERMINALS AND THAT THE SCREW TERMINALS ARE TIGHT!

5-pin connector pinout	
Pin	**Description**
Data	Temperature/Humidity Data. This is the data line connection for the digital temperature/humidity sensors.
+5V Out	This output voltage is used to provide power for the digital temperature/humidity sensors.
Ground	Ground connection for 5VDC output, and Inputs.
Input 1 & 2	Discrete Inputs.

Terminal Connector

Ethernet Connection

Input 1 Input 2 Ground +5V Out Data
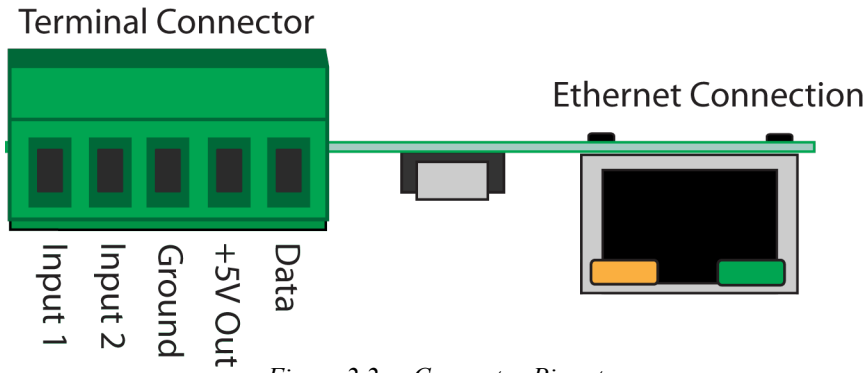
Figure 2.2c - Connector Pinout

Input Connection

Two discrete inputs can be connected to dry contact sensors or switches. Third party sensors can be used for monitoring alarm conditions such as door access switches or transducer thresholds. Inputs can also be used for operator switches and the unit can be configured such that they control the relays.
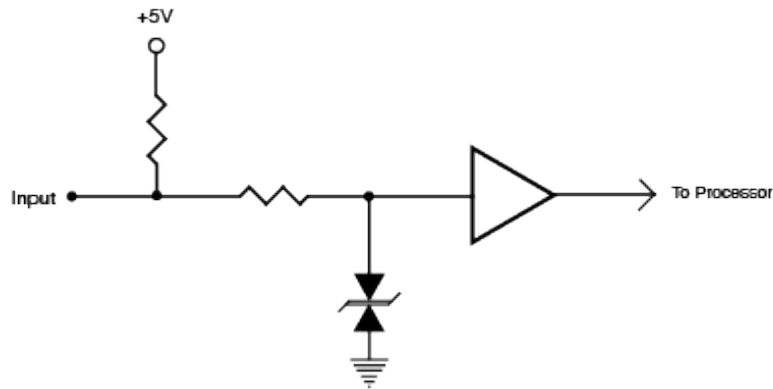
Figure 2.2d - Input Schematic

Connect one terminal of sensor or switch to the input terminal (Input 1 or Input 2). Connect the other terminal to the ground terminal on WebRelay-10 Plus™. When contacts are closed the input is connected to ground and the input is considered to be *ON*. Switch current is less than 2 mA.
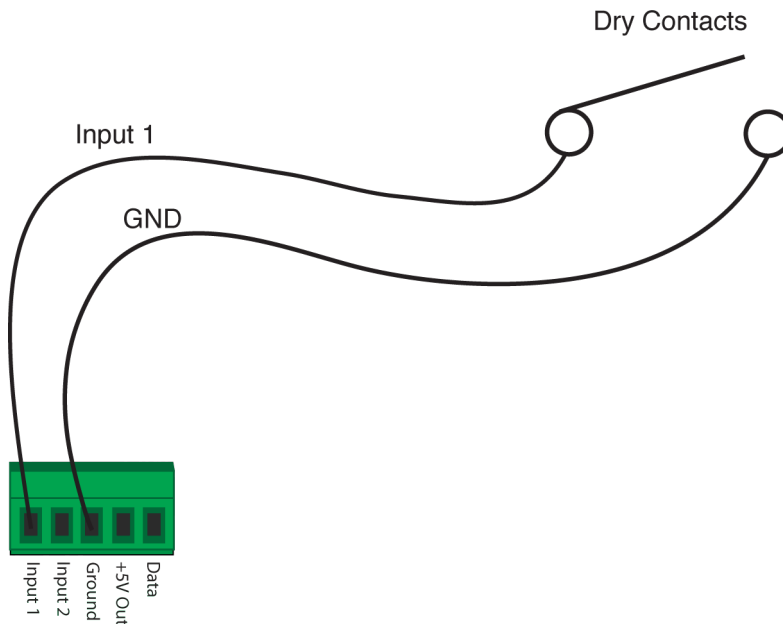
Figure 2.2e - Input Connector

Temperature/Humidity Sensor Connection

Temperature and/or humidity sensors can be used for monitoring environmental conditions. The digital sensors use a one-wire data bus, which allows up to eight sensors to share the same terminals (+5V, Ground, Data). Every sensor on the one-wire bus is assigned a unique serial number when it is manufactured. That number is used to address the device during communication.

The sensors have three wires; the wire color is show in the table below.

Figure 2.2f - Temperature Sensor

Sensor Wire Color	Connection
Red	Vout
Black	Gna
Blue (or White)	Data

Multiple sensors can be connected in two ways, directly connected to the unit (star topology) or "daisy chained" (linear topology) as shown in Figure 2.2g.
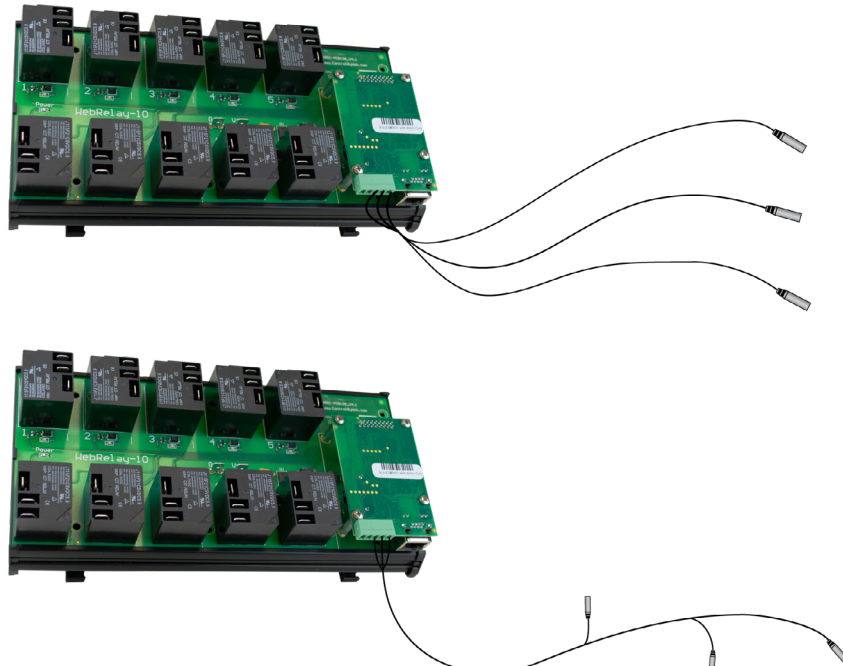
Figure 2.2g - Direct Connection (Star) and Daisy Chain Connection (Linear)

Many factors will determine the maximum length of cable used to connect sensors to WebRelay-10 Plus™. Some of these factors include, but are not limited to, the type of cable used, the number of sensors, ambient electromagnetic noise, and sensor network topology. Combined cable lengths to all sensors of 600 ft using Cat 5e cable have been successful. However, due to the uniqueness of installation environments, results may vary. Please test in the desired environment before permanent installation.

The following are general recommendations that will maximize sensor runs and minimize problems.
Cat 5 and Cat 5e network cable has proven to be an effective and low-cost solution for long runs. Other cable types can be used, but cable capacitance may limit the length. Figure 2.2h shows the recommended connection using Cat 5 network cable.
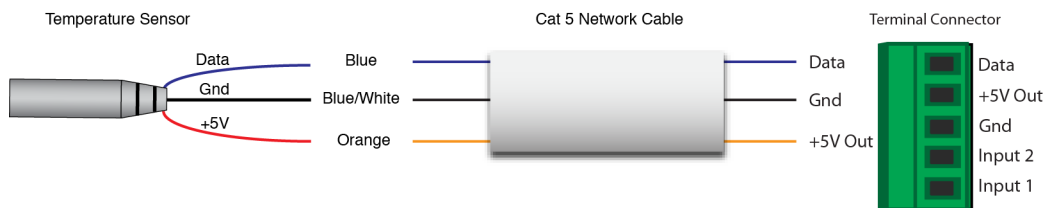
Figure 2.2h - Recommended connection using Cat 5 cable

A linear (daisy chain) rather than a star (direct connection) topology will minimize signal reflections. Appropriate strain relief should be used at the WebRelay-10 Plus™ and other connections that may be subjected to vibration, movement, or repeated handling.

The 1-wire bus is "single-ended" and has no intrinsic noise protection. It is susceptible to interference if the cable is routed near power lines, fluorescent fixtures, motors or other noise sources. Keep the cable wiring short and avoid routing it near other electrical equipment.

Protect any electrical connections with appropriate weather shielding.
Due to the broad range of applications and environments were the WebRelay-10 Plus™ may be employed, installation success on long sensor runs may vary significantly.

2.3 Establishing Communications for Setup

The WebRelay-10 Plus™ is set up using a web browser. The first task is to establish communications between a computer and the WebRelay-10 Plus™ so that the browser-based configuration can begin. To do this, the computer and the WebRelay-10 Plus™ must be physically connected to the same network and both must have IP addresses on the same network. There are two ways to set up the computer and the WebRelay-10 Plus™ so that they are on the same network. The first (Option 1) is to change the IP address of the WebRelay-10 Plus™ to an address that is on the same network as the computer. The second (Option 2) is to change the IP address of the computer to an address that is on the same network that the WebRelay-10 Plus™ is set to by default.

2.3.1 Option 1: Assign a temporary IP address to the WebRelay-10 Plus™

This option is used to TEMPORARILY assign an IP address to the WebRelay-10 Plus™ without the need to change the IP address of the configuration computer. Note that the WebRelay-10 Plus™ will only use this IP address as long as power is maintained. Once power is lost and restored, the WebRelay-10 Plus™ will use the IP address assigned in the setup page and not the temporary address assigned here. This means that once communications are established, the desired IP address should be entered into the Network setup tab using the browser.

To assign the temporary IP address...

1. Make sure the WebRelay-10 Plus™ and the configuration computer are connected to the same network. This will not work through routers or gateways.
2. Assign the address as follows:

<u>Windows</u>:
Open a Command Prompt (select START, then RUN, then type "cmd").

Note: For Vista, the Command Prompt should be run as administrator (select Start, then type "cmd" and right click on "cmd" and select "Run as administrator").

Type:
> *arp -s {new IP address} {serial number of the WebRelay-10 Plus™ }*
> *Note: IP address format is xxx.xxx.xxx.xxx*
> *Serial number format is ss-ss-ss-ss-ss-ss*
>
> For example, to set the WebRelay-10 Plus™ (with serial number 00-0C-C8-01-00-01) to 10.10.10.40 the following command would be used:

```
arp -s 10.10.10.40 00-0c-c8-01-00-01
```

Next, type:
> *ping -l 102 {new IP address}*
> For example, if the new IP address is 10.10.10.40, the following command would be used:

```
ping -l 102 10.10.10.40
```

<u>Linux/Unix</u>:
Open a terminal, change to root user (su -, then enter root password).

Type:

     arp -s {new IP address} {serial number of the WebRelay-10 Plus™ }

     Note: IP address format is xxx.xxx.xxx.xxx
     Serial number format is ss:ss:ss:ss:ss:ss

     For example, to set the WebRelay-10 Plus™ (with serial number 00-0C-C8-01-00-01) to 10.10.10.40 the following command would be used:

```
arp -s 10.10.10.40 00:0c:c8:01:00:01
```

Next, type:

     ping -s 102 {new IP address}

     For example, if the new IP address is 10.10.10.40, the following command would be used:

```
ping -s 102 10.10.10.40
```

<u>Mac OS X</u>:
Open a terminal.
Note: The terminal is in the "Utilities" directory, which is in the "Applications" directory.

Type:

     sudo arp -s {new IP address} {serial number of the WebRelay-10 Plus™ }

     Note: Administrator password is required.
     IP address format is xxx.xxx.xxx.xxx
     Serial number format is ss:ss:ss:ss:ss:ss

     For example, to set a the WebRelay-10 Plus™ (with serial number 00-0C-C8-01-00-01) to 10.10.10.40 the following command would be used:

```
sudo arp -s 10.10.10.40 00:0c:c8:01:00:01
```

Next, type:

     ping -s 102 {new IP address}
     For example, if the new IP address is 10.10.10.40, the following command would be used:

```
ping -s 102 10.10.10.40
```

2.3.2 Option 2: Assign a temporary IP address to configuration computer

If the first option above is not used, you can use this option to communicate with the WebRelay-10 Plus™. By default, the WebRelay-10 Plus™ comes from the factory with an IP address of 192.168.1.2. Communications with the WebRelay-10 Plus™ may be established by assigning an IP address to the configuration computer that is on the same network as the WebRelay-10 Plus™ (for example, the configuration computer could be assigned to 192.168.1.50)

The following example is for those running the Windows operating system:

Step 1: Open the control panel by clicking on the start menu and then clicking on Control Panel (Figure 2.3a).

Note: The control panel shown (Figure 2.3b) is in "Classic View." If the control panel is in "Category View," select the "Classic View" option before proceeding.
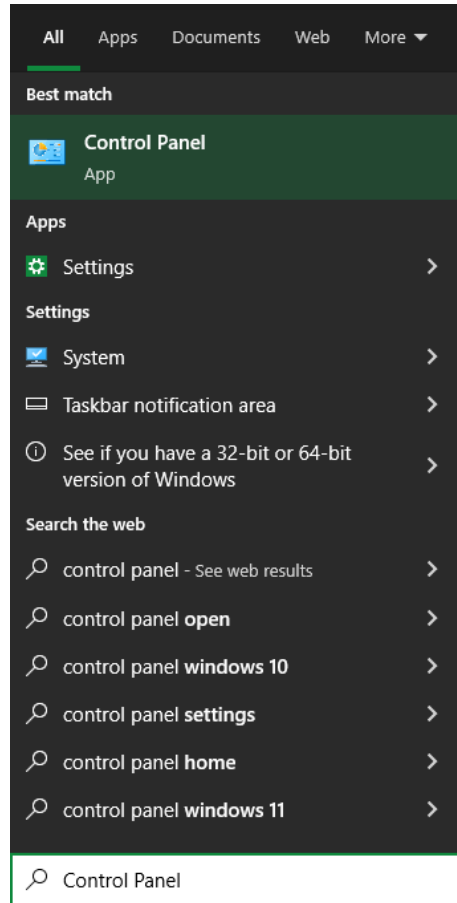
Figure 2.3a - Start Menu
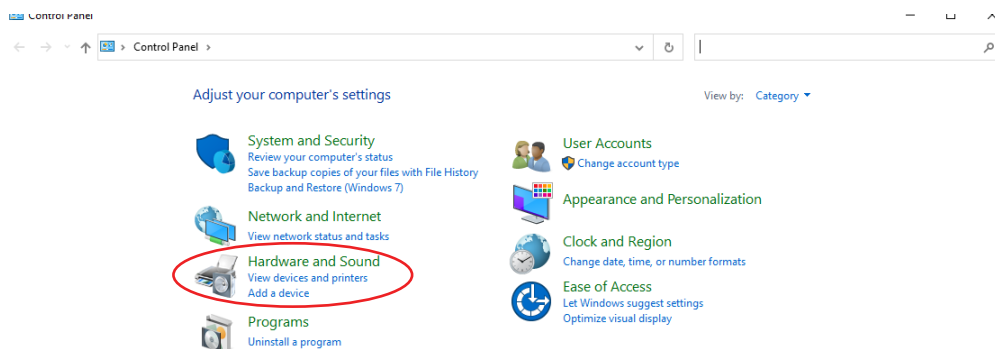
Figure 2.3b - Control Panel

Step 2: Double click on the icon labeled Network Sharing Center. The window shown below will pop-up.

Note: To access the Network Connections folder in Windows Vista, select the Windows Icon (Start Menu) and enter ncpa.cpl into the search bar and press Enter.
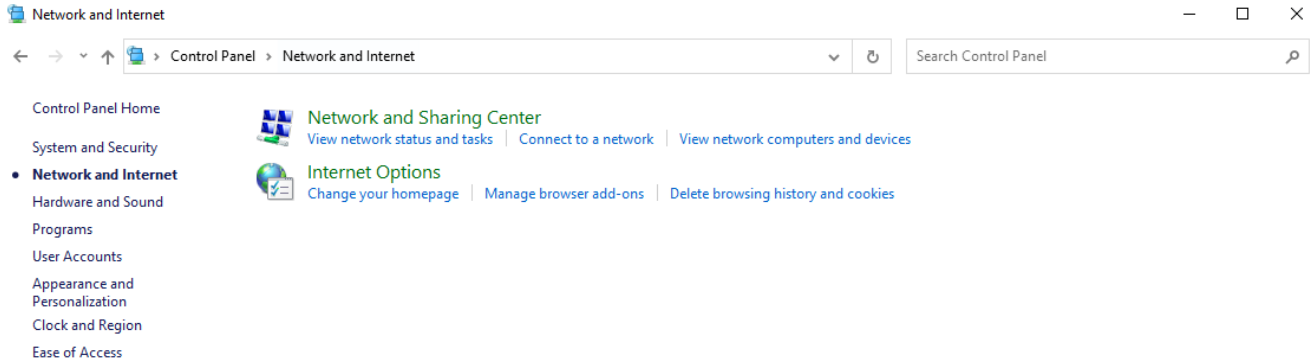
Figure 2.3c - Network and Internet
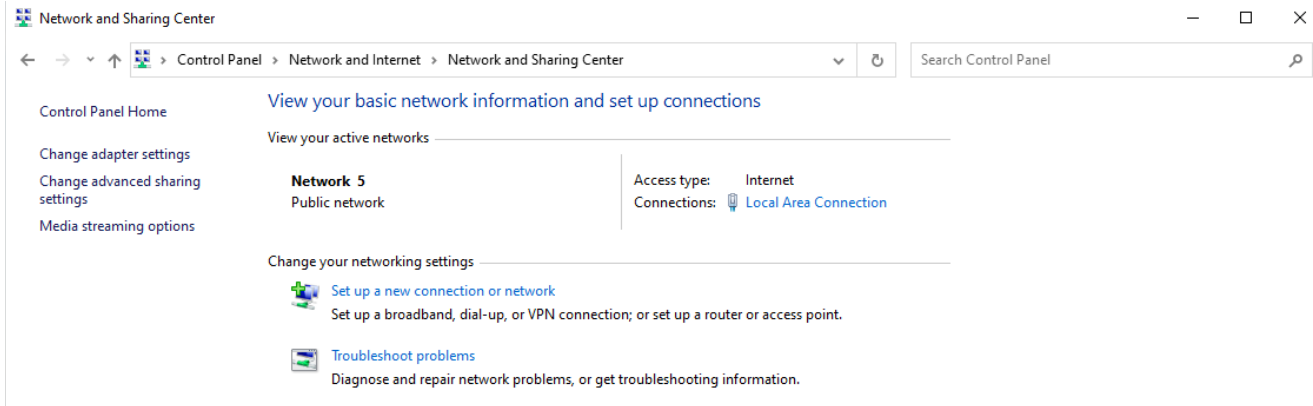
Figure 2.3d - Network and Sharing Center
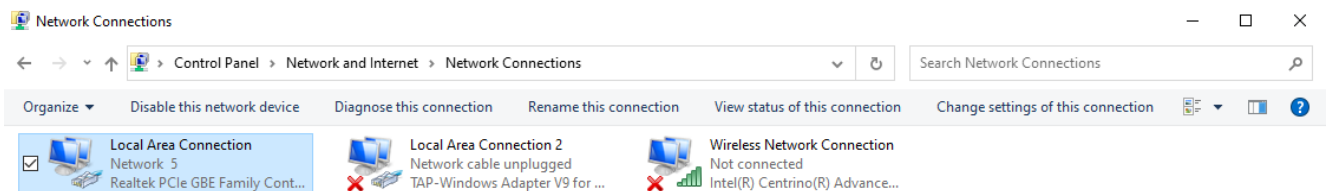
Figure 2.3e - Network Connection

Step 3: Right click on the icon labeled Local Area Connection. In the menu that follows, select the option at the bottom of the menu labeled Properties. The Local Area Connection Properties window will appear. Scroll down to and highlight "Internet Protocol (TCP/IP)," then click the button labeled "Properties."
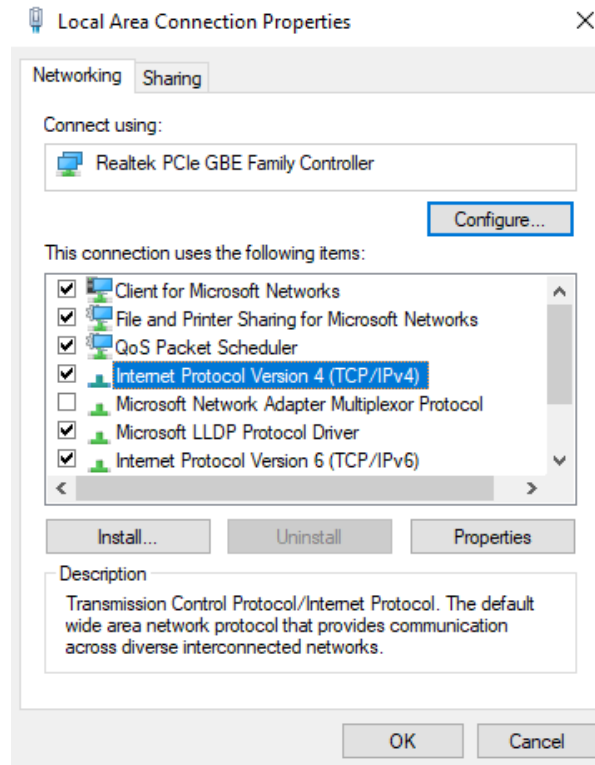
Figure 2.3f - Local Area Connection

Step 4: The following "Internet Protocol (TCP/IP)" Properties menu appears:
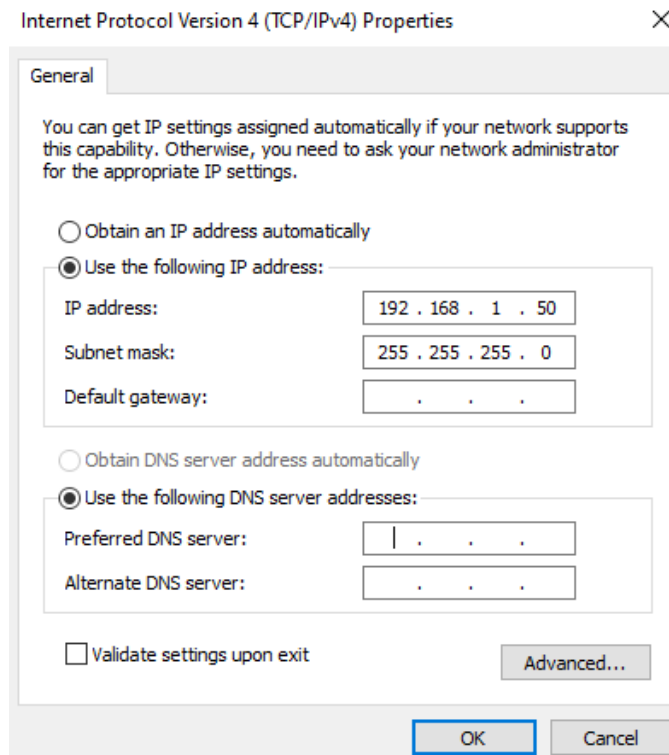
Figure 2.3g - TCP/IP Properties

Step 5: Select the radio button labeled "Use the following IP address" and type in the IP address `192.168.1.50`. Type in a subnet mask of `255.255.255.0`. No need to change the default gateway field. Click OK to accept the new settings.

Note: If "Use the following IP address" is already selected, the values need to be recorded, then reentered once the IP address of the WebRelay-10 Plus™ is successfully changed.

2.3.3 Open Configuration Web Page

Once the network is set up, open the setup page as described in section 2.4. If the setup pages are not accessible, verify that the WebRelay-10 Plus™ is powered on and that the LINK light is illuminated. Check all network connections and settings. Another way to check communications is to ping the WebRelay-10 Plus™ (from the command prompt by typing `ping {WebRelay-10 Plus™ Plus IP address}`).

Section 3: Operation Features

The internal web server presents two classes of web pages; *Setup* pages and *Control* pages. *Setup* pages are used by an installer to provision and configure the module, and a limited version of *Setup* pages can be used by a Manager to create and manager scheduled tasks. The *Control* page allows the local and remote I/O to be monitored and controlled.

To access the setup pages (for `admin` or `manager` login), enter the following URL in the address bar of a web browser:

> `http://{ipaddress}/setup.html`

For example, using the default IP address, enter:

> http://192.168.1.2/setup.html

To access the Control Page, enter the following URL in the address bar of a web browser:

> `http://192.168.1.2`

To access any web pages over an encrypted connection replace http with https. For example:

> https://192.168.1.2/setup.html

> https://192.168.1.2

Before accessing any setup page, the browser will request a username and password. The WebRelay-10 Plus™ modules support various levels of user authentication with different permissions to the setup pages. For a description of *Administrators*, *Managers* and *Users* see General Settings Tab > PASSWORDS.The *Administrator* username is `admin` and the default password is `webrelay` (passwords are case sensitive). For *Manager* access the username is `manager` and the default password is `webrelay`. No password is required by default to access the control page.

The setup pages are divided into six sections. Sections with a ▼ symbol can be expanded to reveal other related settings. <u>When using the Setup Pages, you must click the Submit button at the bottom of a page if you have made changes to a setting on the page.</u>

3.1 Setup Strategy

To configure the WebRelay-10 Plus™ module, follow these basic steps (each of these steps will be discussed in more detail in the next sections):

Step 1: Edit the Network settings to make the module accessible on your network (IP address, gateway and DNS Server IP addresses, etc.). Also setup any email addresses that will be needed for alarms and messages. The device must be power cycled for these settings to take effect.

Step 2: Add any external devices that should be monitored and/or controlled from the module.

Step 3: Name and configure the I/O resources (i.e., relays, inputs, registers, timers, etc.) under the I/O Setup menu.

Step 4: Define any control logic (tasks) or Basic scripts under the Control/Logic menu. Finally, setup and configure the Control Page under Monitor & Control to show the information and control buttons needed for your specific application.

Step 5: View and test the Control Page where you can monitor your local and remote I/O and experiment with any buttons or controls. Test for proper operation of email/text messages and alarms.

3.2 General Settings Tab

The General Settings tab is a collection of menus for configuring the IP network settings, email server settings, device configuration backup and restore features, etc.

3.2.1 General Settings Tab > GENERAL INFORMATION

This is the initial page that is displayed when "/setup.html" is entered into the address bar of the browser. It displays the part number, firmware revision, and serial number of the unit. The optional latitude and longitude settings are used for calculating sunrise and sunset times for Scheduled Tasks. A global setting selects the temperature units. *(Used for remote temperature sensors.)*

Part Number

This displays the full model number of the module.

Firmware Revision

This is the current product revision of the device's firmware.

Serial Number

This is the serial number of the module. The serial number is also the MAC address.

GENERAL SETTINGS	
⚙ GENERAL INFORMATION	
Part Number:	X-400C-I
Firmware Revision:	1.07b
Serial Number:	00:0C:C8:00:00:00
Vin Voltage:	25.8 V
Internal 5V Voltage:	4.98 V
Network Interface In Use:	Ethernet
ⓘ Latitude:	41.6796
Longitude:	-111.8737
Temperature Units:	Fahrenheit Celsius Kelvin

Submit   Cancel

Vin Voltage

This is the voltage on the +Vin power input terminal. If the module is powered from a battery, this value will show the battery voltage.

Internal 5V Voltage

This diagnostic displays the voltage of the internal 5V power rail. It should be 5.0V ±.25V.

Latitude & Longitude

The optional latitude and longitude settings are used for calculating sunrise and sunset times for Scheduled Tasks. This is entered in degrees.decimal degrees format (does not use minutes and seconds). North Latitude is positive and South is negative. East of the prime meridian is positive and West of the prime meridian is negative.

Temperature Units

This global setting sets the temperature units of Fahrenheit, Celsius, or Kelvin for all temperature sensors, both local and remote *(Some ControlByWeb devices do not have a 1-Wire bus for local temperature sensors; however, they do support remote 1-Wire sensors).*

3.2.2   General Settings Tab > NETWORK SETTINGS

The network parameters are set on this page. Configure the network settings to make the WebRelay-10 Plus™ module accessible on your network. **The module must be power-cycled (power disconnected, then reconnected) before new network settings take effect.**

Use DHCP (Dynamic Host Control Protocol)

This option allows DHCP to be enabled or disabled. If this option is set to Yes, the module will request an IP address from the DHCP server each time it is powered on. The default setting is No (this is recommended for most installations). If DHCP is set to Yes, the Network page must be submitted and the module must be rebooted before an IP address will be assigned. Once the module is assigned an IP address by the DHCP server, the new IP address can be found through the list of clients kept by the DHCP server.

Brief Notes About DHCP: All devices on an IP network require an IP address. This is a unique address that identifies each device on the network. DHCP is a mechanism that automatically assigns an IP address to a computer (or other device) when it is connected to a network. This eliminates the need to manually enter the IP address. When a computer is connected to the network, another device on the network called a DHCP server detects the presence of the computer or device and dynamically assigns an IP address. On many small networks, the DHCP server is built into the router.

DHCP works well for "client" devices such as computers, but is not ideal for servers. This is because servers usually don't initiate communications with other devices, but rather they wait for a request from "clients." To make this request, the client must know the IP address of the server. If a server gets its IP address dynamically, the IP address may not always be the same so client devices may not be able to find the server. For this reason, servers usually use an IP address that is fixed and does not change. The module is a server and manual IP address assignment is usually recommended.

IP Address

Enter the IP address for the module in this field. The IP address is specific to the network where the module will be installed, and must be obtained from the network administrator. For more information on IP addresses and remotely accessing the module over the Internet, see *Appendix C: Accessing the Device Over the Internet.* The default setting for this field is: 192.168.1.2

Subnet Mask

The subnet mask defines the size of the local network. This can be obtained from the network administrator. For additional information about sub-netting and IP networking, many tutorials are available on the Internet. The default setting for this field is: 255.255.255.0

Gateway

This specifies the IP address of the gateway router. This can be obtained from the network administrator. The default setting for this field is: 192.168.1.1

Preferred DNS Server:

The IP address of the Primary DNS server is specified here. When DNS services are required, this is the address that will be used. The default setting for this field is 192.168.1.1

This field is only required when the following options are used:

Remote Services	When server is specified by name and not IP address.
Sync time clock with remote NTP server	When server name is specified by name and not IP address.
Email Server	When server name is specified by name and not IP address.
SNMP Manager	When the server for receiving traps and notifications is specified by name and not an IP address.
Remote Devices	When remote device (such as for use as a remote relay) is specified by name and not IP address

Alternate DNS Server

This field is used to specify the IP address of a Secondary DNS server. This is used when the module requires DNS services and the preferred DNS server is not available. The default setting for this field is 192.168.1.1

HTTP Port Enabled

This option enables or disables access to the web server without encryption. For high security applications the HTTP port should be disabled to limit access to only encrypted connections on the HTTPS port.

HTTP Port

The TCP port used for unencrypted HTTP communications with the module is specified here. The default setting for this field is 80, which is the standard HTTP port. It is recommended that the port be left unchanged unless the user has an understanding of TCP/IP and ports. For more information on TCP ports and IP addressing see *Appendix C: Accessing the Device Over the Internet.*

HTTPS Port

The TCP port used for encrypted HTTPS communications. It is recommended not to change this port. When requesting a web page using https://192.168.1.2/setup.html, the web browser will automatically use port 443. If this port is changed to 9000 for example, the HTTPS port will have to be specified in the request https://192.168.1.2:9000/setup.html

Upload/View SSL Certificate

By default, the module comes with a previously generated SSL Certificate that is used for encrypted HTTP communications. The default SSL Certificate can be replaced by clicking on this link, choosing the new Certificate file (PEM format), and uploading the file. SSL Certificates are preserved, even when resetting defaults and updating firmware.

Note: Supports 1024-bit and 2048-bit encryption.

Upload/View SSL Key File

By default, the module comes with a previously generated SSL Key that is used for encrypted HTTP communications. The default SSL Key can be replaced by clicking on this link, choosing the new Key file (PEM PKCS#1 format), and uploading the file. SSL Keys are preserved if the firmware is updated or the module is reset to its default settings.

There are various programs that can be used to generate self-signed certificates and keys. When generating the key, the program must either generate the key in PKCS#1 format, or provide a way to convert it to the PKCS#1 format in order for the module to use it.

One such program is openssl. Newer versions of openssl generate keys using the PKCS#8 format. Older versions generated keys using the PKCS#1 format. A visual inspection of the key file can be a good indicator of the format of the key. PKCS#1 keys start with " -----BEGIN RSA PRIVATE KEY-----". PKCS#8 keys start with "-----BEGIN PRIVATE KEY-----". The following command can be used to generate a key and certificate for use with the module:

```
openssl req -newkey rsa:1024 -nodes -keyout domain.key -x509 -days 365
-out domain.crt
```

If the format of the key is PKCS#8, the following command will convert it to PKCS#1:
```
openssl rsa -in domain.key -out domain_PKCS1.key
```

After executing the previous commands a key named domain_PKCS1.key and certificate named domain.crt will exist.

3.2.3   General Settings Tab > ADVANCED NETWORK SETTINGS

CORS Origin

CORS stands for Cross-Origin Resource Sharing. By default web-pages rendered in a web browser can only request other web-pages from the same server, not a server at another domain or IP. This is a security feature. To allow such requests to a WebRelay-10 Plus™ device, enter the protocol, domain, and port of the remote server that should be given access. For example, to allow the www.ControlByWeb.com site direct access to web pages on the WebRelay-10 Plus™ device, enter http://www.ControlByWeb.com in the CORS Origin field. If using HTTPS instead of HTTP, enter https://www.ControlByWeb.com.

Leaving this field blank, the default, will disable cross origin web requests to the device. Entering an asterisks (*) will allow any remote service to send direct requests for resources.

MTU

MTU is the Maximum Transmission Unit network parameter. This defines the max size, in bytes, of the TCP packets sent out from the device. The valid range is 256 to 1476 bytes. This normally can be left alone, but there are some circumstances where it might be beneficial to change it. One of these circumstances is when the device is to be used over a VPN (virtual private network). VPN's add extra information to TCP packets, if the new packets are too big to physically travel across the network (greater than about 1500 bytes) then the packets will be split up. This causes problems for some firewalls and those firewalls will discard the packets. To fix this, the MTU can be adjusted until the TCP packets do not get split up. Some cellular data networks use smaller packet sizes and MTU adjustment may be necessary.

MODBUS (TCP/IP Slave)

MODBUS/TCP slave service can be enabled or disabled. This setting is ONLY used to enable/disable Modbus TCP/IP slave functionality only and does NOT control the RS-485 Modbus master functionality on the WebRelay-10 Plus™ module. Modbus is a messaging structure protocol used in industrial control and automation. It is an open protocol and offers interoperability with software and devices from other manufacturers. This is enabled by selecting Yes in this field. The default setting for this field is No. (See Section Modbus/TCP (slave) for more information on using the WebRelay-10 Plus™ module on a Modbus network.)

Note: Modbus communications are disabled whenever the User password is enabled because Modbus/TCP does not provide a mechanism for password protection.

The WebRelay-10 Plus™ module functions as a Modbus slave. Host devices, such as PLCs, open a connection with the module on port 502 and then sends requests to read or set I/O states, or sensor values. When the module receives a command, it performs the desired function and returns a response.

Modbus Port

This specifies the port used for Modbus/TCP communications with the module. By default this is set to port 502 which is the standard Modbus port. It can be set within the range of 1 to 65535.

Endianness

32-bit data is treated as two individual, 16-bit words using IEEE 754 floating point format. Floating point format is used for all analog sensors, both local and remote. If *Big* is selected, the module will use big-endian architecture, and the most significant 16-bit word (big end) is sent first. If *Little* is selected, then the module will use little-endian architecture, and the least significant 16-bit word (little end) is sent first. The default setting is *Little* (little-endian). For example, in little-endian format, a 32-bit floating point number represented by '1234 ABCD' is sent as 'CDAB 3412'.

Modbus Address Table

The module has a default Modbus address table with addresses for the local I/O and Vin. As resources such as Registers, and remote I/O are added, additional Modbus addresses are automatically created. Use this tool to view the current Modbus address assignments. The addresses are assigned when the I/O is added and cannot be customized except by deleting the I/O and changing the order in which they are added.

To interpret the Modbus Address Table, find the I/O resource in the left-most column. Then follow the line across to the column of interest. All I/O resources have a corresponding holding register pair for reading the values. This holding register address is listed in the *IO* column of the table and is interpreted as a IEEE-754 floating-point value. The number in the cell is the Modbus address for the given data type (or starting address in the case of floating point numbers). On the module, all holding registers are read and written as pairs. This is because Modbus registers are 16-bits wide and the data types for reading the registers are floating point and require 32-bits. The number listed in the table is the starting address and the corresponding register immediately follows it.

This example table shows the Modbus addresses of the I/O resources and a configured register.

IO Name	IO	Register Addr Pulse Timer	Counter	On Timer	Total On Timer	Coil Addr IO	Input Addr IO
				Register Addr		Coil Addr	Input Addr
IO Name	IO	Pulse Timer	Counter	On Timer	Total On Timer	IO	IO
Analog Input 1	0	--	--	--	--	--	--
Analog Input 2	2	--	--	--	--	--	--
Analog Input 3	4	--	--	--	--	--	--
Analog Input 4	6	--	--	--	--	--	--
Analog Input 5	8	--	--	--	--	--	--
Analog Input 6	10	--	--	--	--	--	--
Analog Input 7	12	--	--	--	--	--	--
Analog Input 8	14	--	--	--	--	--	--
Vin	16	--	--	--	--	--	--
Register 1	18	--	--	--	--	--	--

This images shows Modbus address examples for the X-418. Other products tables will differ depending on the device's I/O.

Remote Services

Remote Services allows internet access to a WebRelay-10 Plus™ device which is installed behind a network router, without the need to setup port forwarding in the router. Typically, when Remote Services is enabled it is used to access the device through our ControlByWeb.cloud service rather than connecting directly to the device.

Remote Services has three main benefits:

First, the web server does not need to know the IP address of the WebRelay-10 Plus™ module. This means that the module can get its IP address dynamically from a DHCP server, simplifying the installation.

Second, since the connection from the module is outgoing, rather than incoming, the local router on the network where the module resides does not need to be configured to forward ports. This simplifies the installation. Since the router configuration is not modified, the risk of compromising security on the local network is eliminated. For more information about the Remote Services see *Section Using an External Web Server.*

Third, connecting to ControlByWeb devices through the ControlByWeb.cloud service allows users to take advantage of additional features and services that are offered on the cloud service such as custom user interfaces, viewing multiple devices at that same time, and providing alerts to the user when the connection goes down.

Enable

Remote Services can be enabled or disabled. If Yes is selected, Remote Services will be enabled as soon as the Submit button is pressed and the WebRelay-10 Plus™ module will immediately attempt to make a connection with the remote server (power cycle not required). Once a connection is established, the connection will remain until it is disconnected by the remote server. By default, Remote Services is configured to connect to the ControlByWeb.cloud service. The default setting for this field is No.

Version

Previous ControlByWeb products supported version 1 of remote services. Choosing version 1 will allow WebRelay-10 Plus™ devices to connect to third party remote services servers. Version 2 of remote services adds some enhancements to the remote services protocol including encryption and client/server authentication. Version 2 is used with the ControlByWeb.cloud service.

Certificate Request Method

Remote Services version 2 use certificates and keys to provide a secure connection between the WebRelay-10 Plus™ modules and the server. Certificates are created on the ControlByWeb.cloud server and must

be transferred to the module for secure communications. There are two methods that can be used to retrieve the certificates and key from the ControlByWeb.cloud service. First, the username and password of the ControlByWeb.cloud service account can be used. Second, a temporary token can be generated on the ControlByWeb.cloud server by the account holder and copied to the module. If a username/password or certificate request token is present when the setup page is submitted, the module will request the certificate, ca, and key from the cloud server. Once the certificate, ca and key are successfully retrieved, the username/password and certificate request fields will be cleared out so those fields will be blank.
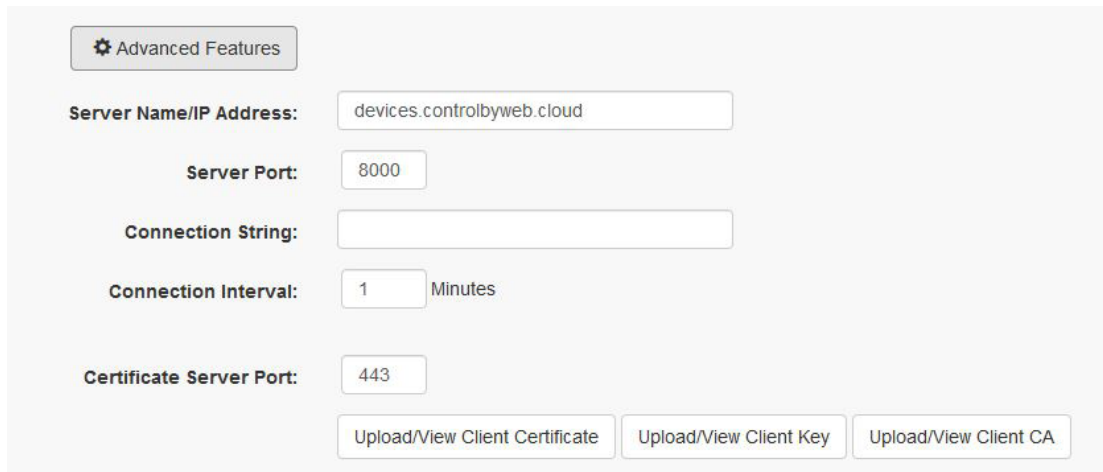
Cloud Username/Password
To obtain the certificate using the method, enter the username and password of the cloud service account.

Certificate Request Token
This method is useful for installations where WebRelay-10 Plus™ products are installed by someone other than the owner of cloud account. In this case, the owner of the cloud account can generate a Certificate Request Token from the ControlByWeb.cloud web site. That token can be copied, and given to the installer rather than the username and password to the account. The installer would enter the token in this field on the WebRelay-10 Plus™ module to setup the cloud connection. The installer can use this token for a fixed time period (12 hours). The installer will be able to connect modules to the account but he/she will have no access to the account.

Advanced Features

The following features do not generally need to be changed, especially when using the device with the ControlByWeb.cloud service. They can be changed if the device is used with a custom service.

Server Name/IP Address
Specify the name or IP address of the Remote Services server here. If the IP address is specified, enter it in this format aaa.bbb.ccc.ddd. For numbers that are less than 100, preceding zeros should not be included (for example, enter 80 rather than 080). This field can be up to 40 characters long, the default setting is: *devices. controlbyweb.cloud*

Server Port
Enter the TCP port used for the Remote Services server. This can be set within the range of 1-65535. The default setting for this field is 8000.

Connection String
This text is sent to the Remote Services server when the connection is established. This string

---

should include any information required by the server for connection. For example, it may include an ID number, customer number, password, etc. The format is entirely dependent upon the server requirements. This field can be up to 80 characters long. By default this field is left blank, as the ControlByWeb cloud service does not use it.

Connection Interval
This field specifies the periodic interval in which the WebRelay-10 Plus™ module attempts to connect to the remote server, or if the module is already connected, it is the interval in which it sends the connection string followed by the current state of the device. This field can be set within the range of 1 to 60 minutes. The default setting for this field is 1 minute.

Certificate Server Port
The certificate server is a server that has been configured to deliver the Client Certificate, Key, and CA when requested by the WebRelay-10 Plus™ module. These certificates and key are required for the module to connect to the cloud service. This field specifies the port used to communicate with that server.

Upload/View Client Certificate
Uploads a client certificate that will be used for authenticating the WebRelay-10 Plus™ module to the cloud service.

Upload/View Client Key
Uploads a client key that will be used for encrypting the module's communications with the cloud service.

Upload/View Client CA
Uploads a CA that has been used to generate the client certificate and key. The WebRelay-10 Plus™ module will use this certificate to verify that the server it has connected to is is the server that it expects to connect to.

SNMP AGENT

The SNMP agent is the server running on the WebRelay-10 Plus™ module responsible for receiving SNMP requests and returning SNMP responses. SNMP Managers are remote servers that the module can send SNMP Trap and Notification messages to. The Agent is always enabled when SNMP is enabled. The SNMP Managers can be enabled/disabled separately. When using SNMP V3, the module supports the User-based Security Model (USM).

SNMP Enabled

SNMP (Simple Network Management Protocol) can be enabled or disabled. The default setting for this option is *No*. (See *SNMP* at the end of this section for more information.)

SNMP Version

The module supports versions V1, V2c, and V3. This option allows choosing what version of SNMP will be used, and will determine what SNMP options are presented. The module will only respond to SNMP requests that use the selected SNMP version.

Agent Settings:

Agent Port

When SNMP is used, this field is used to specify the SNMP port that the module's SNMP Agent listens on. The default setting for this field is 161.

Agent Read Community

The read community string is used for SNMP V1 and V2c requests. It is required to read I/O on the device using SNMP V1 or V2c.

Agent Write Community

The write community string is used for SNMP V1 and V2c requests. It is required to write I/O on the device using SNMP V1 or V2c.

Notification Retries

When sending notifications, this field defines how many attempts the module will make to send the notification to the SNMP Managers. Notifications differ from Traps in that a response is expected back from the SNMP Manager.

Notification Timeout

When sending Notifications, this field defines how many seconds the module waits for a response to a previously sent notification before attempting to send the notification again.

Manager Settings:

Manager Enable

An SNMP Manager is the server intended to receive traps and notifications from the module. The module supports sending traps and notifications to two different SNMP managers. The default setting is *No*.

Manager Hostname/IP

This field is used to specify the hostname or IP address of the SNMP manager. The default setting for this field is 192.168.1.15

Manager Port

This field is used to specify the SNMP Trap or notification port of the SNMP manager. The default setting for this field is 162.

Manager 1 Community

This field is used to defined the Trap/Notification community string used by the SNMP Manager for SNMP V1 and V2c. The SNMP manager will not accept the Trap/Notification without a correct community string.

MIB File

This button generates the Management Information Base (MIB) used for managing the entities in a communication network associated with the SNMP protocol. The file will be automatically generated and downloaded. This file will change as the I/O configured on the device changes or as the SNMP version changes.

SNMP V3 Security Settings:

The following settings appear when the SNMP version is set to Version 3. These settings configure the authentication and privacy protocols used by SNMP V3 User-based Security Model (USM). The Agent, and both SNMP Managers each have their own set of USM settings. When SNMP V3 is selected, the community string settings disappear and are not used. They are replaced by the following settings.

Username
This field defines the security username.

Auth Protocol
This field defines the authentication protocol used: None, MD5, SHA, SHA224, SHA256.

Auth Password
This field defines the authentication password.

Priv Protocol
This field defines the privacy protocol used: None, AES128, AES192, AES256.

Priv Password
This field defines the privacy password used.

SNMP Notes: Simple Network Management Protocol (SNMP) is used to manage and administer network devices. The module supports SNMP V1, V2c, and V3 and can be configured here. Using SNMP, the I/O states of the module can be read as well as some basic information about the module. See *Section SNMP Requests, Objects and Community Strings* for information about how to request information from the module using an SNMP manager.

IP Filtering

IP filtering can be enabled or disabled. IP filtering is used to restrict incoming network connections to only specific IP addresses. If a connection comes in to the module that is not in the range of allowable IP addresses, the connection is terminated. The IP filter only applies to incoming connections and not to outbound connections such as those to DNS servers. The module allows for two different ranges to be defined. If one of the ranges is not needed, the IP address in each of the three fields for a given filter must be 0.0.0.0. IP filter settings only take effect after the module is power-cycled.

⊕ IP FILTERING

Enable:	Yes / No

The following network settings will require reboot to take affect.

Filter 1 Range:	192.168.1.1
	192.168.1.254
Filter 2 Range:	0.0.0.0
	0.0.0.0

Filter Range

The IP addresses within the defined filter range are allowed to communicate with the module. IP addresses outside the ranges defined are rejected. The first field is the start address, the second field is the end address.

3.2.4   General Settings Tab > EMAIL SETTINGS

Email Notification Description

The module can be configured to send messages to up to eight email addresses when certain events occur. Tasks to send email messages when events happen are set up under under the *Control/Logic Tab > TASK/ FUNCTIONS* tab. Events that can trigger email messages include local and remote I/O changes, Vin changes, and much more. In addition, email messages can be sent under the command of BASIC scripts.

When an email message is sent, it looks similar to the the example message below:
Note: The messages will vary depending on the ControlByWeb device (X-410, X-418, X-420, etc.).

```
X-418 - Trigger: Condition 1 Task

Analog Input 1: 4.2 V
Analog Input 2: 2.5 VDC
Analog Input 3: 1.00 VAC
Analog Input 4: 0.001 Amps
Vin: 12.00 V
Time: 09/27/2017 15:30:00
```

Note: The sensor names as well as ON and OFF status text may be customized by the user/installer in the Setup Pages. The I/O names and and ON/OFF status text are assigned in the I/O setup section and are used both for the email messages and the Control Page.

Subject Line

The top line (in this example it reads "X-418 – Trigger: Condition 1 Task"), appears in the subject line of the email message. The text on the left (in the example "X-418") is the same text that appears as the header on the Control Page. It is set in the *Main Header Text field* under the *Monitor & Control Tab > CONTROL PAGE SETUP* tab.

To the right of the *Main Header Text*, the word "Trigger:" appears and this cannot be changed.

To the right of the word "Trigger:", the name of the task which triggered the email message will appear. In the example, the task is called "Condition 1 Task" but tasks can (and should) be given custom names that make more sense to users. Tasks which can trigger emails are Scheduled, Conditional, Automatic Reboot, Test Email, or Basic Script. Tasks are configured under the tab *Control/Logic Tab > TASK/FUNCTIONS*.

Current Status of the Analog Inputs, Vin, and Sensors

The remainder of the email message will display the same information shown on the Control Page. Fields not displayed on the Control Page will not be included in the email message. Displayed fields may be configured in the *Monitor & Control Tab > CONTROL PAGE SETUP* page. Alternatively, the email message can be set to "Full" or "Short" on the Email tab of the Setup pages. Setting the Email Message to "Short" will only include the subject line, which may be more appropriate for alerts that are converted to SMS text messages.

Important Note: Notification email messages are limited to 32 I/O. If the Control Page is configured to display more than 32 I/O, email notifications will contain the first 32 I/O that are displayed on the Control Page.

The Email parameters are set on this page.

SMTP Server

The name of the SMTP (Simple Mail Transfer Protocol) mail server (i.e., mail.example.com) or the IP address of the mail server (i.e., 192.10.10.10) should be entered in this field. There is no default setting for this field.

Note: If the server name is entered and not the IP address, the address of a DNS server will be required in the DNS field.

Connection Security

Select the security method used for sending the email. There are two methods used to securely transmit email messages. STARTTLS and TLS/SSL. When STARTTLS is chosen, encryption will begin after the module makes an unsecured connection to the SMTP server and negotiates TLS. When TLS/SSL is chosen, the connection to the server will be securely negotiated with the SMTP server from the beginning. Both methods are secure. STARTTLS generally requires the use of port 587 and TLS/SSL generally requires the use of port 465. No security method is also an option, and still supported by some SMTP servers. This method generally uses port 25.

Server Port

This field is used to specify the SMTP Mail Server Port. The default setting is 25, which is the standard SMTP port. This port will generally be different if a connection security method is chosen.

User Name (If Required)
If the SMTP mail server requires authentication, the user name must be entered here. There is no default setting for this field.

Password (If Required)
If the SMTP mail server requires authentication, the password must be entered here. There is no default setting for this field.

Return Email
The module itself will not receive email messages, but when the module sends email messages, it must include a return email address. This field is used to specify the return email address. Note that although the module will send email messages to any email address specified in this field, some email filters (spam filters) will not allow messages through that include an invalid email address. There is no default setting for this field.

Email 1 to Email 8
Enter the email addresses of up to eight recipients. Email notifications will not be sent until tasks are created to send them.

Email Message
Choose either "Full" or "Short" email formats. Full sends email messages with all visible fields on the Control Page. Short sends messages only showing what triggered the email.

Send Test Email
This button sends a test email to the first email address in the email list using the currently configured network and email settings. By pressing this button, the settings are automatically submitted, and then a progress window will pop-up and display the result of the test email:

- Success
- Failed DNS Lookup
- No Response from DNS server. Check DNS addresses, Power Cycle Device, etc.
- Failed
- Bad Username/Password
- Missing recipient email address(es)
- Server Address/Hostname does not exist or is misspelled
- Server Address/Hostname is blank
- No Response from SMTP server. Check server address and security/port combination

3.2.5 General Settings Tab > PASSWORDS

The module requires passwords to access specific resources. The passwords can be changed on this page. Passwords must be 6 to 18 characters, both alphabetic and numeric characters are recommended. A hide/show check-box selects if the password will be shown. When a password is changed, it must be entered twice for verification. If the password is not entered identically in both fields, the password will not be changed.

Administrator

Administrators have access to all Setup, Task, and Control pages. This access privilege is normally used by a system integrator or installer to setup I/O, control logic and user interface. The Administrator username is admin (all lower case). The default password is webrelay (all lower case). To access these pages go to http://{ipaddress}/setup.html

Manager

Managers have access to Tasks and Control pages. This access privilege can be used for example, to allow a door access schedule to be changed for a holiday. A Manager cannot change network settings, I/O, or control logic. The Manager username is manager (all lower case). The default Manager password is webrelay (all lower case). Manager access can be enabled or disabled. When a manager logs into the Setup pages, they will receive a subset of the menu options allowing them to edit Scheduled Tasks and view the Control pages.To access these pages go to http://{ipaddress}/setup.html

User

Users have access privilege to the Control Page only. This access privilege is for users and operators to monitor sensors or control outputs that are on the Control Page. The password access for Users can be enabled or disabled. When this field is set to Yes, a password will be required to view the Control Page. The default setting for this field is No. The User username is user (all lower case). The default User password is webrelay (all lower case). To access these pages go to http://{ipaddress}

Note: Since Modbus has no provision for passing passwords, Modbus will be disabled if the Control Page password is enabled.

Device PSK

The Device PSK is a pre-shared key used by other ControlByWeb devices when communicating with the module securely. When adding a remote ControlByWeb device to a module, that device's preshared key is asked for during the configuration. This is where that PSK is defined. Note that not all ControlByWeb devices support use of PSK.

The PSK is a 32-byte hexadecimal key with valid characters being a-f and 0-9. Anything can be used as the devices PSK as long as it is 32 bytes long and contain only valid characters. The Generate PSK button can be used to request the module to generate a secure, random PSK. This is the preferred way to generate the PSK. Once generated, a new PSK need not be generated unless the old one has been compromised. A new PSK is generated when the device is reset to factory defaults.

3.2.6 General Settings Tab > DATE & TIME

The module uses the time of day for scheduled events, such as turning local/remote I/O on or off at scheduled times, and for logging other local/remote I/O (a time stamp is included with each logged event). The time is stored and displayed in 24-hour time format. The module has a capacitor-backed, real-time-clock circuit that will sustain the time for several days in the event of a power failure.

Current

This is the current date and time maintained in the module. The time is stored and displayed in 24-hour format.

Set:

This drop-down list offers two options for setting the time: *Manually* or *Sync with NTP server.*

The options that follow this field will change based upon how this option is set.

Manually	Requires the user to enter the time and date.
Sync with NTP server	Allows the user to set the clock automatically by using an NTP (Network Time Protocol) server.

Manual Time Configuration

Date
The current date is entered by selecting the month, day, and year using the drop-down boxes.

Time (24 Hour Format)
Enter the time as HH:MM:SS. (HH represents hours in 24-hour format [00-23], MM represents minutes [00-59], SS represents seconds [00-59].)

UTC Offset
World time zones are defined as an offset from Universal Time (GMT). It is common for many servers and data loggers to use GMT as their official time, even when they are not located within the GMT time zone. The default value for this field is -7 (Mountain Standard Time). For convenience, the time can be converted to local standard time by entering the offset here. This manual cannot include the UTC Offset for all parts of the world, but the offset for GMT time and the four major US

Time zones are listed here.

> GMT Time: 0
> Eastern Standard Time: -5:00
> Central Standard Time: -6:00
> Mountain Standard Time: -7:00
> Pacific Standard Time: -8:00

SYNC With NTP Server

NTP Host Name
This field is used to specify the name or IP address of the NTP server. If a name is specified, a working DNS server address must be entered into the Network settings. If the IP address is specified, it should be entered in the following format aaa.bbb.ccc.ddd where each of the letters represents a number between 0 and 255. This field can be up to 40 characters. There is no default value for this field.

Many NTP Internet servers are available. In addition, many desktop computers will function as an NTP server (both Mac and PC). If a desktop computer is used, firewall settings may need to be adjusted to allow for NTP communications on UDP port 123.

Public NTP servers can be found at www.pool.ntp.org. Some of these are listed below.
    US Servers (http://www.pool.ntp.org/zone/us):
      0.us.pool.ntp.org
      1.us.pool.ntp.org
      2.us.pool.ntp.org
      3.us.pool.ntp.org

    North America (http://www.pool.ntp.org/zone/north-america):
      0.north-america.pool.ntp.org
      1.north-america.pool.ntp.org
      2.north-america.pool.ntp.org
      3.north-america.pool.ntp.org

    Europe (http://www.pool.ntp.org/zone/europe):
      0.europe.pool.ntp.org
      1.europe.pool.ntp.org

2.europe.pool.ntp.org
3.europe.pool.ntp.org

Australia (http://www.pool.ntp.org/zone/au):
0.au.pool.ntp.org
1.au.pool.ntp.org
2.au.pool.ntp.org
3.au.pool.ntp.org

South America (http://www.pool.ntp.org/zone/south-america):
0.south-america.pool.ntp.org
1.south-america.pool.ntp.org
2.south-america.pool.ntp.org
3.south-america.pool.ntp.org

Africa (http://www.pool.ntp.org/zone/africa):
0.africa.pool.ntp.org
1.pool.ntp.org
2.pool.ntp.org

NTP Sync Interval
This option allows the user to specify how often the time on the module will be synchronized with1 the time server. When the Submit button on this page is pressed, the module will immediately synchronize with the time server. If Daily, Weekly, or Monthly options are selected, the module will thereafter re-synchronize with the time server at the period interval specified starting at 12:00 AM (00:00).

To prevent multiple ControlByWeb devices from overwhelming the NTP server at power on, the exact time the NTP Request occurs is 12:00 AM (00:00) plus the minute equivalent of the last two digits in the models serial number. For example, if the last two digits in the model's serial number were -09, the NTP Request will occur 9 minutes after 12:00 AM. The default value of this setting is Once (the unit will immediately sync with the NTP server, but will not automatically sync again).

Sync on Power Up
When this option is set to Yes, the module will be synchronized with the time server each time it is powered.

Note: If the module will lose power on a frequent basis, it may be beneficial to set this option to No. Some servers are configured to dis-allow access from client devices that excessively request their services. The default value of this setting is No.

Daylight Savings

Enable
Daylight Savings can be enabled or disabled. The default setting is Yes.

In many parts of the United States and in some other countries, the time is shifted forward by one hour during the summer months. This is an effort to conserve energy by making the daylight last longer into the evening hours. If this option is set to Yes, the time on the module will automatically be shifted forward by one hour between the hours of 12:00 AM – 5:00 AM on the Daylight Savings Start date set below, and it will shift back to standard time between the hours of 12:00 AM – 5:00 AM on the Daylight Savings End date set below. The time change is made at a random time within the previously mentioned, five-hour time frame, in order to prevent multiple devices from simultaneously requesting a time and overwhelming the NTP server.

Note: When daylight savings time adjustment is enabled, Scheduled Tasks will be adjusted for the new time. Logged data includes a time stamp based upon the current time in the device followed by DST when the device is in daylight savings and STD when it is not. To avoid confusion, many servers and data loggers are set to remain on GMT time and do not shift for daylight savings.

Daylight Savings Start Date/Time
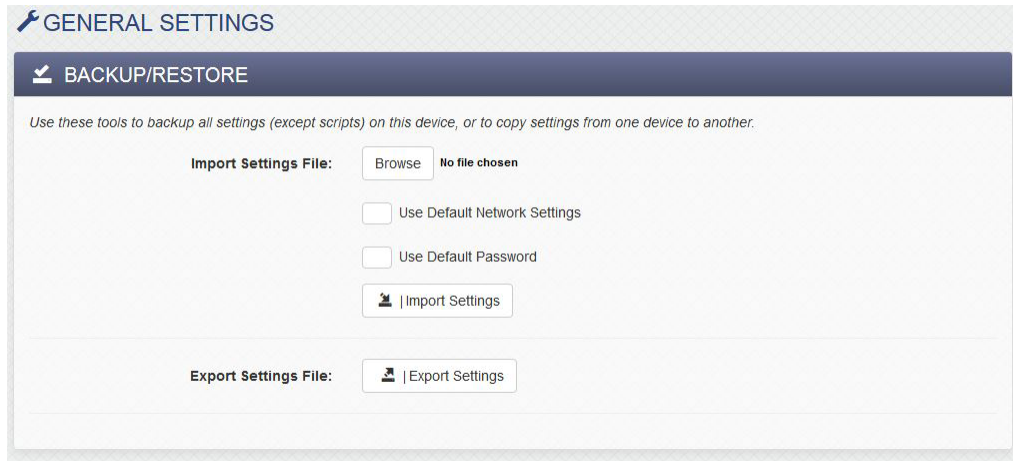
This is the date and time that daylight savings will start. Note that on this date, between the hours of 12:00 AM – 5:00 AM, the current time will be shifted forward by one hour (i.e. the time will jump from 12:02 AM [00:02] to 1:02 AM [01:02]). By default this is set to the 2nd Sunday in March which is the date used in the United States.

Daylight Savings End Date/Time

This is the date and time that daylight savings will end. On this date, between the hours of 12:00 AM – 5:00 AM, the current time will be shifted backward by one hour (i.e. time will jump from 12:02 AM [00:02] to 11:02 PM [23:02] the day before). By default this is set to the 1st Sunday in November which is the date used in the U.S.

3.2.7   General Settings Tab > BACKUP/RESTORE

Use these tools to backup the settings (excluding scripts and SSL Certificates) on this device, or to copy settings from one device to another. This is useful for "cloning" or copying devices, or to maintain a backup copy of the settings.

Import Settings File

To import settings from an external file, first click the Browse button and select the desired *settings.txt* file on your computer. If *Use Default Network Settings* is checked, any network settings in the file are ignored and the network settings are forced to the default state. If *Use Default Password* is checked, the Administrator password in the file is ignored and the Administrator password is forced to the default state. Click the *Import Settings* button to import (load) the settings from the selected file.

Export Settings File

Click the *Export Settings* button to export all of the current settings to a "settings.txt" file. If desired, the file can be opened and examined with a text editor. Note that BASIC scripts and SSL Certificates are not included in this settings file and should be saved separately.

3.3 Remote Devices / Devices Tab

The Remote Devices tab is used for all WebRelay-10 Plus™ modules.

These modules can monitor and control the I/O located on up to 32 other ControlByWeb devices (including expansion modules). I/O can be directly added to the X-400 via the expansion bus, and thirdparty modbus sensors can be added to the X-404. This feature can be used to implement a distributed control system.

With distributed control, intelligent control devices, such as the module are placed in close proximity to the sensors, relays, motors and valves that are to be monitored and controlled. The control devices are connected together into a control system with Ethernet communication. This means no central control cabinet is needed with long wire runs to the sensors, relays, motors and valves in your facility.

Figure 1: The screenshot below shows a sample device list from on an X-404 module (note that the button labeled "Add Modbus Sensor" is only available on the X-404 and X-404C modules).

With Ethernet communication, the control system can accommodate applications with long distances between the devices being monitored and controlled. Each device on the bus implements a portion of the control application.

To access I/O on remote devices or sensors, first add the device or to this device list, then add and configure their I/O under the *I/O Setup* tab. Remote devices, expansions modules, and third-party modbus sensors appear in the Device List table on the Devices tab and can be created, edited or deleted as needed.

You can add a devices to the module in two ways:

1. **Automatically:** Click *Find Devices* – This will look for expansion modules, and other X-400 Series devices (X-410, X-418, etc.) on the network and allow you to automatically add them to the module. (On the X-404, generic modbus sensors will not be detected and must be added manually.)

Model #	Serial #	Address	Add/Edit
X-15s	000000000064	N/A	Add +
X-21s	000000000A8B	N/A	Add +
X-15s	000000000026	N/A	Add +
X-11s	000000000007	N/A	Edit
X-16s	000000000521	N/A	Add +
X-15s	000000000012	N/A	Add +
X-15s	000000000013	N/A	Add +
X-15s	000000000024	N/A	Add +
X-11s	00000000002E	N/A	Add +
X-15s	000000000025	N/A	Add +
X-410	00:0C:C8:03:E1:6E	192.168.1.90	Add +
X-410	00:0C:C8:03:C2:72	192.168.1.100	Add +
XW-110	00:0C:C8:03:9F:5D	192.168.5.193	Add +

2. **Manually:** Click *Add Remote Device*. Depending on the *Model* selected in the pull-down list ▼, the available settings will change to accommodate the capabilities of the remote device. For example, only certain devices can support encrypted communications:

3.3.1 Adding Ethernet and Expansion Modules

The following settings will appear when adding expansion modules (on the X-400) and remote devices.

Device Name
This text field allows setting a descriptive device name that will be referenced throughout the setup pages. Up to 24 characters may be entered in this field. The default text is "Device 1". Set the name to a descriptive value such as "Warehouse Lights".

Model
Click the ▼ symbol and choose from the list of supported remote devices. When changing the device type of a previously configured device, only I/O compatible device types will be enabled.

Serial Number
The serial number of the remote device is entered here. The serial number is only required for X-400 series devices and expansion modules. The default value is 000CC8000000

The following settings will only appear for remote devices (non-expansion modules).

IP Address/Hostname
The IP address or hostname of the remote device. Up to 61 characters may be entered in this field.

Port
The TCP port number of the remote device. This must match the port (HTTP port) set in the remote device. The valid range is 1 to 65535. The default port number is 80. If *TLS PSK Encryption* is selected (see below), the default port number is 443.

Security Configuration
Select either *Unencrypted* or *TLS PSK Encryption*. When both devices are on the same local network unencrypted communication is usually acceptable. When devices communicate across the Internet, encrypted communication is recommended. Performance with encrypted communication will be slightly slower.

Device Control Password
This setting is shown if the security configuration is set to *Unencrypted*. Enter the password required to access the I/O on the remote device. This password will be the same as the 'User' password on the remote device.

Device PSK
This setting is shown if the security configuration is set to *TLS PSK Encryption*. This is the pre-shared key required to access the I/O on the remote device using encryption. The pre-shared key can be found on the Passwords tab on the remote device.

Device Admin Password
Enter the admin password for the remote device. With this password, the module will automatically add itself to the remote device's Device List. This setting is only shown if the remote device supports this feature. This feature will only work if the remote device is connected to the Ethernet and both the Serial Number and IP address are set (above).

Remote Device Monitor and Control:

Note that this section only applies to remote devices and does not apply to expansion modules connected to the expansion bus on the X-400. Communication with newer remote ControlByWeb devices can use both event-driven and polled communication models. Older ControlByWeb devices only used polling to monitor the inputs of remote devices. This works as long as a small delay is acceptable and direct communication to the remote device is available. (No routers with port forwarding disabled, etc.) Control of relays could be instantaneous as long as direct communication with the remote device is available.

The WebRelay-10 Plus™ contains a mechanism which allows event-driven communications in both directions. Rather than relying on polling, I/O changes are "pushed" to other devices. This has the advantage of avoiding the latencies of the polling interval, and since the information is pushed, communications can occur through a router or firewall without the need to configure port forwarding. The mechanism is enabled by selecting *Instant Send* or *Instant Receive* communications settings. These options will only appear when communicating with devices that support this.

   Standard
   If this check-box is enabled, the remote device is controlled and monitored by sending relay commands when needed and periodically polling its status. Direct communication with the remote device is required. If outside the current network, port forwarding must be configured.

   Poll Interval
   This option appears when "Standard" is checked. It specifies how often the local device reads the state of the remote device.

   Instant Receive
   If this check-box is enabled, the remote device can instantly send specific I/O changes to the local (this) device. The remote device will push I/O state to the local device periodically and/or when triggers are set up on the remote device to do this. This check box basically means that this device will accept the instant send messages from the remote device.

   Instant Send
   If this check-box is enabled, the local (this) device instantly pushes specific I/O changes to the remote device. The remote device can use these state changes to instantly change relays or perform other logic functions. In addition to enabling this option, triggers should be set on this local device for each I/O change that should cause the I/O state to be sent. Triggers are set up by an creating tasks and setting actions under those tasks to send the I/O state. (see Control/Logic Tab > TASK/FUNCTIONS). Note that when I/O status is sent, it sends the status of ALL I/O, not just the trigger. Also note that in addition to sending the I/O status when trigger events occur, the I/O status will be sent periodically at the specified Push Interval below.

   Push Interval

   This option appears when "Instant Send" is selected. It defines the time period that local data is pushed to the remote device.

3.4 I/O Setup Tab

Each WebRelay-10 Plus™ module supports a total of up to 64 Input/Output (I/O) resources. These can be a mix of local I/O, registers, timers, and Vin. These can also include I/O found on remote devices (relays, analog inputs, digital inputs, etc.). Fixed resources, such as the device's local I/O automatically appear under the I/O Setup tab.

When another ControlByWeb device is added to the module (local module), the user has the option to add I/O (relays, digital inputs, analog inputs, etc.) from that device to the local module. The I/O that was selected when adding devices will automatically appear in the appropriate section under the I/O Setup tab. I/O from remote devices or sensors can be added or deleted at any time.
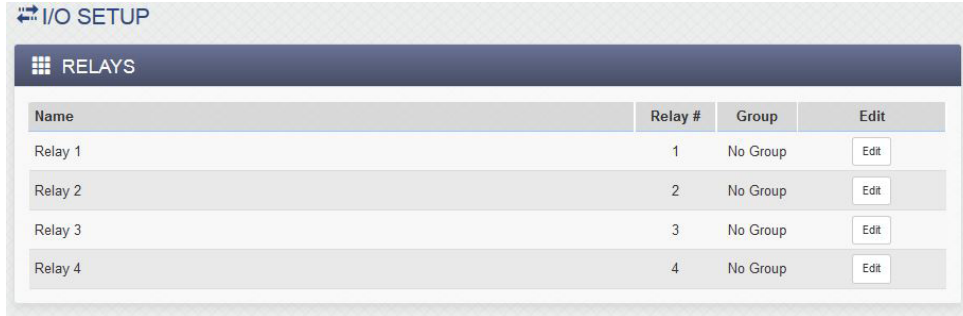
Once the I/O from other devices is added, it can be used just like local I/O that is built into the local module. For example, it can be used in control logic using the task builder, logged, or added to the Monitor and Control page. For more information about setup options for remote I/O types (relays, analog inputs, digital inputs, etc.), please see that device's users manual.

If an I/O resource is deleted, any dependent task in the Task Builder, or Control Page widget which references that particular I/O is also automatically deleted.

If logging is enabled for the I/O resource which is being deleted, the I/O resource is automatically removed from the logging list and the log file is reset (a warning is shown that the log file is about to be deleted.) The log file is deleted since the log file format is dependent on that I/O being logged. If a remote device is deleted, any dependent I/O resources, Tasks, Control Page widgets, and logs are also automatically deleted.
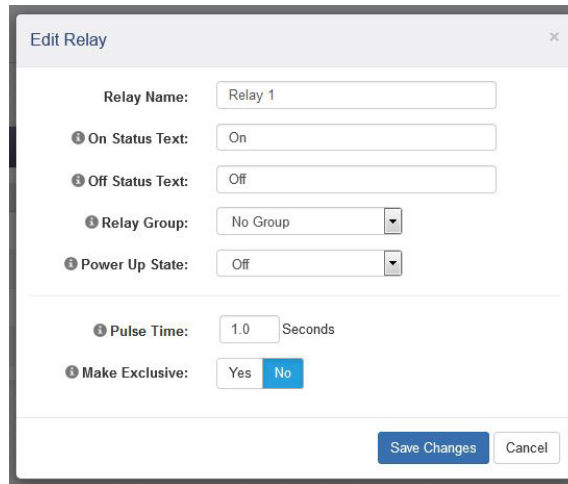
3.4.1 I/O Setup Tab > RELAYS

This page allows configuration of all relays whether on an expansion or remote device.

Click the respective *Edit* button to configure each relay. Expansion module relays have some settings that remote relays don't have and vice versa.

Relay Options

Relay Name
This text field describes the function of the selected relay. The label text appears to the left of the corresponding relay status on the *Control Page* and in the email message when email alerts are enabled. When later defining the control logic, logging and other settings, this label appears in the pull-down lists of relay resources. Up to 27 characters may be entered in this field. The default text is "Relay #". Set the name to a descriptive value such as "Warehouse Fan".

On Status Text
The text in this field specifies the text that will be displayed in the *Control Page* and in email messages when the relay is On. Up to 16 characters may be entered in this field. The default text is "On". Set the status text to a descriptive value such as "Fan On".

Off Status Text
The text in this field specifies the text that will be displayed in the *Control Page* and in email messages when the relay is Off. Up to 16 characters may be entered in this field. The default text is "Off". Set the status text to a descriptive value such as "Fan Off".

Device

For remote and expansion relays, this settings specifies the device that the relay is found on.

Device's Relay #

For remote and expansion relays, this settings specifies the relay number on the device.

Relay Group
The module can group some or all of the relays into "relay groups." Combining relays into groups provides a simple method of controlling multiple relays at once. For example, a conditional task can have an action to turn on relay group 1 that would cause all the relays assigned to relay group 1 to turn on at the same time when triggered. Each relay can still be controlled independently while assigned to a relay group. In applications where a large number of relays are used, this scheme allows one task to control all of the relays with one Action.

Local Relay Number
This number will be associated with this relay in the modulestate.xml and state.json files. If it is left unassigned, the relay will not appear in the state.xml or state.json files.

Expansion Relay Options

Power Up State:
This drop-down menu specifies the state of the relay when the module is powered up:

Off	The relay will be off upon power up. This is the default setting.
On	The relay will be on when power is applied to the module.
Last State	Upon power up, the relay will return to its last state when the module was powered off.
Scheduled State	When the device is powered up, its state will be determined by any scheduled tasks configured to change the relay state. For example, if a scheduled task turns the relay on at Noon and another scheduled task turns the relay off at 1:00 p.m everyday, then if the device loses power and comes back on between Noon and 4:00 p.m., the relay will be on. If it comes back on any other time, it will be off.
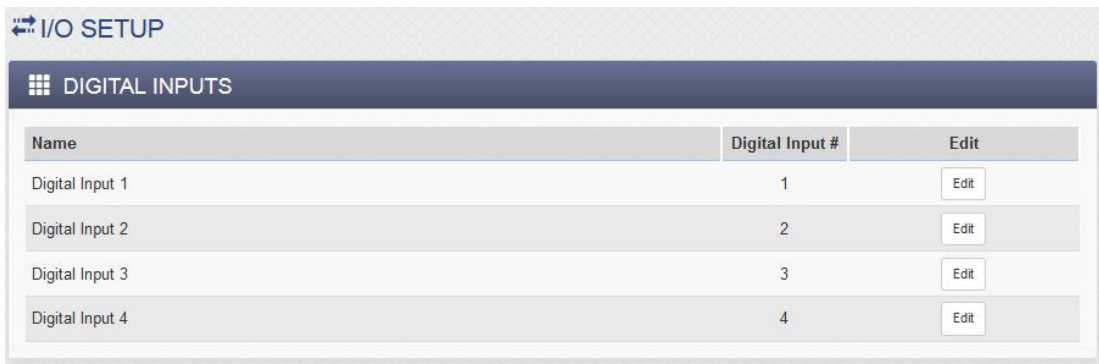
Pulse Time
When the module receives a pulse command for the relay (through the web page or through a command), the relay will pulse for the time specified in this field. Note that when a pulse command is sent through the command interface, the command can include a pulse time which will override this value. The time is specified in seconds and can range from 0.1 seconds to 86400 seconds in 0.1 second increments. The default time is 1.0 seconds.

Note that if additional pulse commands are issued to the relay before the pulse time has expired, the relay will remain "on" and the pulse timer will be set to the full pulse time once again, extending the total "on" time.

3.4.2   I/O Setup Tab > DIGITAL INPUTS

This page is used to configure expansion module digital inputs, as well as any remote digital inputs located on other ControlByWeb devices that have been added. The digital input resources can be named and have binary (true/false) values. Digital inputs appear in tables on the digital inputs tab and can be created, edited or deleted as needed.

Each of the digital inputs can be configured for use with a 24-bit digital counter which counts the number of pulses on the respective digital input. The counters can count up to 200Hz and can be used with paddle wheel flow-meters, traffic sensors and other sensors which output pulses. The counters can be named and work similar to *Registers* where the counter value is accessible to Conditional Tasks, BASIC scripts and control page widgets. The counters associated with digital inputs 1 and 2 are stored in battery-backed memory and will retain their count through a power loss; however, the counters will not continue to increment while the module power is off.
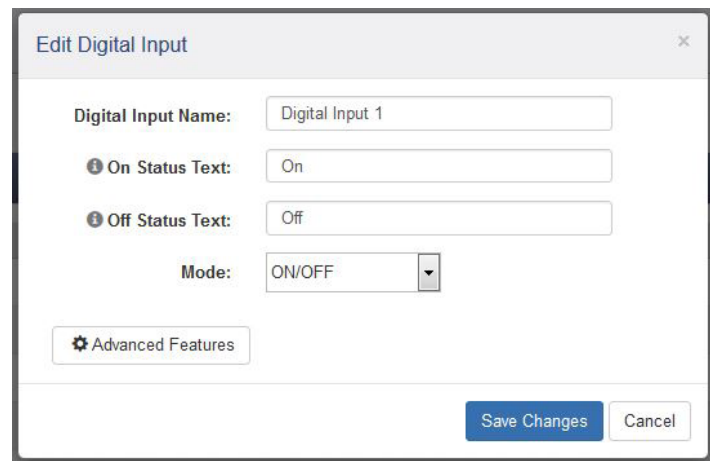
Click the respective *Edit* button to configure each digital input.

Digital Input Options

Digital Input Name
This text field describes the function of the selected digital input. The label text appears to the left of the corresponding digital input status on the Control Page and in the email message when email alerts are enabled. When later defining the control logic, logging and other settings, this label appears in the digital input pull-down list. Up to 27 characters may be entered in this field. The default text is "Digital Input #". Set the name to a descriptive value such as "Warehouse Door".

On Status Text
The text in this field specifies the text that will be displayed in the Control Page and in email messages when the digital input is On. The digital input is considered "on" when sufficient voltage is applied to the input. Up to 16 characters may be entered in this field. The default text is "On". Set the status text to a descriptive value such as "Door Open".

Off Status Text
The text in this field specifies the text that will be displayed in the Control Page and in email messages when the digital input is Off. Up to 16 characters may be entered in this field. The default text is "Off". Set the status text to a descriptive value such as "Door Closed".

Mode:

The digital input can be configured for several operating modes:

On/Off	A single I/O resource with the name "name" is created, where name is the digital input name defined above. The resource has binary (true/false) values.
Counter	The digital input is configured as a digital counter which counts the number of pulses on the respective digital input. Two I/O resources are created: name and name.counter. The counter value can be scaled into engineering units and is accessible to conditional tasks, BASIC scripts, Modbus and control page widgets. The counters associated with digital inputs 1 and 2 are stored in battery-backed memory and will retain their count through a power loss; however, the counters will not continue to increment while the module's power is off.
Frequency	The frequency of the respective digital input is measured by counting the number of pulses that occur during a period of one second. The value is updated once each second. A single I/O resource is created in the form name.frequency. The frequency value can be scaled into engineering units and is accessible to Conditional Tasks, BASIC scripts, Modbus and Control Page widgets.

Units

Text entered here will be displayed on the right of the value when shown on the Control Page. This text does not affect the underlying value. Set the units to GPM, ea., mph, etc.

Decimal Places

The number of digits displayed to the right of the decimal point. This does not affect the accuracy of the underlying value.

Slope (Multiplier)

The counter and frequency values can be scaled into engineering units. The module reads the raw value from the counter and calculates a new value in engineering units. This scaled value is calculated using the following y=mX+b formula..

Scaled Value = Slope * Count + Offset

The calculated scaled value is used for logs, email messages, the Control Page, XML pages, and is returned when Modbus values are read.

The "slope" in the formula above is provided by the user and is entered in this field. When both the slope and offset are set to their default values (m=1 and y=0) the scaled value equals the actual count.

Offset

The offset in the formula above is provided by the user and is entered in this field. The default value for this field is 0.

Device
For digital inputs, this settings specifies the device that the digital input is found on.

Device's Digital Input #
For digital inputs, this settings specifies the digital input number on the remote or expansion device.
Note: For remote counters, frequency, etc., the values are read from the remote device and not calculated locally.

Local Digital Input Number
This number will be associated with this digital input in the module state.xml and state.json files. If it is left unassigned, the digital input will not appear in the state.xml or state.json files.

Counter Options

Digital Input Count Mode

Increment when ON	Count on the rising edge of the pulse
Increment when OFF	Count on the falling edge of the pulse
Increment when input changes	Count on both rising and falling edges of the pulse

Reset Count At

The counters automatically "roll over" to zero when the count reaches the value of this setting. This setting is an integer for the raw count, before any slope multiplier or offset is applied. When *name.counter.* is accessed by Conditional Tasks, BASIC scripts and Control Page widgets, the slope multiplier and offset are applied, and the value is a single precision, floating-point value.

Using an example to clarify, if Reset Count = 5 the counter will count as follows...

0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0

Reset Counter
This button manually resets the counter to zero

Advanced Features
The digital inputs can be configured for several additional operating modes:

>   *Hold Time (de-bounce):* The input must remain *On* for this amount of time before the logic considers this input to be *On*. For switches and other noisy sensors, this eliminates problems with falsely detecting multiple *On* states before the input becomes stable. The default setting is 20ms and can be set from 0-250ms. The digital input is processed by this setting before it is applied to the on/off, counter, and frequency logic. This setting directly limits the maximum input frequency that can be measured.

>   *Measure On-Time:* When this feature is enabled, the time that the input is "on" is measured. This value will be cleared when the input changes from "off" to "on". When the input changes from "on" to "off" the value will be frozen and can be accessed as long as the input remains "off". The *On-Time* only starts counting once the debounce time has expired and continues after the input goes off until the debounce timer expires again. An I/O resource is created in the form: *name.onTime*

>   *Measure Total On-Time:* When this feature is selected, the accumulated time that the input is on is measured. When the input is "off," the timer stops. When the input goes "on" again, the timer resumes. This feature is useful for example, to measure the total run time of a machine to determine when maintenance is needed. The Total On-Time only starts counting once the debounce time has expired and continues after the input goes "off" until the debounce timer expires again. An I/O resource is created in the form: *name.totalOnTime*.

3.4.3 I/O Setup Tab > 1-Wire SENSORS

This tab is used to configure local, expansion, and remote 1-Wire sensors (temperature, humidity, and/or other 1-Wire sensors). Up to 16 1-Wire sensors can be connected to the module. The module will automatically detect the type of sensor connected. Once added to the module, 1-Wire sensors appear in the table on the 1-Wire Sensors tab and can be created, edited or deleted as needed.

Every 1-Wire sensor comes from the factory with a unique, non-changeable address. Once the sensor is connected to the module, the module must associate a sensor name with the sensor address. You must define a sensor name and associate an address for each 1-Wire sensor. For remote 1-Wire sensors, only the device and sensor number (not address) are needed (the 1-Wire address should be configured locally on the remote device).

For example, a list of named, associated 1-Wire sensors is shown:

To add a new local 1-Wire sensor, click *Add 1-Wire Sensor* in the local 1-Wire Sensors table.

Enter a descriptive name for the sensor in the *Sensor Name* box such as "Outdoor Temperature". The text in this field appears to the left of the corresponding temperature/humidity reading on the Control Page. This text also appears in the email status message when email notifications are enabled. When later defining the control logic, logging and other settings, this label appears in the pull-down lists of sensor resources. This field can be up to 27 characters in length.

Click ▼ to see a list of the connected 1-Wire sensors. The addresses of the non-associated sensors on the bus are shown in the drop-down list together with the sensor type of each sensor. Click the Refresh button to make a fresh scan of the bus for sensors. For each sensor being added, select the appropriate address from the sensor address drop-down list. As 1-Wire sensors are added to the module, the addresses are removed from the pull-down ▼list of available addresses. The X-DTHS-P temperature/humidity probe is a composite device and has a single 1-Wire address. The drop-down list will show both a temperature and a humidity ID with the same sensor address. Select the appropriate ID (temperature or humidity) and the sensor address that matches the sensor address printed on the housing of the X-DTHS-P.

Alternatively, connect the sensors to the bus one at a time. The procedure is to start with one sensor and add it with the appropriate sensor name by selecting the sensor address within the drop-down list. Press *Submit*, connect a second sensor, and press the Refresh List button. Associate the second sensor to the appropriate sensor name. Continue this procedure until all sensors are set up.

The *General Settings* page has a setting to select the display for *Fahrenheit*, *Centigrade* or *Kelvin* units. 1-Wire sensor addresses that have already been associated will not show in the list of available 1-Wire sensors.

1-Wire Options

Decimal Places
The number of digits to the right of the decimal point. This setting does not affect the accuracy of the underlying value. It is applied whenever the 1-Wire value is displayed in the Log files, Control Pages, Emails, Etc.

Local 1-Wire Number
This number will be associated with this 1-Wire sensor in the module state.xml and state.json files. If it is left unassigned, the 1-Wire sensor will not appear in the state.xml or state.json files.

Local & Expansion 1-Wire Options

Offset
If desired, an offset can be added to the 1-Wire sensor's measurement. By default, the offset is 0.0

FinalValue = Sensor's Measured Value + Offset

Remote 1-Wire Options

Device
For remote 1-Wire sensors, this settings specifies the device to which the remote 1-Wire sensor is connected.

Device's Sensor
For remote 1-Wire Sensors, this settings specifies the 1-Wire number on the remote device. (Not the address of the 1-Wire sensor.)

Sensor Type
The module can determine the type of 1-Wire sensor if it is connected directly to the device. For remote 1-Wire sensors the sensor type must be configured for the module to know which type of sensor it is.

3.4.4 I/O Setup Tab > REGISTERS

Registers are working variables that can be changed externally through the Control Page, XML requests, or the BASIC script. Registers can allow a BASIC script to react to user input. These variables are considered to be floating point numbers, just like other variables in the BASIC script. Registers can also hold boolean information where "1"=true and "0"=false. This tab has settings for each of the configured Registers. By default, one register (register1) is pre-defined and is battery-backed such that its value is retained across power loss. Registers appear in tables on the Registers tab and can be created, edited, or deleted as needed. Up to 16 Local Registers can be created. The module can also access remote registers found on other devices.

I/O SETUP

REGISTERS

Name	Register #	Units	Edit
Register 1	1		Edit
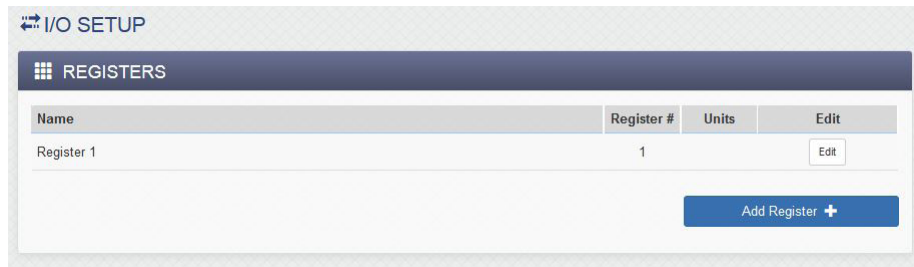
Add Register ✚

Register Options

Register Name
Text entered here will be displayed in the left column of the control page. Up to 27 characters may be entered here. The default text is "Register 1". This is also the resource name used in a BASIC script. When later defining the control logic, logging and other settings, this label appears in the pull-down lists of Register resources.

Decimal Places
The number of digits displayed to the right of the decimal point. This does not affect the accuracy of the underlying value.

Add Register ✕

Register Name:	Register 2
ⓘ Power Up State:	Scheduled State ▾
ⓘ Decimal Places:	2
Units:	
Minimum Value:	0.00
Maximum Value:	100.00

Add Register   Cancel

Units
Text entered here will be displayed on the right of the value when shown on the control page. This text does not affect the underlying value. Set the units to F, ft, mph etc.

Minimum Value
If this register can be changed externally through the *Control Page, XML* requests, or the BASIC script it is may be useful to restrict the range of the register value. For example, a thermostat set-point register can be restricted to a minimum value.

Maximum Value
If this register can be changed externally through the *Control Page, XML* requests, or the BASIC script it is may be useful to restrict the range of the register value. For example, a thermostat set-point register can be restricted to a maximum value.

Local Register Number
This number will be associated with this register in the module state.xml and state.json files. If it is left unassigned, the register will not appear in the state.xml or state.json files.

Local Register Options

Power Up State
At power-up the register is set to either an *Initial Value* (set below) or the *Scheduled State*. The Scheduled State will cause the Register value to be that defined by any scheduled tasks as if the power was always on.

Remote Register Options

Device
The device where the remote register is found.

Device's Sensor #
The register number on the remote device.

3.4.5   I/O Setup Tab > VIN

The power supply voltage (Vin) to the module is internally measured and can be displayed, logged, or used in logic. This page configures the name for the Vin voltage.

Note: The module can also monitor the power supply voltage of remote devices. For POE devices, the power is provided over the Ethernet cable and the Vin+ monitor will show 0-volts.
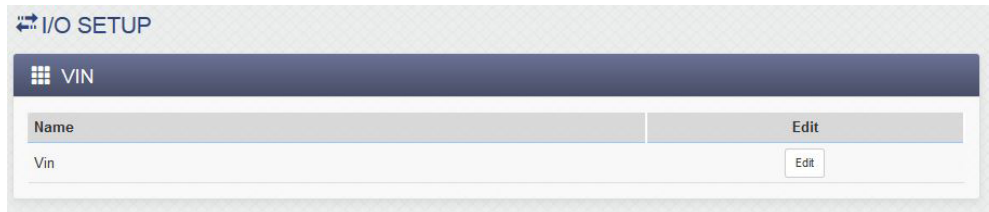
To edit the Vin name, click *Edit.*
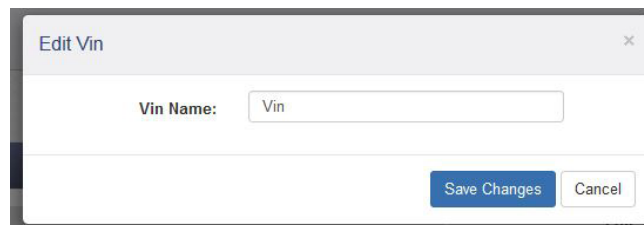
Vin Options

Vin Name

This text field is used to describe the power supply voltage value. By default it is set to "Vin." The text appears to the left of the Vin status on the *Control Page*. This text will also appear in email messages when email alerts are enabled. This field may be up to 27 characters long. Set the name to a descriptive value such as "Power Supply" or "Battery Voltage".

Remote Vin Options

Device

The device where the remote Vin is found.

3.4.6   I/O Setup Tab > TIMERS

Timers are I/O resources for measuring the time between events/tasks or generating controlled delays.

The timers are essentially 24-bit down counters with a one-second resolution. Timers appear in a table on the Timers tab and can be created, edited, or deleted as needed. Up to 16 Timers can be created. The timer's value is accessible to Scheduled and Conditional Tasks, BASIC scripts, Modbus and Control Page widgets. There are no remote timers.

A timer is started via an *Action* by setting it to a non-zero value. The timer counts downwards and stops when it reaches 0. When the timer reaches zero it generates a "timer expires" event. The "*timer expires*" events appear in the drop-down list ▼ of available *Triggers* for Scheduled and Conditional Tasks.
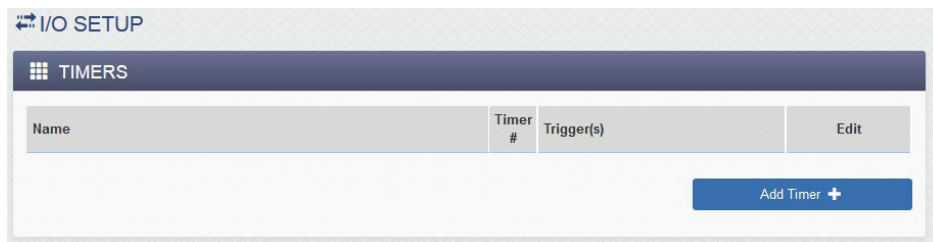
To add a timer, click *Add Timer.*

Timer Name
This text field is used to describe the timer. By default it is set to "Timer #." This field may be up to 27 characters long. Set the name to a descriptive value such as "Warmup Delay".

(Note: This text appears to the left of the timer status field on the Control Page. It also appears in the pull-down lists under "Timer" resources when configuring tasks, logging, and other settings.)

Power Up Value
At power-up, the timer is set to this value. If the power-up value is something other than 0, the timer will begin to count down once the device is powered.

Trigger(s)
Scheduled and Conditional Tasks can start and stop a Timer. Once a timer is added and named in this tab, it will appear in the pull-down list ▼ of available *Actions* for Scheduled and Conditional Tasks. The Action can be specified to either reset the counter (stop) or to set it to a fixed value (start). Actions that control the timer are shown in the Trigger(s) field. *(See example to the right for reference.)*

3.5 Control/Logic Tab

The sub-menus under the Control Logic tab define the logic of the module. Conditional and Scheduled tasks as well as BASIC scripts are defined under this tab group.

3.5.1 Control/Logic Tab > TASK/FUNCTIONS

"Tasks" are control logic functions that perform specific things, such as turn a remote relay on/off under certain conditions or at certain times. Tasks are assigned a name, configured for a desired function, and saved. Once saved, they appear in tables on the Tasks/Functions tab and can be edited or deleted as needed.

Tasks are configured using settings and pull-down menus to define the logic function without the need to learn boolean or other programming syntax. For more complex logic functions or logic functions not supported by this tab, see *Section* Control Logic Tab > BASIC SCRIPT.

Up to 50 Scheduled Tasks and 50 Conditional Tasks can be created. Tasks can run once, periodically, or continually depending on its settings. Under the Control/Logic tab, you can observe all of the tasks, their start times, run modes, triggers and actions.

Scheduled tasks run at specific times. They can be configured to happen only one time, or on a repeated schedule.

Normal/Override
The Tasks/Functions tab has a status display which shows the current time of day and whether the *Normal* or *Override* schedule is running.

An **Override Schedule** can temporarily cause certain Scheduled tasks to stop operating (i.e., holidays), and resume normal schedules after the Override Schedule expires.

Up to 20 Override Schedules can be added.

For debug and testing, open the *Control* Page in another browser window (or another browser) and together with this page you can monitor which tasks are running and what they are doing in response to user input.

See Control Logic Examples Using Tasks/Functions for examples of using Tasks for control logic.

Conditional (boolean) tasks run only if specified conditions are met (e.g., "If an input is ON then turn a relay ON." -or- "If a temperature sensor is above 32 degrees, send an email alert and turn a relay ON.").

3.5.1.1 Scheduled Tasks

Scheduled tasks run at a specific time and on specific days of the week (e.g. turn a light on at 08:00 on Monday through Friday – or – start/clear logging at 04:00 on the first day of the month).

Scheduled tasks can also be configured to be conditional based on specified resources.

Each *Scheduled task* can initiate up to three *Actions*. Actions are control outputs that "do" something such as: turn a remote relay on or off, send an email, send a SNMP trap, or make a data log.

Scheduled Tasks appear in a table on the Control/Logic Setup tab and can be created, edited, or deleted as needed. Up to 50 Scheduled Tasks can be created.

Name	Start Date/Time	Repeat	Actions	Next Occurance	Run Mode	Edit

📅 SCHEDULED ⓘ   Add Scheduled Task ✚

To add a Scheduled task, click *Add Scheduled* Task.

Add Scheduled Task   ✕

Task Name: Scheduled Task 1

ⓘ Run Mode: Always

Start Date: January   14 , 2010

Start Time: Set

08 : 00 : 00

Condition: None (Optional)

ⓘ Set Action 1:   None (Optional)

ⓘ Set Action 2:   None (Optional)

ⓘ Set Action 3:   None (Optional)

Set Repeat: No Repeat

Add   Cancel

Task Name:
This text field is used to describe the task. By default it is set to "Scheduled Task 1". This field may be up to 24 characters long. Set the name to a descriptive value such as "Unlock Front Door".

Run Mode:
The Run Mode defines when the scheduled task is active. It has the following options:
- Always   Always active.
- Normal Schedule   Active, unless overridden by an Override schedule.
- Override Schedule Active only during an Override schedule.
- Off   Never active (disabled).

Start Date:

Select the month, day, and year which the scheduled task is to occur for the first time using the dropdown boxes.

Start Time:

▪ ***Set:*** Select the time of day which the scheduled task is to occur for the first time.
▪ ***Sunrise:*** Enter the offset (HH:MM) from sunrise. Choose *Before* or *After* sunrise.
▪ ***Sunset:*** Enter the offset (HH:MM) from sunset. Choose *Before* or *After* sunset.

Note: *The sunrise and sunset times are automatically calculated based on the latitude and longitude settings made in the General Settings Tab > GENERAL INFORMATION tab.*

Condition – Scheduled Task (Optional):

Scheduled tasks can optionally be controlled (enabled) based on an I/O's state (local and remote). In order for I/O to appear in the condition pull-down menu ▼, the I/O must be first added and named in the I/O Setup tab. The scheduled task only runs if the conditional I/O state evaluates as "true" at the scheduled time.

If ***None*** is selected, no conditional logic will be evaluated when performing the task.

If a ***Relay*** resource is selected, the following conditions are available:

ON	The scheduled task runs if the relay is On
OFF	The scheduled task runs if the relay is Off

If a ***Digital Input*** resource is selected, the following conditions are available:

ON	The scheduled task runs if the input is On
OFF	The scheduled task runs if the input is Off

If a ***Digital Input Counter*** resource is selected, the following conditions are available:

Value = X.X	If the Counter value is equal to the setting, the compare is true.
Value >X.X	If the Counter value is greater than the setting, the compare is true.
Value < X.X	If the Counter value is less than the setting, the compare is true.

If a ***1-Wire Sensor*** resource is selected, the following conditions are available:

Value ≥ X.X	If the sensor value is greater than or equal to the setting, the compare is true.
Value <X.X	If the sensor value is less than the setting, the compare is true.
Deadband	Hysteresis for the compare

If an ***Analog Input*** resource is selected, the following conditions are available:

Value ≥ X.X	If the analog input value is greater than or equal to the setting, the compare is true.
Value <X.X	If the analog input value is less than the setting, the compare is true.
Deadband	Hysteresis for the compare (*see more details below*)

If a pseudo Digital Input resource is selected, the following conditions are available:

ON	The scheduled task runs if the input is On
OFF	The scheduled task runs if the input is Off

If a Register resource is selected, the following conditions are available:

Value = X.X	If the Register value is equal to the setting, the compare is true.
Value >X.X	If the Register value is greater than the setting, the compare is true.
Value < X.X	If the Register value is less than the setting, the compare is true.
Deadband	Hysteresis for the compare (see more details below)

If a Vin resource is selected, the following conditions are available:

Value ≥ X.X	If the Vin value is greater than or equal to the setting, the compare is true.
Value <X.X	If the Vin value is less than the setting, the compare is true.
Deadband	Hysteresis for the compare (see more details below)

If a Timer resource is selected, the trigger is true when the timer expires (reaches 0).

Deadband

The *Deadband* settings prevent alarms from triggering excessively when an analog value vacillates around the trigger point.

With high alarms, the value must fall below the high alarm point minus the *Deadband* and then back above the high alarm point again, before the high alarm will be triggered again.

Likewise the *Deadband* on the low alarm requires the value to rise above the low alarm point plus the Deadband and then back below the low-alarm point again, before the low alarm will be triggered again.

For example, if the *Deadband* is set to 1 degree, and a high alarm is occurs at 95 degrees, the *Deadband* ensures that once the high alarm is triggered, it won't trigger again until the temperature first drops below 94 degrees (95 – 1).

Actions - Scheduled Task

Up to three actions can be specified for each Scheduled Task. Actions can change the state of local and remote I/O, counters, registers, and other resources. To be available in the task's action pull-down menus ▼, the resource must be first added and named in the I/O Setup tab.

If *None* is selected, nothing is done when the task runs.
If *Relay* is selected, the following actions are available:

On	The relay is turned on
Off	The relay is turned off
Pulse On	The relay is turned on for the amount of time specified as the pulse time for that relay and then turned off. The pulse time is set for each individual relay in the relay settings under the I/O tab on the device where the relay is physically located. Note: If a new pulse command is set to the relay before the pulse time expires, the pulse timer will be reset to its full value and the relay will remain on until the pulse timer expires.
Toggle On/Off	The relay changes state to the opposite of its current state.
Condition-1 I/O Value	The relay is set to the state of the I/O in Condition-1 of the trigger.
Opposite of Condition-1 I/O Value	The relay is set to the opposite state of the I/O in Condition-1 of the trigger.
Pulse On (heartbeat mode)	This option is used only for remote relays. While the trigger is true, a continual stream of pulse commands will be sent to the relay. The pulse commands will be sent out at the same rate as the "Poll Interval" which is set for the remote device (under the Remote Devices tab). If the Poll Interval time is less than the pulse time for the relay, the relay will receive pulse commands faster than the pulse timer is allowed to expire, so the relay will remain on until the trigger becomes false or the relay doesn't receive the command due to network failure. This option is useful for "fail safe" applications where it is important to turn the relay off in case of failure.

If *Digital Input Counter* is selected, the following actions are available:

Log	The counter value is logged
Reset	The counter is reset to 0
Log & Reset	The counter value is logged, then reset to 0

If *Register* is selected, the following actions are available:

Set-To	Set the value of the register to a specific value
Increment By	Increment the register by the specified value
Decrement By	Decrement the register by the specified value

If *Timer* is selected, the following actions are available:

Start Timer	Set the timer to the specified value
Clear (stop) Timer	Reset the timer to 0, and any actions that depend upon the timer will be cancelled.

If *Log* is selected, the following actions are available:

Log	A data log entry is made of the resources enabled in Logging Tab > LOG I/O
Reset Log	Clear (Erase) the log file
Pause Logging	Temporarily pause all logging to the log file.
Resume Logging	Resume logging after previously pausing it.

If *Email* is selected, an email is sent when the task runs.

Send Email To	With the drop down box, select the Email address to use. The address list is configured in the General Settings tab > Email settings tab.

If **SNMP TRAP** is selected, the following actions are available:

Send SNMP Trap for Cond 1 I/O value	Send SNMP Trap to Configured SNMP Managers for I/O in Condition 1
Send SNMP Trap for Cond 2 I/O value	Send SNMP Trap to Configured SNMP Managers for I/O in Condition 2
Send SNMP Trap for Cond 1 and 2 I/O value	Send SNMP Trap to Configured SNMP Managers for I/O in Condition 1 and 2.

If **SNMP NOTIFICATION** is selected, the following actions are available:

Send SNMP Notification for Cond 1 I/O value	Send SNMP Notification to Configured SNMP Managers for I/O in Condition 1
Send SNMP Notification for Cond 2 I/O value	Send SNMP Notification to Configured SNMP Managers for I/O in Condition 2
Send SNMP Notification for Cond 1 and 2 I/O value	Send SNMP Notification to Configured SNMP

If **Remote Services Notification** is selected:

Send Device State to Remote Service	The local device state is sent to the remote services server if configured.

If **Push I/O State to Remote Receiver Devices** is selected:

Push I/O State to Remote Receiver Devices Action	The state of the module is sent to all remote devices configured to receive such messages.

If **Set I/O Color** is selected, an I/O's status color can be changed on the Control Page.

Repeat: The *Scheduled Task* is repeated:

No Repeat	Runs only once
Seconds	Repeats once every X seconds
Minutes	Repeats once every X minutes
Hourly	Repeats once every X hours
Daily	Repeats once every X days
Weekly	Repeats weekly on the selected days of the week
Monthly	Repeats monthly either on selected days of the month, or the 1st Sunday, etc.
Yearly	Repeats yearly on the 1st, 2nd, 4th Day of the Start Date Month, or the 1st, 2nd, 4th day of the Start Date every month that year

3.5.1.2 Conditional Tasks

Conditional tasks occur (run) if certain conditions are met. Conditional tasks are made by comparing specified resources using simple and/or (boolean) logic. For example, if a digital input goes on, turn a relay on.

Conditional tasks can be configured to be active continually or only during specific time periods each day/week.

Each *Conditional Task* can initiate up to three *Actions*. Actions are control outputs that "do" something such as turn a relay on or off, send an email, send a SNMP trap or make a data log.

Conditional Tasks are event driven and do not evaluate asynchronously. Their triggers are only evaluated when the conditions of the trigger change state. Conditional Tasks appear in a table on the Control/Logic Setup tab and can be created, edited or deleted as needed. Up to 50 *Conditional Tasks* can be created.

To add a Conditional Task, click the "*Add Conditional Task +*" button, and a popup window appears.

Task Name

This text field is used to describe the task. By default it is set to "Conditional Task 1". This field may be up to 24 characters long. Set the name to a descriptive value such as "High Temp Shutdown".

Triggers - Conditional Tasks

The "Trigger" section is a boolean logic comparison of two I/O. Triggers ONLY occur when the conditions change to true. The logic operations are event driven, and are not combinational. Conditions must change to false and back to true in order to re-trigger.

Note: To appear as a condition in the task drop-down menus ▼ , the I/O must be first added and named in the I/O Setup tab.

The two conditions (arguments) can be evaluated with "AND" or "OR" boolean logic. If both conditions are set to *None*, the Conditional Task is never evaluated. If one trigger is set to *None*, only one condition is evaluated for a True or False state.

If a **Relay** is selected the following logic states are available:

On	The condition is true if the relay is on
Off	The condition is true if the relay is off
Changes	The condition is true if the relay changes state

If a *Digital Input* is selected the following logic states are available:

On	The condition is true if the digital input is on
Off	The condition is true if the digital input is off
Changes	The condition is true if the digital input changes state

If a **Digital Input Counter** is selected, the following conditions are available:

Value =X.X	If the Counter value is equal to the setting, the condition is true.
Value >X.X	If the Counter value is greater than the setting, the condition is true.
Value < X.X	If the Counter value is less than the setting, the condition is true.

If a **1-Wire Sensor** selected, the following conditions are available:

Value ≥ X.X	If the sensor value is greater than or equal to the setting, the condition is true.
Value <X.X	If the sensor value is less than the setting, the condition is true.
Deadband	Hysteresis for the comparison

If **Analog Input** is selected, the following conditions are available:

Value ≥ X.X	If the analog input value is greater than or equal to the setting, the condition is true.
Value <X.X	If the analog input value is less than the setting, the condition is true.
Deadband	Hysteresis for the comparison *(see more details below)*.

If a **Pseudo Digital Input** resource is selected, the following conditions are available:

ON	If the pseudo digital input is on, the condition is true.
OFF	If the pseudo digital input is off, the condition is true.

If a **Register** is selected, the following conditions are available:

Value =X.X	If the Register value is equal to the setting, the condition is true.
Value >X.X	If the Register value is greater than the setting, the condition is true.
Value < X.X	If the Register value is less than the setting, the condition is true.

If a **Vin** selected, the following conditions are available:

Value ≥ X.X	If the Vin value is greater than or equal to the setting, the condition is true.
Value <X.X	If the Vin value is less than the setting, the condition is true.
Deadband	Hysteresis for the comparison (see more details below).

If a **Timer** *is* selected, the condition is true when the timer expires (reaches 0).

Deadband
The *Deadband* settings prevent alarms from triggering excessively when an analog value vacillates around the trigger point.

With high alarms, the value must fall below the high alarm point minus the *Deadband* and then back above the highalarm point again, before the high alarm will be triggered again.

Likewise the *Deadband* on the low alarm requires the value to rise above the low
alarm point plus the Deadband and then back below the low-alarm point again, before the low alarm will be triggered again.

For example, if the *Deadband* is set to 1 degree, and a high alarm is occurs at 95 degrees, the *Deadband* ensures that once the high alarm is triggered, it won't trigger again until the temperature first drops below 94 degrees (95 – 1).

During – Conditional Tasks
Conditional tasks can be configured to run continually or during specific days each week.

Always
Runs continually.

Set Time
Specify the time interval and day(s), when this task will evaluate.

Note: If the Start Time begins after the end time (i.e., Start Time: 8:00pm, End Time: 6:00am), then the time interval will start and continue through the end time of the next day.

Actions – Conditional Tasks
Up to three actions can be specified for each Conditional Task. Tasks can change the state of a remote relay, counters, registers and other remote I/O. To be available in the task's action pull-down menus ▼, the resource must be first added and named in the I/O Setup tab.

If **None** is selected, nothing is done when the task runs.
If a **Relay** Action is selected the following actions are available:

On	The relay is turned on

Off	The relay is turned off
Pulse On	The relay is pulsed on (then off)
Toggle On/Off	The relay changes state
Condition-1 I/O Value	The relay is set to the state of condition 1 I/O
Opposite of Condition-1 I/O Value	The relay is set to the opposite state of condition 1 I/O
Pulse On (heartbeat mode)	The relay is pulsed continually over and over while the trigger is true. When the trigger changes to false, the relay will go off. This is useful for remote relays and detecting network drop-outs, etc.

If a **Digital Input Counter** Action is selected, the following actions are available:

Log	The counter value is logged
Reset	The counter is reset to 0
Log & Reset	The counter value is logged, then reset to 0

If *Register* is selected, the following actions are available:

Set-To	Set the value of the register to a specific value
Increment By	Increment the register by the specified value
Decrement By	Decrement the register by the specified value

If *Timer* is selected, the following actions are available:

Start Timer	Set the timer to the specified value
Clear Timer	Resets the timer to 0

If *Log* is selected, the following actions are available:

Log	A data log is made of the resources enabled in Logging Tab > LOG I/O
Reset Log	Clear (Erase) the log file
Pause Logging	Temporarily pause all logging to the log file
Resume Logging	Resume logging after previously pausing it

If *Email* is selected, an email is sent when the task runs.

Send Email To	With the drop down box, select the Email address to use. The address list is config-ured in the General Settings Tab > EMAIL SETTINGS tab.

If *SNMP TRAP* is selected, the following actions are available:

Send SNMPTrap for Cond 1 I/O value	Send SNMP Trap to configured SNMP Managers for I/O in Condition 1
Send SNMP Trap for Cond 2 I/O value	Send SNMP Trap to configured SNMP Managers for I/O in Condition 2
Send SNMP Trap for Cond 1 and 2 I/O value	Send SNMP Trap to configured SNMP Managers for I/O in Condition 1 and 2

If *SNMP NOTIFICATION* is selected, the following actions are available:

Send SNMP Notification for Cond 1 I/O value	Send SNMP Notification to configured SNMP Managers for I/O in Condition 1
Send SNMP Notification for Cond 2 I/O value	Send SNMP Notification to configured SNMP Managers for I/O in Condition 2
Send SNMP Notification for Cond 1 and 2 I/O value	Send SNMP Notification to configured SNMP Managers for I/O in Condition 1 and 2.

If *Remote Services Notification* is selected:

Send Device State to Remote Service	The local device state is sent to the remote services server if configured.

If **Push I/O State to Remote Receiver Devices** is selected:

Push I/O State to Remote Receiver Devices	The state of the module is sent to all remote devices configured to receive such messages.

If **Set I/O Color** is selected, an I/O's status color in the Control Page can be changed.

3.5.1.3 Automatic Reboot Tasks

The module can function as a reboot controller with fully configurable setup options. It can be used to reboot servers, computers, satellite modems and other devices if they fail to respond to pings sent from the device. The device pings a device on a network, and if unresponsive, starts, stops, or reboots the device connected to a ControlByWeb device's relay. Automatic Reboot Tasks appear in a table on the Control/Logic Setup tab and can be created, edited or deleted as needed. Up to 16 Reboot Tasks can be created. The status field in each Reboot Task dynamically shows what the task is doing. For example, status messages such as: "Waiting for 1st Ping: 25 sec", or "Pinging, Failed Pings: 5, Reboot Attempts:10" will appear.

There are three ways that the ControlByWeb device can be wired to reboot a device. All three reboot methods have different requirements for wiring connections and relay timing.

Connect a ControlByWeb device's relay series with the power source of the device you would like to reboot. Simply switch the power off then on. If the AC line voltage is to be switched off, a line-voltage interposer relay may be needed depending on the ControlByWeb device's relay specifications.

If the DC power to a modem or other device is to be switched off, the DC connections can be made to one of the internal relays in the ControlByWeb device.

Some relays share a common, such as in the four relays of the X-410, so in this case the other relays cannot be connected to other loads.

If the other relays are needed, use an interposer relay to switch the DC power to the device to be rebooted.

> 1. Connect a ControlByWeb device's relay in parallel with the on/off switch of the device. This is useful for devices with momentary power buttons that will not automatically start operating when power is applied. These devices require a user to momentarily press a switch before the device will boot. An interposer relay may or may not be needed.

> 2. Connect a ControlByWeb device's relay to the reset button or circuit of a device. In this configuration, the device does not power down the device, but simply forces the device to reset.

To add an Automatic Reboot task, click *Add Auto Reboot* Task.

Task Name

This text field is used to describe the task. By default it is set to "Reboot Task 1". This field may be up to 24 characters long. Set the name to a descriptive value such as "Server Watchdog".

Ping IP Address/Name

This is the IP address or hostname that the ControlByWeb device will ping. This is typically the IP address or hostname of the device that will be re-booted upon failure (such as a server, router, computer, etc.). It could however, be the address of a device on the opposite side of a communications link. This is useful to auto reboot communications devices such as CSU/DSUs, satellite modems, routers, etc.

Ping Interval

This is the time interval (in seconds) between successful ping requests. This time can be set from 1 to 6000 seconds (100 minutes or 1.67 hours).

Unsuccessful Ping Interval

When ping requests fail (no response from device), it may be desirable to begin pinging at a shorter time interval. This allows the ControlByWeb device to determine in less time that the device is not functioning. After each unsuccessful ping, the device will wait this time interval (in seconds) before the next ping attempt. This time can be set from 1 to 6000 seconds (100 minutes or 1.67 hours).

Delay Before First Ping After Reboot

When the ControlByWeb device determines it's time to reboot a device due to loss of communication, it will wait this time interval (in seconds) before beginning to ping the device after the relays the device is connected to have power cycled the device. This time delay allows the device to boot before being pinged. This time can be set from 0 to 6000 seconds (100 minutes or 1.67 hours).

Delay Before First Ping After Reboot

The device will be rebooted after this number of consecutive ping failures. This number can be set between 1 and 255.

Max Reboot Attempts

After the device has been re-booted this number of times without any successful pings, it is assumed that there are problems with the device and the ControlByWeb device will no longer attempt to reboot the device. At this point, the device will leave the device on and continue pinging the device, but will not reboot the device anymore. Once the device responds to pings again, the device will return to pinging and rebooting the device. This number can be set between 2 and 255.

Reboot Timer (T)

Timer value used in reboot option. If the power source to the device is controlled by a relay, this is the time the power is switched off. If the relay is connected to the on/off switch, this is the time the on/off switch is momentarily activated. If the relay is connected to the reset circuit, this is the time the reset is momentarily activated. This time can be set from 1 to 6000 seconds (100 minutes or 1.67 hours).

Reboot Options:

Pulse Relay Off T seconds	Connect the NO contacts of the interposer relay to the AC power line of the device being monitored. If the device fails to respond to pings, the power is turned off, then on.
Pulse Relay On T seconds	Connect the NC contacts of an interposer relay to the AC power line of the device being monitored. If the device fails to respond to pings, the power is turned off, then on. Connect the NO contacts of the relay to the reset circuit of the device being monitored. If the device fails to respond to pings, the reset is momentarily activated.
Turn Relay OFF until successful ping resumes	If the device is unresponsive, stop the device connected to a relay
Turn Relay ON until successful ping resumes	If the device is unresponsive, start the device connected to a relay

Relay/Register

With the drop-down box, select the relay (or register) from the list that will reboot the monitored device. A Register can be used in applications where more complex logic or control is needed and a Task monitors the Register to initiate a reboot. Use a Register and Task for example, if reboots are wanted only on a schedule during certain times each day.

Email Notifications

Email notifications of a reboot sequence can be *Enabled* or *Disabled*. If enabled, an email will be sent whenever the device is rebooted. The email will actually be queued until the device is back up, in case the device is needed in order to send a successful email.

Select Email Address

With the drop-down box, select the Email address to send reboot notifications. The address list is configured in the General Settings Tab > EMAIL SETTINGS tab.

3.5.1.4 Override Schedules

Override Schedules temporarily disable Scheduled Tasks that are set to *Normal Schedule* in the Run Mode column. It enables any Scheduled Tasks that are set to *Override Schedule*. Tasks that are set to *Always* will not be effected by any Override Schedules. Override Schedules are especially helpful for creating holiday schedules. Override Schedules appear in a table on the Control/Logic Setup tab and can be created, edited, or deleted as needed. The module supports up to 20 different override schedules.

A task running on a Normal schedule can be setup by an Administrator (with *Administrator* credentials) and can be used, for example, to open door locks during office hours. Override schedules allow the normal schedule to be temporarily disabled without disturbing the underlying schedule. For example, an employee (with *Manager* credentials) can temporarily change the schedule to accommodate a special holiday without the need to change or adjust the Normal schedule.

An example of this would be the 1st of January. An override schedule could be created that starts 00:00 on January 1st and ends at 23:59 on January 1st. This override schedule would stop normally scheduled tasks from running every year on the 1st of January.

Override schedules are periodic annually (e.g., the event will occur every year at the same time.)

To add an Override Schedule, click *Add Override Schedule*. Click the *Edit* button to edit an override schedule, click the "X" button to delete an override schedule.
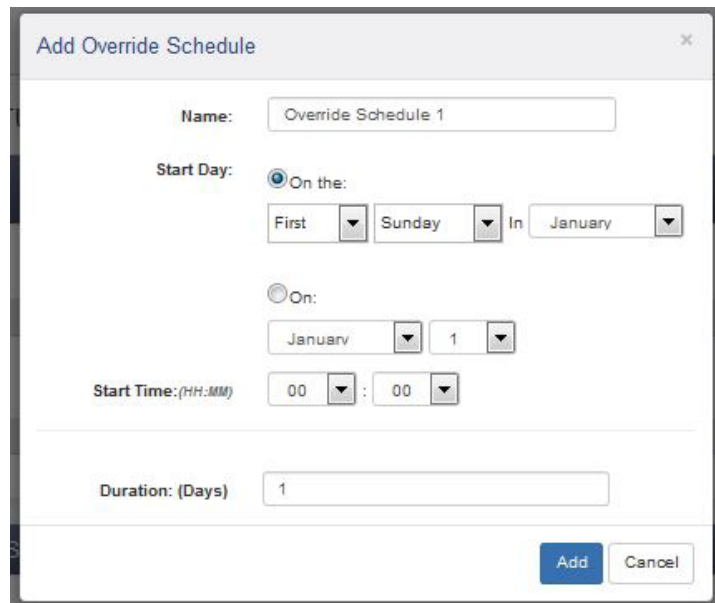
Name:

This text field is used to describe the override schedule. By default it is set to "Override Schedule 1". This field may be up to 24 characters long. Set the name to a descriptive value such as "Office Closed".

Start Day/Time:

Different options will appear depending on the option you choose:

▪ Choose "On the" to enter a start day of the month, start time, and duration (days) for the override schedule.

▪ Choose "On" to enter a specific start date and time, and a specific end date and time.

3.5.2   Control/Logic Tab > BASIC SCRIPT

The module can run simple custom programs written in a modified version of BASIC. This page is used to load and execute these programs. Before it can be loaded to the module, a script must first be prepared as a text (.txt) file. The file must then be uploaded to the module via the Basic Script tab.

The maximum size of script that can be uploaded to the device is 4K bytes. Information on writing a basic program for the module can be found in *BASIC ScriptsBASIC Scripts*. See Control Logic Examples Using BASIC Scripts for examples of BASIC scripts.

Note: An ASCII standard text file format should be used, such as Windows Notepad, Programmer's Notepad, vi, or other text editor that output the file as a .txt. Rich Text Format (.rtf) used by Microsoft WordPad is NOT compatible.

Currently Loaded Script
This field displays the .txt file that is uploaded to the module. The script (text) displayed on this screen cannot be edited. In order to edit any script, the .txt file must be edited externally and uploaded to the device again.

Basic Script File
To upload a basic script file to the device, click *Basic Script File*. A file upload dialog box opens which allows you to choose the file to upload. To upload the script, click *Open*.

Interpreter Status
This field displays whether the program is continuing to run, has stopped or finished, or if there are errors contained in the script. If there are errors in the script, the line on which the error occurred is displayed. Clicking the Refresh Status button will update this field.

Run Script
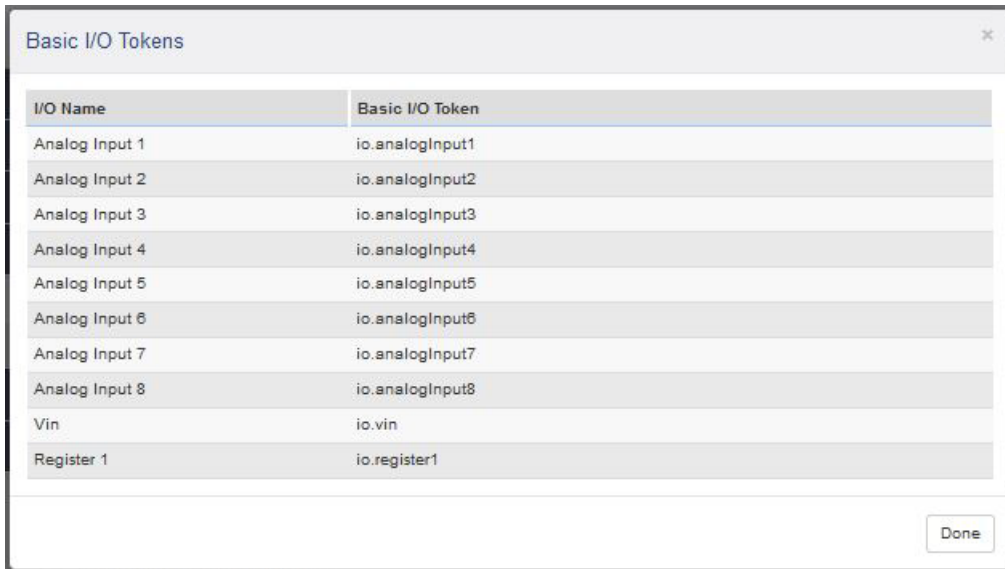
This option selects whether or not the selected script will run after it has been uploaded. The Submit button must be clicked to run the script.

BASIC I/O Tokens

After I/O resources are added and named, they can be controlled and monitored within BASIC scripts. BASIC scripts must reference I/O resources (relays, temp sensors, digital and analog inputs, etc.) in the form of a token, which is the characters "io" followed by the name of the resource (e.g., io.name). If the resource name has embedded spaces, they must be removed in the io.name token. The first character must be lower case.

For example, if a relay is named "Warehouse Fan", the fan can be turned on with the BASIC statement "LET io.warehouseFan = 1". If a resource name is changed, the resource name in the BASIC script must be renamed to match.

The button, "View Basic I/O Tokens" displays a list of all of the I/O resources currently available together with the equivalent token to be used in BASIC scripts to reference the respective I/O resource.

I/O Name	Basic I/O Token
Analog Input 1	io.analogInput1
Analog Input 2	io.analogInput2
Analog Input 3	io.analogInput3
Analog Input 4	io.analogInput4
Analog Input 5	io.analogInput5
Analog Input 6	io.analogInput6
Analog Input 7	io.analogInput7
Analog Input 8	io.analogInput8
Vin	io.vin
Register 1	io.register1

Basic I/O Tokens ✕ ... Done

Note: This screenshot was taken from the X-410 device. Other ControlByWeb devices will show different I/O Names and Basic I/O Tokens.

3.6 Logging Tab

Record data such as changes in I/O state, sensor data, and events. Both periodic and event-based logging are supported. The logged data is stored in internal nonvolatile memory and can be retrieved by entering the command http://{module's IP address}/log.txt. For more information on logging, see Section Log Files.

The module can make up to 30 log entries per second with event-driven logging. Each log entry has a time stamp with millisecond resolution. Two Conditional Task actions are available to pause and resume logging.

The log is stored in non-volatile, flash memory using a circular buffer (oldest data is over written). On all X-400 series devices 3072kB of memory space is reserved for logging, except for the X-410 which has 512 kB of memory space reserved. Log entries are composed of the following components, 30-byte header, 16-bytes for relays, 16-bytes for inputs, and 4 bytes for each analog value being logged (counters, 1-Wire sensors, vin, registers, etc). The space allocated for relays and inputs does not change depending on whether any relays or inputs are being logged.

Note: Changing the log settings will erase the current log file.

3.6.1 Logging Tab > GENERAL LOGGING SETTINGS

Enable Logging
When this option is set to Yes, the module will record data as configured on this page. The default setting for this option is *No. **Note:** This option controls data logging, but not system logging. System logging is always enabled.*

Start Time
If a logging interval is specified (periodic logging rather than event logging), logging will occur relative to this start time. For example, if the start time is 01:00 and the Interval is 6 hours, logging will occur at 01:00, 07:00, 13:00, and 19:00. Start time is specified in 24-hour time format. *Note: If "Event Only" is selected, this setting has no effect.*

Interval
This field is used to specify the time period of logging. A numerical value is entered into the text field, and the unit of time is selected using the adjacent radio buttons. The range of values in this field is 1-60. Time units are *Minutes*, *Hours*, and *Days*. Select *Event Only* to disable periodic logging. *Note: If "Event Only" is selected, this setting has no effect*.

Next Log Time
This field shows when the next periodic log event is scheduled to occur. If logging is disabled, the next log time will indicate "Disabled." If logging is enabled, but periodic logging is disabled (by selecting *Event Only*) the next log time will indicate "Event Logging Only."

Note: This information is updated only when the page is refreshed.

3.6.1.1 Logging Tab > LOG I/O

This setup page is used to select which resources are logged, and which resources asynchronously trigger logging.

Whenever a log event (triggered or periodic) occurs, each resource which has its Log check-box enabled will be included in the periodic log. If a resource has its *Trigger Log Event* check-box enabled, a log will asynchronously occur when that resource changes its state or value.

Trigger and Delta
Analog resources log data when the value changes by the amount specified in the *Delta* fields. The default setting for these fields is shown below.

Sensor	Trigger Delta
Vin	2.0 V
Registers	2
Temperature	2
Humidity	2%

Asynchronous Log Events

Digital Inputs <u>or</u> Relays	Logging will occur whenever the stat of that I/O changes.
1-Wire Sensors	Logging will occur whenever the sensor changes by the trigger delta.
Registers	Logging will occur whenever the Register changes by the trigger delta.
Vin	Logging will occur whenever the Vin voltage changes by the trigger delta.

3.6.1.2 Diagnostic Settings

XML, MODBUS, and SNMP requests received by the module can generate a log entry. Each communication protocol has a check-box to enable logging for that protocol.

Exclude Logged Requests
When logging is enabled for XML Requests, Modbus Requests, or SNMP Requests, you can filter out *Reads* (such as reading inputs' states via an XML request) or *Writes* (such as changing the output state via an XML command).

On the *Control Page Setup* tab, you can specify the refresh rate of the Control Page. Each time the page is refreshed, an XML request is sent and logged as a Read. Since the default refresh rate is 3 seconds, the log file can get cluttered by many XML request logs. By filtering out *Reads*, refresh requests will not be logged. Likewise, by selecting *Writes*, write requests will not be logged. The default selection is *None*.

3.6.1.3 Send Log File
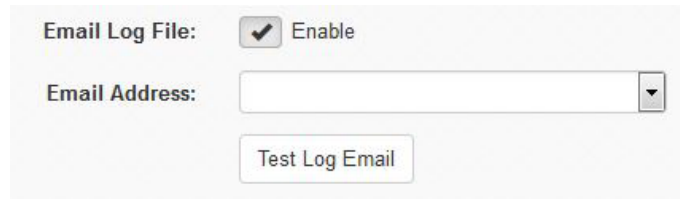
The module can be configured to send its log file to a remote server at a specific time once per day. The file can be emailed and/or uploaded to an FTP server. Only new log entries since the last successful upload, are sent. The log file is not deleted.

Daily Send Time
Select the time each day (24-hour format) the log file is to be sent.
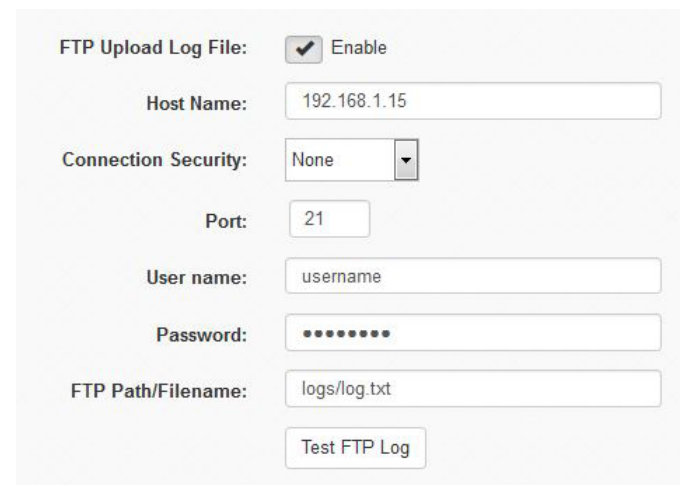
Email Log File
If this check-box is enabled, a log email will be sent daily. The log email will contain entries that are new since the last successfully sent log email. When this box is checked, a drop-down box and Test Log Email button appear. Using the drop-down box, select the email address to send daily log. The address list is configured in *General Settings Tab > EMAIL SETTINGS*. The *Test Log Email* button causes the log file to be emailed immediately, without waiting until the daily send time.

FTP Upload Log File
If this check-box is enabled, the log file is sent to an FTP server once each day. Only the log entries that are new since the last successfully FTP upload will be sent each day.

Host Name
The IP address or hostname of the remote FTP server is entered here. The default value is 192.168.1.15

Connection Security
Choose *None*, *Implicit TLS*, or *Explicit TLS*. The security and port number must be configured to match the FTP server. The *None* option uses no encryption on the connection. *Implicit TLS* uses SSL to encrypt the channel before any FTP commands are sent or received. *Explicit TLS* allows the connection to start unencrypted and then upgrades the connection to encrypted through FTP commands.

Port
The TCP port number of the FTP server. This must match the TCP port set in the FTP server. The valid range is 1 to 65535. The default port number is 21.

User name
The username for connecting to the FTP server.

Password
If the FTP server requires a password, the password is entered here.

FTP Path/Filename
When uploading the log file to an FTP server, the FTP server will have a default location where it will place the log file. This field is used to define the filename of the log file stored on the FTP server. Sub-directories can also be defined here.

The actual filename used for saving the file on the FTP server will be the specified file name with the date and time appended to it. For example log.txt would be log_YYYYMMDDHHMMSS.txt where YYYY is the year, MM is the month, DD is the day of the month, HH is the hour, MM is the minutes, and SS is the seconds. This date and time comes from the module, and is the time that the module uploaded the file.

Test FTP Log
The *Test FTP Email* button causes the log file to be sent immediately. If nothing is in the log file, no file will be sent. The log information sent when this button is pressed will not be sent again when the button is pressed again or at the daily upload time.

3.7 Monitor and Control Tab

Under this tab there are buttons for viewing and customizing the Control Page as well as graphing the log file.

3.7.1 Monitor & Control Tab > CONTROL PAGE

This button displays the current Control Page. The Control Page is what users typically use to control and monitor I/O on this device. You can access the Control Page either through the setup menu, or directly by entering the IP address of this device into the address bar of a web browser. The Control Page is highly configurable. The Control Page Setup tab determines which resources are shown on the Control Page and how they are presented and displayed.

Here's an example of a Control Page of an X-400 module:

This example shows an X-400 with a temperature sensor, register (named Master Relay Switch), and eight relays that were added from an X-12s expansion module.

Note: If a sensor is disconnected or fails, the data value will be shown as x.xx in yellow.

3.7.2 Monitor & Control Tab > CONTROL PAGE SETUP

The content, format, and presentation of the Control Page is configured on this page. For example, the settings below illustrate the settings for the Control Page shown in the previous section.

After making changes to the header, logo, footer, and control ordering, you must click the *Submit* button for the changes to take effect.

Note: Any browsers currently viewing the Control Page while making changes will need to be refreshed before seeing the new changes. Do this by clicking the refresh button on your browser.

Header Text
The text entered here appears at the top of the *Control Page*. It also appears in the header of the email text when the email notification is used. This field can be up to 30 characters in length.

Show Header Logo
A graphic logo can be displayed in the upper left-hand corner of the Control Page. This setting enables or disables displaying the logo.

Header Logo
The *ControlByWeb* logo is shown by default. To upload a new logo, click *Choose New Logo*. A file upload dialog box opens, allowing you to browse for the file on your computer. The file must be a *.png format. Click the button to initiate the file upload. The logo file size must be less than 12k bytes. Once a logo has been uploaded, the default ControlByWeb logo will be overwritten. The ControlByWeb logo will not be restored if the device is reset to factory defaults (there is only room for one logo.)

Footer Text/HTML
The bottom of the Control Page has a configurable "footer" field. The text entered in this setup box appears as a footer on the Control Page. The text can include HTML hypertext links as shown in the example.

Refresh Rate
The Control Page continually updates its contents by setting a timer in the web page that causes it to be reloaded at a specified time interval. The web page content will be refreshed at the time interval specified in this setting. It can be set from 1 to 60 seconds. The default Refresh Rate is 3 seconds. Slower refresh rates will use less network bandwidth.

Control Page Widgets

A list of I/O resources that can be added to the Control Page appear next. The "widgets" appear on the Control Page in the same order as the setup list. Resources such as analog inputs automatically appear in this list and can be removed from the Control Page, if desired. For all other I/O (registers, counters, timers, etc.), you must have previously added them in I/O Setup tab before adding them to the Control Page.

To add one or more widgets, click the *Add I/O to Control Page button*

To edit a widget, click the *Edit* button

To remove a widget, click the *Delete* button.

To re-arrange the order of the widgets, click the widget handle and drag the widget to a different row.

When finished, click to *Submit* at the bottom of the page to save your edits.

To change the name of the resources (analog inputs, registers and Vin) on the Control Page, edit the respective settings in the *I/O Setup Tab* and change the resource name. While naming each resource, it is good practice to edit the associated status text. For example, for a digital input perhaps name the input "Warehouse Door", set the *On Status Text* to "Door is Open" and the *Off Status Text* to "Door is Closed".

Some widgets have more setup options than others. Widgets for the analog inputs, Vin and timers can be added but have no options to edit.

Once the I/O has been added, rearrange the I/O by clicking on the icon and dragging to the desired position.

Click *Submit* once you have the finished making changes to the page.

Add I/O to the Control Page

To add one or more I/O, click the Add I/O to *Control Page* button. From the list of available I/O resources, check one or more *Add* boxes to add the selected resources to the Control Page.

After adding the desired widgets you can edit the widgets one by one, as needed, for the desired appearance and function. The visual presentation of the labels, buttons and colors can be highly configured. The display options available for the widgets depend on the specific I/O.

Note: Certain I/O resources (analog inputs, 1-Wire sensors, etc.) do not have edit buttons/widgets on the Control Page Setup tab; however, the I/O information can be configured in the I/O Setup tab.
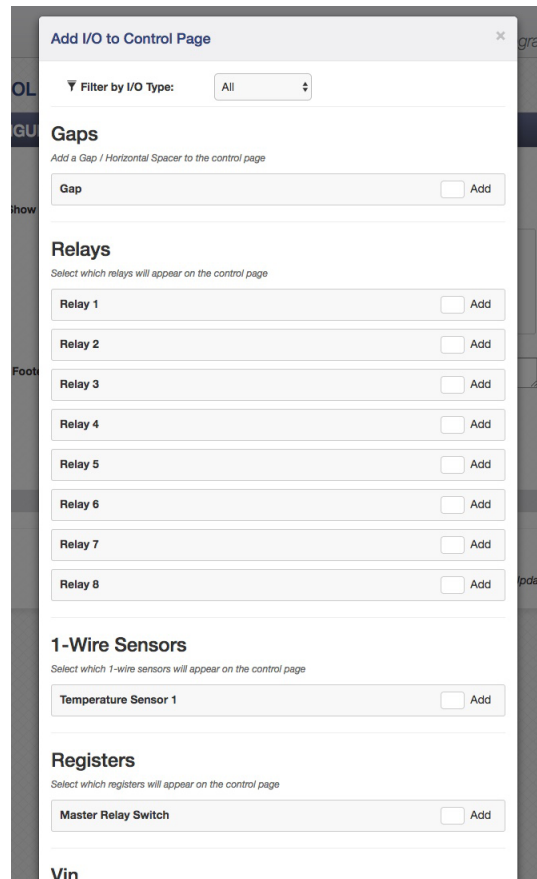
Gap Widgets

Gap widgets are cosmetic widgets that allow sections of the Control Page to be separated by a small space. This is useful when grouping similar controls together.

1-Wire Sensor Widget Editor

There is no widget editor for 1-Wire sensors. The information which will appear on the Control Page will be configured in the I/O Setup tab > 1-Wire Sensors.

Relay Widget Editor

State
Show or Hide the relay state.

On Status Color
This setting specifies the color that will be displayed on the Control Page when the relay is On. Options are Red, Green, Yellow, Blue and White. The default color is Green.

Off Status Color
This setting specifies the color that will be displayed on the Control Page when the relay Off. Options are Red, Green, Yellow, Blue and White. The default color is Red.

ON Button
Show or Hide the On button.

ON Button Label
The text in this field specifies the text that will be displayed on the ON Button on the Control Page to turn on the relay. Up to 16 characters may be entered in this field. The default text is "On".

Off Button
Show or Hide the Off button.

OFF Button Label
The text in this field specifies the text that will be displayed on the OFF Button in the Control Page to turn off the relay. Up to 16 characters may be entered in this field. The default text is "Off".

Toggle Button
Show or Hide the Toggle button.

Toggle Button Label
The text entered in this field appears in the 'Toggle' button. Up to 16 characters may be entered in this field. The default text is "Toggle".

Pulse Button
Show or Hide the Pulse button.

Pulse Button Label
The text entered in this field appears in the 'Pulse' button. Up to 16 characters may be entered in this field. The default text is "Pulse".

Digital Input Widget Editor

State
Show or Hide the state (value).

On Status Color
This setting specifies the color that will be displayed on the Control Page when the input is considered On. Options are Red, Green, Yellow, Blue and White. The default color is Green.

Off Status Color
This setting specifies the color that will be displayed on the Control Page when the input is considered Off. Options are Red, Green, Yellow, Blue and White. The default color is Red.

Input Counter Widget Editor

Count/Frequency
Show or Hide the counter/frequency value.

Allow reset in control page
When enabled, the count value on the Control Page may be clicked to reset the counter's value back to 0 after confirmation.

Analog Input Widget Editor

There is no widget editor for analog inputs. The information which will appear on the Control Page will be configured in the I/O Setup Tab > Analog Inputs.

Register Widget Editor

Registers can be changed externally through the Control Page, XML requests, ModBus, or the BASIC script. This allows a BASIC script to react to user input. These variables are considered to be floating point numbers, just like other variables in the BASIC script.

Status:

State
Show or Hide the state (value).

Control Buttons:

There are four different types of control buttons for a register widget, each types has different options:

1. Type = Increment/Decrement

Increment Button
Show or hide the button

Decrement Button
Show or hide the button

Step
This setting is the amount the Register will be incremented or decremented when the respective increment/decrement button is clicked. The value can be an integer or floating point number.

2. Type = Custom Buttons

This setting provides up to 4-buttons with custom labels and set-values. Use one or more of these buttons to force a Register to specified fixed values.

Use these buttons for example, to provide a fan speed control with buttons labeled High, Medium and Low and perhaps with 100%, 50% and 20% for the button setvalues.

Button (1-4)
Show or Hide the button.
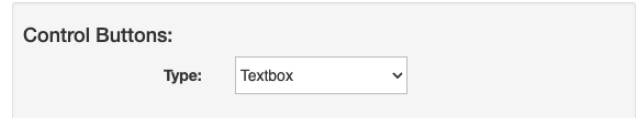
Label (1-4)
The text in this field specifies the text that will be displayed in the button. Up to 16 characters may be entered in this field.

Value (1-4)
When the user clicks the button, the Register defined in the Widget being edited will be set (forced) to the specified fixed value. The setting can be any integer or floating point value.

3. Control Button Type = Text Box

Use this setting to allow the user to manually change the value of a Register. The input can be an integer or floating point value. The input value can be restricted between minimum and maximum values defined by settings for the Register itself. See *Section* I/O Setup Tab > Registers

Use this input field for example, to enter a thermostat set-point value.

4. Analog I/O Status Colors

The status colors of analog I/O (analog inputs, 1-Wire sensors, thermocouples, Vin, high-frequency inputs, analog outputs, timers, counters, on timers, total on timers) can be changed by creating Tasks under Control/Logic which set the I/O status colors.
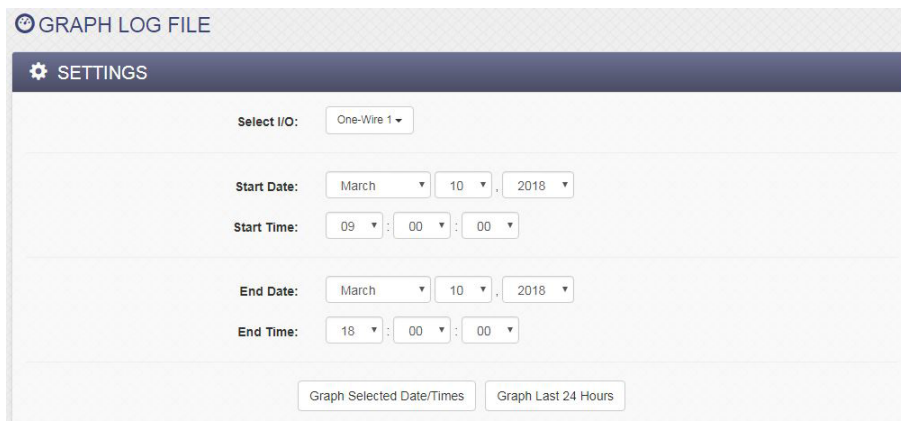
5. Digital Input and Relay Status Colors

The status colors of digital inputs and relays can be configured directly in their respective control page widgets.

3.7.3 Monitor & Control Tab > GRAPH LOG FILE

The data in the log file can be displayed in a graphical format. Use this page to configure and view the graph. With the *Start* and *End* settings you can "zoom in" to display the data over a specific time interval. When the Graph is first loaded, the last 24 hours of the log file are downloaded and displayed. Pressing the "Graph Last 24 Hours" button will update the graph with the last 24 hours of the log file. Pressing "Graph Selected Date/Time" will update the graph with the selected range of time in the Start/End drop-down menus. Note: The larger the range of time, the longer it takes to download the log file for graphing.

3.7.3.1 Settings

Select I/O
Select which I/O from the log file is to be graphed. Up to 10 I/O can be graphed at one time.

Start Date & Time
This is the start time for the graph. Select the month, day, year and time of day using the drop-down boxes.

End Date & Time
This is the stop time for the graph. Select the month, day, year and time of day using the drop-down boxes.

3.7.3.2 Graph

The graph includes a legend describing the colors and I/O in the top-right corner of the graph. The graph is interactive in that if the mouse is hovered over a specific data-point, it displays the precise value. To zoom in on a specific region of the graph, click and drag to highlight the area of interest. To return to a normal zoom, click the Graph button.

Section 4: Control Page

The Control Page is normally what users see and use and it can be highly configured to fit your needs. The Monitor & Control Tab > CONTROL PAGE SETUP tab has settings that determine what resources are displayed and how they are formatted.

Administrators, *Managers* and *Users* have separate access privilege to the Control Page. See Error: Reference source not found for a description of each access privilege.

4.1 Browser Operation

There are two ways users can access the Control Page by using a web browser:

1. Control Page

The first is by typing the IP address of the module directly into the web browser address bar. For example, using the default IP address, the user would enter http://192.168.1.2 (If the IP address was changed from the default, the user must use the new IP address.)

Note that if any port is used other than the default port 80, the port must also be included in the request. For example, accessing the unit at port 8000 would be as follows: http://192.168.1.2:8000

To access the module using HTTPS, the user would enter: https://192.168.1. 2

If any port is used other than the default HTTPS port 443, the port must also be included in the request. You will usually get a warning when accessing the module over HTTPS, this is because the module is using a default, self-signed SSL certificate.

The I/O on the Control Page updates every three seconds unless the update interval has been changed in the Control Page Setup.

2. Control Page via Setup Page Tabs

The second method of accessing the Control Page is through the Setup Page tabs (http://192.168.1.2/setup.html).

Choose Monitor & Control Tab > CONTROL PAGE.
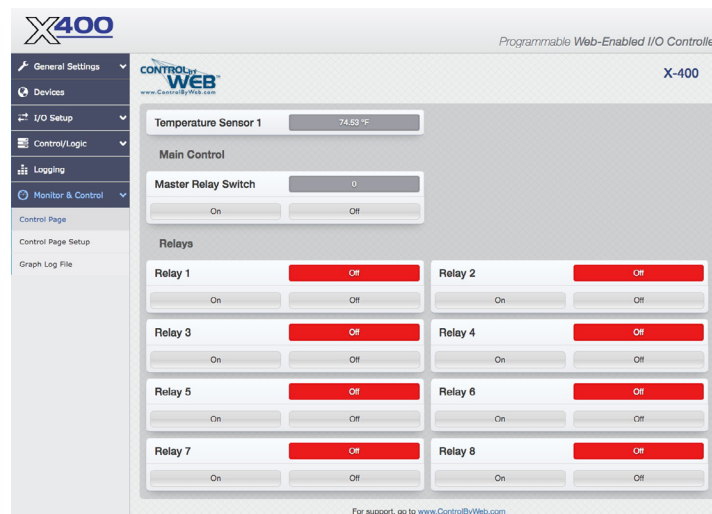
Section 5: Example Control Scenarios

This section presents various control examples using *Scheduled Tasks*, *Conditional Tasks*, and BASIC scripts to solve example applications.

5.1 Control Logic Examples Using Tasks/Functions

The following examples illustrate tasks or functions that can be performed by the module. These examples may be used as a tutorial to illustrate how to perform certain functions and can be a good starting point for more advanced logic.

<u>**Important:**</u> *When creating tasks that perform actions when something is triggered, the action will not automatically reverse when the trigger goes away. For example, if a trigger turns a remote digital output on when temperature rises above 50 degrees, you must create a second task to turn the digital output off when the temperature falls below 50 degrees.*

5.1.1 Send an email if the temperature is less than 33 °C between 9:30PM and 10PM each day

In this example the Deadband of 1 degree was used to prevent multiple email messages being sent due to "chatter".

Another email message won't be sent until the temperature rises above 34 degrees (33 plus 1 degree deadband) and then back below 33 degrees.

The "During" section of the task limits the time when the triggers are effective to 9:30 P.M. and 10:00 P.M. every day of the week. Outside this window email messages will not be sent even if the temperature drops below 33 degrees.

Add Conditional Task

Task Name: Send email if low temp

If:

TRIGGER

Triggers ONLY occur when conditions change to true. The logic operations are event driven, and are not combinational. Conditions must change to false and back to true to re-trigger.

Condition 1: Outdoor Temperature

Value: < 33.0 C

Deadband: 1.0 C

And

Condition 2: None (Optional)

During: Always | Set Time

Start Time: (HH:MM:SS) 21 : 30 : 00

End Time: (HH:MM:SS) 22 : 00 : 00

Day(s) of Week: Su M Tu W Th F Sa

Then:

ACTIONS

Actions only occur at the moment the trigger status changes.

Set Action 1: Send Email

Send Email To: None selected ▾

Set Action 2: None (Optional)

Set Action 3: None (Optional)

5.1.2 Send Email if Input 1 stops toggling

Monitor Pseudo Digital Input 1 for state changes. If Input 1 doesn't change state every 5 seconds or faster an email message will be sent.

This is using two tasks. Task 1 starts (or re-starts) a timer each time the pseudo input changes. Task 2 sends the alert if the timer ever expires.

5.1.3   Send an email message every 24 hours

5.1.4 Network monitor between two X-400 Series modules (advanced)

In this example we will configure two X-400 devices to share a register. In the example we are using this as a network monitor, but this example illustrates how to share register or I/O states between modules.

One X-400 (we will call X-400_ONE) will be configured to share a register with a second X-400 (we will call X-400_TWO). X-400_ONE will change that register value once every 10 seconds. X-400_TWO will monitor that register value. If X-400_TWO detects no changes to the register for more than 12 seconds it will send an email alert.

Setup X-400_ONE communications with X-400_TWO:

1. Connect both devices to the network and assign IP address to each of them.
2. Open the setup page for X-400_ONE and select "Remote Devices" tab. Click the "Find Devices" button. The other X-400 (X-400_TWO) should appear in the list. If multiple X-400 units are installed on the network you may need to identify it by serial number. Select that X-400 device so the "Add Remote Device" window appears. Within that window...

   - Name the device "X-400_TWO"
   - Select Model X-400 (should already be selected)
   - Serial number should be filled in with correct serial number.
   - IP address should have correct IP address for X-400_TWO and port should be correct.
   - Set up the security options and enter passwords for X-400_TWO
   - In the bottom portion of the window, select "Instant Send"
   - Click "Add Device"

Wait a minute and then you should see X-400_TWO in the Remote Device List and the status should indicate a response time (in milliseconds) from the remote unit (the smaller the response time the better).

Now X-400_ONE is set up to push its I/O state to the remote device. By default it will push the state every 3 seconds (PUSH interval) but we want it to push its state instantly when the register changes so we will set that up in Conditional Tasks.

Setup of X-400_ONE to toggle register and share it with X-400_TWO (add two Conditional Tasks):

1. Open setup page for X-400_ONE (should already be there from above setup)
2. Set up a timer (called Timer 1) with a Power Up value of 8 seconds. This timer will be used to change the register and the Power Up value will cause it to start automatically.
3. Set up a task and call it "Toggle Register to 1". Set the trigger to "Timer 1 Expires AND Register 1 = 0". Set the Actions to "Set Register 1 to 1" and "Start Timer 1 for 10 Seconds" and "Push I/O State To Remote Receiver Device"
4. Set up a task and call it "Toggle Register to 0". Set the trigger to "Timer 1 Expires AND Register 1 = 1". Set the Actions to "Set Register 1 to 0" and "Start Timer 1 for 10 Seconds" and "Push I/O State To Remote Receiver Device"

Setup of X-400_TWO to monitor the register from X-400_ONE and sent alert if the register doesn't change within 12 seconds:

1. Open setup page for X-400_TWO and select Remote Devices tab. You should see X-400_ONE in the list of devices but the name will show as the serial number rather than the name X-400_ONE. You can click edit and change the name to X-400_ONE. If the X-400_ONE device doesn't appear in the list you will need to add it to the list manually.

2. Once the X-400_ONE device appears as one of the Remote Devices you will need to add its register to the local I/O. Click on I/O Setup, then Registers. At the bottom of the table, click on the button called "Add Remote Register". Add the register "X-400_ONE Register 1" to the list. You should now see this new register listed under "Remote Register".
3. We need a timer to go to I/O Setup and create a timer called "Timer 1". Give it a power up value of 30 seconds to make sure both X-400 units have plenty of time to boot before sending error messages.
4. Now add the logic. Click on "Control/Logic" and create a new Conditional Task. Call this task "Set Timer When 1". Set trigger Condition 1 to "X-400_ONE Register 1 = 1". Set Action 1 to "Start Timer" with a time of 12 Seconds.
5. Add a second Conditional Task. Call this task "Set Timer When 0". Set trigger Condition 1 to "X- 400_ ONE Register 1 = 0". Set Action 1 to "Start Timer" with a timer of 12 Seconds.
6. Add a third Conditional Task. Call this task "Send Alarm". Set trigger Condition 1 to "Timer 1 Expires". Set Action 1 to "Send Email" and specify the email address where the message should go (if no email addresses appear you will need to set them up first).

5.2 Control Logic Examples Using BASIC Scripts

Most advanced logic tasks can be accomplished using the module's Task Builder; however, the module has a BASIC interpreter that can be used for more advanced tasks.

I/O resources are not fixed on the module. After registers and other I/O are added, they can be used within BASIC scripts. BASIC scripts must reference I/O resources such as digital I/O in the form of io.name where "name" is the resource name defined for each I/O under the I/O Setup Tab.

If the resource name has embedded spaces, they must be removed in the io.name statement. The first character must be lower case. For example, if a digital I/O is named "Warehouse Fan", the fan can be turned on with the BASIC statement "LET io.warehouseFan = 1". If a resource name is changed during development and testing, the resource name in the BASIC scrip must be renamed to match.

5.2.1 If an analog input is in the alarm state, send an email every hour

```
DO

'If 2.5 < sensor1 < 4.5 then send an email every 1 hour

IF io.temp1 < 4.5 THEN
    IF io.temp1 > 2.5 THEN
        IF t0 = 0 THEN
            EMAIL
            LET t0 = 36000              '3600 seconds
        END IF
    END IF
END IF

LOOP
END
```

5.2.2 Monitor 4 doors, send an email if a door is open more than 5-minutes

Send an email if a door has been open for more than 5 minutes, repeat the email every 5 minutes thereafter while the door is open. Note, we use the digital input mode of the analog inputs for this scenario.

```
'Send an email alert after a door has been open for more than 5 minutes.
'Continuously set a timer for 5 minutes if the door is closed

'Setting initial timer values
'Using variables a-d allow simple changes to alarm times
LET a = 3000        'Input1 300.0 seconds
LET b = 3000        'Input2 300.0 seconds
LET c = 3000        'Input3 300.0 seconds
LET d = 3000        'Input4 300.0 seconds

LET t1 = a
LET t2 = b
LET t3 = c
LET t4 = d

'Begin main program, sequentially service each door.
DO

'If door1 is closed, then set timer for 5 minutes.
'If door1 is open, then send an email and reset timer after timer expires

'Door 1
IF io.analogInput1 = 1 THEN
    LET t1 = a
ELSE                    'Else if door is open
    IF t1 = 0 THEN
        EMAIL io.analogInput1
        LET t1 = a
    END IF
END IF

'Door2
IF io.analogInput2 = 1 THEN
    LET t2 = b
ELSE                    'Else if door is open
    IF t2 = 0 THEN
        EMAIL io.analogInput2
        LET b
    END IF
END IF

'Door3
IF io.analogInput3 = 1 THEN
    LET t3 = c
ELSE                    'Else if door is open
    IF t3 = 0 THEN
        EMAIL io.analogInput3
        LET t3 = c
    END IF
END IF

'Door4
```

```
IF io.analogInput4 = 1 THEN
    LET t4 = d
ELSE                            'Else if door is open
    IF t4 = 0 THEN
            EMAIL io.analogInput4
            LET t4 = d
    END IF
END IF

LOOP
END
```

5.2.3 Send an email if the AC power fails

Send an email if the AC power (via input1) has been off for 60 seconds. Send a follow-up email when the power is restored. Send the power restored email only if a 'power off' email was previously sent.

Note: A simple AC to 12VDC wall transformer connected to the digital input can be used to detect the loss of AC power.

```
'Script will send an email after power has been off for a specific amount of time.
'Will also send an email when power is on only if a 'power off' email has been sent.

LET a = 0 'Power on email sent if true
LET b = 0 'Power off email sent if true

DO

IF io.input1 = 1 THEN                    'If power is on
    LET t1 = 600
    IF a = 0 THEN                   'If no 'power on' email has been sent
            IF b = 1THEN            'If 'power off' email has been sent
                    EMAIL io.input1
                    LET a = 1       'Set power on email sent flag
            END IF
    END IF
    LET b = 0
END IF


                                    'Once power has been off a specific time
IF t1 = 0 THEN
    IF b = 0 THEN
            EMAIL io.input1
            LET b = 1               'Set power off email sent flag
    END IF
END IF

LOOP
END
```

5.2.4 Monitor a generator. Send an email if it runs for more than 10 seconds.

Send an email if a generator runs for more than 10 seconds, and every 30 minutes afterward. Send an email when the generator turns off if a 'generator on' email was sent.

```
'Input1 is used as the source for the generator status
'Input1 = 1 : Generator off
'Input1 = 0 : Generator on

LET a = 0                      'Generator on email sent flag
LET t1 = 100                   '10.0 seconds

DO

'Generator off
IF io.input1 = 1 THEN
     LET t1 = 100                   '10.0 seconds
     IF a = 1 THEN                   'If generator on already email sent
          EMAIL io.input1           'Email everything okay
          LET a = 0                 'Set generator on flag back to false
     END IF
END IF

'Generator on and timer expires
IF t1 = 0 THEN                  'If timer has expired
     EMAIL io.input1            'Email generator on
     LET t1 = 18000            '30 minutes until another email
     LET a = 1                 'generator on email flag
END IF

LOOP
END
```

Section 6: Auxiliary Operations

6.1 XML

Custom computer applications may be created to monitor and control the module. This method does not use a web browser. There are two XML pages that can be used to monitor and control the module: state.xml and customState.xml.

All requests to the XML files must be in the form of HTTP GET requests see *Section HTTP GET Requests (for custom applications)* for details.

See *Section External Server and Remote Services* for more information on network configurations when using XML.

6.1.1 state.xml

The state.xml page served by an X-4xx only shows the I/O that have been assigned a 'Local I/O Number'. The XML tag names on this page represent the I/O type and Local I/O number on the X-4xx. For example, if two relays have the Local I/O Numbers 1 and 5 assigned and no other I/O on the X-4xx have a Local I/O Number assigned, then the state.xml page would appear as follows:

```
<datavalues>
        <relay1>0</relay1>
        <relay5>0</relay5>
        <vin>12.2</vin>
        <utcTime>1262394580</utcTime>
        <timezoneOffset>-25200</timezoneOffset>
        <serialNumber>00:0C:C8:00:00:00</serialNumber>
</datavalues>
```

Notice that the power supply of the X-4xx 'vin' always appears in state.xml, as do the time, timezone, and serial of the X-4xx.

Reading state.xml

XML Monitor All Functions
The state of the I/O configured on the X-4xx can be read by requesting state.xml. This can be demonstrated by entering the following URL into the address bar of a web browser (substituting the IP address as necessary):

http://192.168.1.2/state.xml

The following state.xml page is returned (this can vary, depending on the module and browser):

```
<datavalues>
        <digitalInput1>0</digitalInput1>
        <onTime1>0.00</onTime1>
        <totalOnTime1>0.00</totalOnTime1>
        <count1>0.00</count1>
        <frequency1>0.00</frequency1>
        <relay1>1</relay1>
        <relay2>0</relay2>
        <relay3>0</relay3>
        <relay4>0</relay4>
        <analogInput1>4.96</analogInput1>
```

```
        <analogInput2>4.96</analogInput2>
        <analogInput3>4.97</analogInput3>
        <analogInput4>4.97</analogInput4>
        <oneWireSensor1>x.x</oneWireSensor1>
        <vin>12.2</vin>
        <register1>0</register1>
        <utcTime>1262394580</utcTime>
        <timezoneOffset>-25200</timezoneOffset>
        <serialNumber>00:0C:C8:00:00:00</serialNumber>
</datavalues>
```

The numbers enclosed by the tags, <tag>, indicate the current state or value monitored by the module. Values for each tag are described in the table below.

Custom computer applications can open a TCP/IP connection with the module and send a GET request for the state.xml file to obtain the current state of the module.

XML Tags*	Monitor Values
<digitalInputX>	0=off (voltage not applied to input X) 1=on (voltage applied to input X)
<onTimeX>	Time in seconds how long the input was on since last coming on
<totalOnTimeX>	Total time in seconds how long the input has been on
<countX>	The count value associated with input X
<frequencyX>	The frequency associated with input X
<relayX>	0=off (coil off) 1=on (coil energized)
<digitalIOX>	0=off (voltage not applied to input X) 1=on (voltage applied to input X)
<analogInputX>	Value of analog input X
<vin>	Scaled internal Vin measurement
<frequencyInput>	Value of the X-420 frequency input
<registerX>	Value of the register X
<utcTime>	Current UTC time expressed in seconds since January 1st, 1970
<oneWireSensorX>	x.x = Indicates that the 1-Wire sensor could not be read 77.3 = Current sensor value 77.3 F = Current sensor value with units included using param showUnits=1
<timezoneOffset>	Value to offset utcTime for local time
<serialNumber>	00:00:00:00:00:00, serial number of X-418.

Controlling with state.xml

Commands can be sent to the module to control the I/O. The parameters used differ depending on if state.xml is used or customState.xml is used. State.xml uses the io type names and local io numbers when controlling the I/O. CustomState.xml uses the camelcase version of the user configurable name when controlling the I/O.

XML Digital Output Control

Commands are sent using the parameter composed of the io type and digital I/O number digitalIOX (X is replaced by 1 for digital IO 1, or 2 for digital IO 2, etc). A few examples of using digitalIOX are given here.

Command	Description
state.xml?digitalIO1=0	Turn Digital I/O 1 *OFF*
state.xml?digitalIO1=1	Turn Digital I/O 1 *ON*
state.xml?digitalIO2=0	Turn Digital I/O 2 *OFF*
state.xml?digitalIO2=1	Turn Digital I/O 2 *ON*

XML Pulse Digital Output

When the pulse command is sent, the output will turn *ON* for the Pulse Duration specified in the Digital I/O *setup page*. It is also possible to send a pulse time command that is different than the configured Pulse Duration. This is done by sending a pulseTime variable. The pulseTime variable does not change the Pulse Duration in the setup page and it is not stored or recorded. The pulseTime variable only changes the duration for the single pulse initiated by that command. In other words, you must issue the pulseTime command for each pulse command that differs from the preset value. For example, if the X-420 is configured for a pulse duration of 1.5 seconds. To issue one 1.5 second pulse, three 5 second pulses, and another 1.5 second pulse to digital I/O 1, you would issue the following commands:

Command	Description
state.xml?digitalIO1=2	Pulse Digital I/O 1 for the preset time (1.5 seconds).
state.xml?pulseTime1=5&digitalIO1=2	Pulse Digital I/O 1 for 5 seconds.
state.xml?pulseTime1=5&digitalIO1=2	Pulse Digital I/O 1 for 5 seconds.
state.xml?pulseTime1=5&digitalIO1=2	Pulse Digital I/O 1 for 5 seconds.
state.xml?digitalIO1=2	Pulse Digital I/O 1 for the preset time (1.5 seconds).

NOTE: The pulseTime argument MUST come before the digitalIO1=2 command.

XML Set On Time Counters

The on time counter and total on time counter values can be set using the onTimex and totalOnTimex commands respectively:

Command	Description
state.xml?onTime1=0	Reset the onTime1 counter to 0
state.xml?onTime1=5	Reset the onTime1 counter to 5
state.xml?totalOnTime1=0	Reset the totalOnTime1 counter to 0
state.xml?totalOnTime1=5	Reset the totalOnTime1 counter to 5

XML Set Counter

The counter values can be set using the countx command:

Command	Description
state.xml?count1=200	Set counter1 to 200.

XML Set Register Values

The register values can be set using the registerx command:

Command	Description
state.xml?register1=10.5	Set register1 to 10.5.

6.1.2   customState.xml

The customState.xml works very similarly to the state.xml page. The difference is in the customState.xml, the XML tag names are set to a name very similar to the name of the I/O it is associated with. For example, if you were to set the name of 'Register 1' to 'My Register 1' the register would show up in the customState.xml as 'myRegister1'. Using these names allows applications interfacing with the module to use more appropriate names when reading and writing I/O values.

Controlling I/O with the customState.xml is the same as with the state.xml except you may now use the names shown in the XML tags. For example, if you wanted to set a register named 'My Register 1' to the value 10, you would use the following command:

http://192.168.1.2/customState.xml?myRegister1=10

You will need to reference the customState.xml for the exact names to use for each of the configured I/O.

6.2 JSON

This device can also read the current state of the devices I/O through JavaScript Object Notation (JSON). The JSON format may be preferred for some custom applications as some languages provide native methods for processing data in JSON form.

There are two primary files that can be requested from this device in JSON form. The state.json and the customState.json. Each of these files provide the same information as their XML counterparts, but will be formatted with JSON. Additionally, the methods of controlling the I/O will be the same as with the XML pages. Note that all requests to the JSON files must be HTTP GET requests, see *Section HTTP GET Requests (for custom applications)*

6.3 HTTP GET Requests (for custom applications)

The module has a built-in web server that responds to HTTP GET requests. These GET requests are sent from web browsers when a web page is requested. They are fairly similar to the actual addresses seen in the browser's address bar. The following section describes the HTTP protocol and how it can be used to control and monitor the module. All examples below show the state.xml, but will be the same for customState.xml and the JSON files. It is also assumed that a TCP/IP connection has already been established from the custom application to the device. The following are the messages that should be sent over the TCP/IP connection to control and monitor the device.

6.3.1 Using GET for Control and Monitoring

No Password

GET requests to the device for XML files.

Example request state.xml:
>   GET /state.xml HTTP/1.1\r\n\r\n

Example set Register 1 to the value 25:
>   GET /state.xml?register1=25 HTTP/1.1 \r\n\r\n

Password Enabled

If the User account is enabled on the module, and the state.xml page is requested through a browser, the user will be prompted for a password. If the request is sent from custom application, the HTTP request will need to contain the password encoded as Base64.

The following is an html request header without the password:
>   GET /state.xml?register1=25 HTTP/1.1 (Terminated with two \r\n.)

The following example adds the password:
>   GET /state.xml?register1=25 HTTP/1.1 (Terminated with one \r\n.)
>   Authorization: Basic bm9uZTp3ZWJyZWxheQ== (Terminated with two \r\n.)

bm9uZTp3ZWJyZWxheQ== is the Base64 encoded version of the user "name:password," none:webrelay.

6.4 SNMP Requests, Objects and Community Strings

All configured I/O and some simple network parameters can be retrieved using Simple Network Management Protocol (SNMP). For most cases, using SNMP is as simple as locating the appropriate Management Information Bases (MIB) files and loading them into the SNMP manager software. The module will generate an MIB file, based on its I/O configuration, for use with the SNMP manager software. If the I/O are configured (added/deleted), a new MIB will need to be generated.

SNMP is configured under the *Advanced Network* setup tab. See *Section* General Settings Tab > ADVANCED NETWORK SETTINGS for more information.

The module supports the following Packet Data Units (PDU):

- GetRequest
- GetNextRequest
- GetBulkRequest
- SetRequest
- Trap
- Notification

6.4.1 Standard Objects

The module supports several standard RFC1213 objects that usually come with SNMP management software. If not, an Internet search for RFC1213-MIB will turn up multiple links.

RFC1213 Object	Response
system.sysDescr	X-418
system.sysObjectID	X418
system.sysUpTime	Time in hundredths of seconds since the device was last powered.
system.sysName	X-418*

*The sysName is customizable under the **Control Page Setup** tab. It's the Control Page Header.

6.4.2 Module Objects

All configured I/O on the module can be monitored and controlled through SNMP. The MIB file can be generated for the module by going to General Settings Tab > ADVANCED NETWORK SETTINGS and pressing the button Generate and Download MIB File. This file should be regenerated whenever there are changes made to the I/O.

6.4.3 TRAPS

Send SNMP traps when a relay changes state, when a particular sensor value is reached, or when the supply voltage is out of the desired range. Traps are configured as actions in Conditional and Scheduled tasks. As more I/O are added to the module, more I/O will appear in the MIB file.

6.4.4 Notifications

Supports sending of SNMP Notifications when the SNMP version is 2c or 3. Notifications are similar to traps accept they require a response to be sent back from the SNMP manager. Retries will occur if the SNMP manager does not return a response. This makes notifications more reliable than traps. Notifications are configured as actions in Conditional and Scheduled tasks.

6.4.5 Community Strings

The module allows customization of both the read and write community strings. The proper community string will be needed for all read and write requests. By default both read and write community strings are webrelay. Community strings are only used by SNMP versions 1 and 2c. SNMP version 3 uses a different security mechanism.

6.4.6 SNMP V3 User-Based Security Model

The module supports the SNMP V3 User-Based Security Model (USM). This replaces the community strings "security" of SNMP V1 and 2C. The details of USM can be complicated, but the main thing is that both the module's security settings and the SNMP Manager's security settings need to match for it to work.

There are two protocols used for USM. The first, authentication protocol, allows the SNMP manager to authenticate the module and vice versa. The second, privacy protocol, allows the SNMP communication to be encrypted. Each protocol has an "algorithm" and a password associated with it. There is also a security username that is shared by both protocols.

6.5 External Server and Remote Services

Note: The following methods are supported by the module; however, Xytronix Research & Design, Inc. does not provide or support custom third-party applications, or external web servers.

6.5.1 Accessing the module with custom software or third-party applications

Custom applications can send commands to the module for monitoring and control functions using HTTP requests for XML or JSON files. (See *Section XML* and *JSON* for more information) The application interface can be used to provide a custom user interface, access to multiple units in a single screen, and allow for automation, logging, and other application-specific features.

6.5.2 Using an External Web Server

Rather than accessing the module directly from a computer, an external web server can be used. The term "external" web server is used here to mean a separate web server (such as Apache, IIS, or NGINX) that is not the web server built into the module. In this scenario, users access custom web pages that reside on the external web server and the external web server communicates with the module.

An external web server can integrate multiple ControlByWeb devices into a single control page. In other words, the user may not be aware that he/she is using multiple ControlByWeb™ devices, but rather the user sees an integrated control page for the entire system. In addition, the use of an external web server allows programmers to create custom user interfaces that take advantage of the additional resources typically available on larger web servers, including more memory and various web programming languages.

There are two approaches that an external server can use to communicate with the module and other ControlByWeb™ devices: *Direct Server Control*, and *Remote Services*.

Direct Server Control

The first approach is for the external server to create a TCP connection whenever it needs to access the ControlByWeb device. In this case, the external server opens the connection, sends commands and/or reads the device, and closes the connection.

This method is ideal when the web server and all of the ControlByWeb devices are on the same network (without routers between them). In this case, the server can communicate with the devices directly and securely since data never has to leave the local network.

When the server and the ControlByWeb devices are on different networks, routers must be configured to allow appropriate access. If a public network is used, such as the Internet, security precautions should be considered.

Remote Services

The second approach is for the ControlByWeb device to initiate a connection using *Remote Services*. The settings under the *Advanced Network* tab in the setup pages will enable the device to open a TCP connection with an external server. Once the connection is open, the external server can send commands and/or read the device. The external server can leave the connection open (so that it never closes) or close the connection.

Remote Services is ideal for installations where the server and the device are installed on different networks. This is especially useful when each device is installed on a separate private network. For example, if the user doesn't control the network connections where the ControlByWeb device is installed, *Remote Services* would initiate a TCP connection over the Internet with the control computer. Since the ControlByWeb device initiates the connection, the control computer doesn't have to know the IP address of the device. This means that the device can be installed using DHCP. In addition, no special router configuration is required. This makes the network installation of the device very simple, and since no incoming ports need to be opened in the router, security is not compromised. See *Section General Settings Tab > ADVANCED NETWORK SETTINGS* for more information.

The ControlByWeb device can be configured to establish a connection when triggered by an event, such as an I/O state changing. This is done by setting a conditional task with the action being *Send Device State to Remote Service*.

With *Remote Services* enabled, a connection attempt will be made periodically according to the Connection Interval setting in the *Advanced Network* setup tab. The Connection String consists of static information about the device, a user-defined character string configured in the *Advanced Network* tab and ends with sending the state. xml.

The connection string is also sent at the same interval once the connection is open. The external server is responsible for closing the connection when it is done.

A three-character "ACK" response is expected in return to every connection string. If the "ACK" is not received within 10 seconds, the ControlByWeb device will close the connection.

When an event occurs and a connection is open, the state.xml file is sent.

6.6 Log Files

The module logs information to two different log files; log.txt and syslog.txt. Both log files are text files and are stored in nonvolatile memory; this data will not be lost due to power failure and the syslog.txt is not cleared when restoring factory defaults. The log files are stored in circular buffers which write from the beginning of the allocated memory space to the end and then repeat from the beginning (overwriting the original data). The nonvolatile memory is divided into 4K byte sectors. Each time data gets erased to make room for new data, a full sector (4K bytes) is erased at one time.

Data Log File – log.txt

This log file is user-configurable under the *Logging* tab, and stores real-world data such as relay states, digital/ analog input values, and events such as remote I/O state changes (see *Section* Logging Tab > GENERAL LOGGING SETTINGS for more information). It can be up to 3072K bytes long and is displayed using a comma-separated value formatting scheme.

Inputs, Relays, Counters, Vin, and Sensors will only be displayed if they are selected in the *Logging* setup tab. The file is read by requesting the log.txt file from the module. For example, using the default IP address the following command would be used to request the log file:

> http://192.168.1.2/log.txt

Example File Format:

> MM/DD/YYYY HH:MM:SS.mmm DDD, Digital I/O 1-2, Analog Inputs 1-4, Frequency Input, Vin, Sensors 1-16, Trigger: Source

Date and Time Format:

> MM – Month (1-12)
> DD – Day (1-31)
> YYYY – Year (1970 - 2106)
> HH – Hour in 24 hour time (0 -23)
> MM – Minutes (0-59)
> SS – Seconds (0-59)
> mmm – Milliseconds
> DDD – DST When in daylight savings, STD When not in daylight savings time

Sample File:

The file can then be saved using the 'Save As...' option under the 'File' menu of the web browser. If the TCP port has been changed (not port 80), the port will be required to read the file. For example, using the default IP address, and port 8000, the log file would be read as follows:

> http://192.168.1.2:8000/log.txt

The log.txt file may be erased with the following command:

> http://192.168.1.2/log.txt?erase=1

Note: If the User account is enabled in the Setup pages, the password will be required to access the log file.

System Log File – syslog.txt

The syslog file records various system events, which can be used for diagnostics and troubleshooting purposes.

File Format:

MM/DD/YYYY HH:MM:SS, (category to which message applies): (message)

Sample File:

01/02/2010 04:08:13 DEVICE: Power Up.
01/01/2010 00:41:05 DEVICE: Reset factory defaults.

Listed below is a description of messages that a user may see:

Category	Message	Description
DEVICE	Power Up	Device has been properly powered.
	Initialize	Device is ready.
	Reset Factory Defaults	Device has been reset to factory defaults.
EMAIL	Failed DNS Lookup	Unable to lookup mail server due to an incorrect DNS setting.
	Bad Username	Email was not sent due to an incorrect user name.
	Bad Password	Email was not sent due to an incorrect password.
	Authentication Required	A user name and password are required by the mail server.
	No Response	No response from SMTP server.
CLOCK	RTC Reset	Real Time Clock has been reset.
NTP SERVER	Request Attempt	Device attempting to connect to NTP Server.
	Failed DNS Lookup	Unable to lookup NTP server name due to an incorrect DNS setting.
	Success	Device successfully connected to NTP Server.
REMOTE SERVER	Failed DNS Lookup	Unable to lookup Remote Server due to an incorrect DNS setting.
	Connection Attempt Started	Connection port with Remote Server has been opened.
	Connection Closed	Connection port with the Remote Server has been closed.
DHCP	Address Acquired	IP address request successful, and IP address assigned.
	Lease Renewal	IP address assigned to device was renewed.
MAIN_MCU	New Firmware Loaded	New firmware has been loaded to the device.

This file is read by requesting the syslog.txt file. For example, using the default IP address, the following command would be used:

http://192.168.1.2/syslog.txt

Note: The setup username and password are required to access this file.

If the TCP port has been changed (not port 80), the port will be required to read the file. For example, using the default IP address, and port 8000, the log file would be read as follows:

http://192.168.1.2:8000/syslog.txt

To erase the file, use: http://192.168.1.2/syslog.txt?erase=1

6.7 Modbus/TCP (slave)

The module can be controlled and monitored using Modbus/TCP protocol. This provides a standard means of using the module with devices and software from other manufacturers. This section is not a tutorial on Modbus and it is assumed that the reader is already familiar with Modbus. Detailed Modbus information can be found at http://www.modbus.org.

Note: Modbus communications are disabled whenever the User account is enabled. This is because Modbus/TCP does not provide a mechanism for password protection. Make sure the User account is disabled (default) and Modbus functionality is enabled on the Advanced Network.

The module functions as a Modbus slave. Host devices, such as PLCs, open a connection with the module on port 502 (configurable under *Advanced Network* tab) and then send requests to read or set I/O states, or sensor values. When the module receives a command, it performs the desired function and returns a response.

The module can have additional I/O added and removed that can cause changes to the Modbus address map. For an up-to-date map of I/O on the module to the addresses for Modbus, please select the *View Modbus Address Table* button from the Setup pages under the *Advanced Network* tab.

The Modbus Address Table will list all the addresses to be used when accessing I/O on the module. It is broken down into three groups: register addresses, coil addresses, and discrete input addresses. Register addresses are used with the modbus register functions. Coil addresses are used with the modbus coil functions, and discrete input addresses are used with modbus discrete input functions.

The following is an example of a Modbus Address Table. Note that while the module might not have all the I/O hardware present in the table, it can have remote I/O for those types.

| IO Name | IO | Register Addr | | | | Coil Addr | Input Addr |
		Pulse Timer	Counter	On Timer	Total On Timer	IO	IO
Relay 1	0	512	--	--	--	0	--
Input 1	2	--	1026	1538	2050	--	1
Analog Input 1	4	--	--	--	--	--	--
Vin	6	--	--	--	--	--	--

The first column in the Modbus Address Table lists the name of the I/O. The seconds column lists the register address for the I/O. All I/O can be read and written as a register using this address and the modbus register functions regardless of the I/O type. Columns 3 through 6 are the register addresses used to access auxiliary functions of relays that have Pulse Timers, and digital inputs that have Counters, On Timers, and Total On Timers. The seventh column contains addresses used to read and write digital I/O and relays using the modbus coil functions. The last column contains addresses used to read digital I/O and inputs using the modbus discrete input functions.

The following sections provide an overview and explanation of the different Modbus functions.

6.7.1 Modbus Function Code Summary

The module supports the following function codes for reading and writing I/O. If the I/O type does not exist on the module, the function code is still supported for use with remote I/O of that type.

Code Name	Modbus Function	Usage
Read Coils	01	Read Relays and Digital I/O (Configured as outputs)
Read Discrete Inputs	02	Read Digital Inputs and Digital I/O (Configured as inputs)
Read Multiple Registers	03	Read Vin, Sensors, Registers, Counters, Analog Inputs (All I/O both Local and Remote)
Write Single Coil	05	Write Relays and Digital I/O (Configured as outputs)
Write Multiple Coils	15	Write Digital Inputs and Digital I/O (Configured as outputs)
Write Multiple Registers	16	Write Digital I/O Pulse Counters, Registers, Counters (All writable I/O)

Multiple commands may be sent without closing and re-opening the connection, but if no data is transferred for 50 seconds, the connection will time out and close. To keep the connection open, a read request can be sent periodically.

The module has two TCP sockets available for Modbus/TCP. This allows two connections to be open at one time. Requests for more than two open connections will be rejected.

When errors occur, an error code is returned. Most Modbus client software will interpret this code in a human readable form. The code is comprised of the original function code plus 0x80. For example, an error during the read coils function 0x01 would return 0x81. Each error has a qualifying exception number. The following are the possible exception codes and their meanings:

    0x01 - Function code not supported (also when Modbus is disabled in the setup pages).
    0x02 - Incorrect starting address/quantity of output combination.

6.7.2 Read Coils - Modbus Function Code 01 (0x01)

Read the state of relays and digital I/O configured as outputs.

Request

Start Address: Refer to Modbus map in setup pages
Coil Quantity: Refer to Modbus map in setup pages. Multiple Outputs may be read at the same time by specifying the correct starting address and quantity of coils to be read.

Response

The module will respond to the request with a data field of one or more bytes depending on the number of coils read. Each bit represents a coil status. A '1' indicates the Output is ON. A '0' indicates that the Output is *OFF*.

Bit zero of the return value will be the state of the coil corresponding to the start address. For example, if a start address of 0x0001 is used, bit zero will be the status of the first relay or digital I/O.

Coil State Byte							
Bit 7	**6**	**5**	**4**	**3**	**2**	**1**	**Bit 0**
X	X	X	X	X	X	Digital I/O 2	Digital I/O 1

Errors

The sum of the start address and coil count cannot exceed the maximum coil count or an error response will be returned.

The following are possible error responses:

Coil Read Error Function Code (1 byte):	0x81

Exception Codes (1 byte):	0x01 – Function code not supported. 0x02 – Incorrect combination of start address and quantity of Digital I/O

6.7.3 Read Discrete Inputs – Modbus Function Code 02 (0x02)

This function returns the state of digital inputs and digital I/O (when configured as inputs).

Request

Start Address: Refer to Modbus map in setup pages
Input Quantity: Refer to Modbus map in setup pages. Multiple Inputs may be read at the same time by specifying the correct starting address and quantity of inputs to be read.

Response

The inputs states are indicated by bits in the status byte(s). A 1 indicates that the input is switched *ON*. A 0 indicates that the input switched *OFF*. Bit zero of the return value will be the state of the discrete input corresponding to the start address. For example, if a start address of 0x0001 is used, bit zero will be the status of input 2.

Discrete Input State Byte							
Bit 7	6	5	4	3	2	1	Bit 0
X	X	x	x	x	x	Digital I/O 2	Digital I/O 1

Errors

Input Read Error Function Code (1 Byte):	0x82

Exception Codes (1 byte):	0x01 – Function not supported.
	0x02 – Incorrect combination of start address and quantity

6.7.4   Read Holding Registers – Modbus Function Code 03 (0x03)

The Read Holding Registers function is used mainly to read analog I/O such as the analog inputs, registers, counters, and vin. All I/O can be read using this function code. The holding register addresses can be found in the Modbus Address Table in the Advanced Network setup page under Modbus.

<u>Request</u>

32-bit sensor values are read from 16-bit register pairs. Consequently, senors addresses and registers must be even numbers.

Start Address: Refer to Modbus map in setup pages

Input Quantity: Refer to Modbus map in setup pages. Multiple Registers may be read at the same time by specifying the correct starting address and quantity of registers to be read. The number of registers read must be divisible by 2.

<u>Response</u>

32-bit floating-point values are returned, either as little-endian or big-endian numbers, depending on the configuration in the *Advanced Network* tab.

With little-endian ordering, a temperature reading of sensor 1 would return 0x800042A2. The least significant word would be 8000 hex and the most significant word would be 42A2. This hexadecimal value converts to a temperature reading of 81.25 degrees.

If a temperature or humidity sensor is not installed, a value of 0xFFFFFFFF (NaN) is returned. Other inputs will show measured values of the open circuits.

<u>Errors</u>

Sensor Read Error Function Code (1 byte):	0x83

Exception Codes (1 byte):	0x01 – Function code not supported.
	0x02 – Incorrect combination of start address and input quantity

6.7.5 Write Single Coil – Modbus Function Code 05 (0x05)

Control digital outputs and relays one at a time.

Request

Start Address: Refer to Modbus map in setup pages
Output Value: 0x00 (Off), 0xFF(On)

Response

The response mirrors the requested state, 0x00 or 0xFF.

Errors

Single Coil Write Error Function Code (1 Byte):	0x85

Exception Codes (1 byte):	0x01 – Function not supported. 0x02 – Address out of range. 0x03 – Padding value.

6.7.6 Write Multiple Coils - Modbus Function Code 15 (0x0F)

One or more bytes can be written to set the state of multiple digital outputs and relays, each bit representing one digital output or relay.

Request

Digital output and relay states are controlled by specifying the start address of the first digital output to be controlled, the number of the digital outputs to be affected, and the digital output state byte(s).

A value of 0xFFFF would be used to turn *ON* up to 16 digital outputs and relays. A value of 0x0000 would be used to turn them *OFF*. A value of 0xF0 would turn off the first 4 digital outputs while turning on outputs 5 through 8.

Start Address (2 bytes):   Refer to Modbus map in setup pages
Output Quantity (1 bytes):   Refer to Modbus map in setup pages
Byte Count (1 byte):   Refer to Modbus map in setup pages (Output Quantity divided by 8)
Digital I/O Value (Byte Count bytes):   0x0000 – 0xFFFF

Digital I/O State Byte							
Bit 7	6	5	4	3	2	1	Bit 0
X	X	X	X	X	X	Digital I/O 2	Digital I/O 1

Response

The quantity value is returned.

Errors

Multiple Coil Write Error Function Code (1 Byte):	0x8F

Exception Codes (1 byte):	0x01 – Function not supported 0x02 – Incorrect combination of start address and Digital I/O quantity 0x03 – Byte count out of range

6.7.7 Write Multiple Registers – Modbus Function Code 16 (0x10)

The Modbus Write Multiple Registers function can be used to set the state of writable analog I/O such as registers and analog outputs.

Request

Start Address: Refer to Modbus map in setup pages
Input Quantity: Refer to Modbus map in setup pages. Multiple Registers may be written at the same time by specifying the correct starting address and quantity of registers to be read. The number of registers written must be divisible by 2. The values sent are in IEEE 754 floating point format. Also, the endianness configured in the modbus setup pages is taken into account.

In little-endian the value 81.25 would be as follows: 0x800042A2. The least significant word would be 8000 hex and the most significant word would be 42A2. In big-endian, the least significant word would be 0x42A2 and the most significant word would be 0x8000.

Response

The request is acknowledged by responding with the register quantity that was written.

Errors

Pulse Function code Error (1 Byte):	0x90

Exception Codes (1 byte):	0x01 – Feature not supported. 0x02 – Address quantity not an even number. Incorrect combination of start address and register count.

    Xytronix Research & Design, Inc.

6.8 BASIC Scripts

BASIC (Beginners All-purpose Symbolic Instruction Code) is a computer programming language that has been in use for many years. The module has an integrated BASIC interpreter for a simple BASIC script. This provides a great deal of flexibility by allowing users to customize basic functions of the unit.

The interpreter only supports a small subset of the BASIC commands that are available for larger computers. Some non-standard commands have been added, and some commands may function differently on the module's BASIC interpreter than on other platforms. The following is a short tutorial on the supported BASIC functions.

Example scripts are available at https://www.controlbyweb.com/support/tutorials/basic-scripttutorial.html. Contact customer support if further assistance is required.

6.8.1 Structure

A BASIC script is written as a .txt file, which is then uploaded to the device using the *Script* setup tab. The maximum script size is 4-Kbytes. Each line within the script contains a single statement. Line numbers are not used.

Statements are not case sensitive; however, variables are.

IF THEN, FOR loops, and DO loops can only be nested up to 5 times per command. For those not familiar with nesting, the following is an example of nested FOR loops:

```
FOR a = 0 to 100
    FOR b = 0 to 100
    NEXT b
NEXT a
```

Every program must end with an END statement. Subroutines would then follow after the END statement, if required. The last line of the script should be left blank.

6.8.2 Line Format

Every line follows the same format. The basic format is:

statement (variable) (=, <, >, <=, >=, <>) (expression) (THEN)

The fields in parentheses are optional depending on the statement. One space must be used between all statements, numbers, variables, operators, expressions, etc. Multiple spaces are invalid.

Comments may be inserted, but must begin with an apostrophe. All text on a line after the apostrophe is ignored. For example:

```
LET a=1 'this will return an error because of insufficient spacing
LET a = 1 'this is valid
LET a =  1 'this will return an error because of too many spaces
```

Only a single variable or literal is allowed on the left side of any operator. The following example is incorrect and will return an error. The error occurs because there is more than a single value to the left of the comparison operator (a + b is to the left of =).

```
IF a + b = 3 THEN 'this will return an error
```

To fix the above line, replace 'a + b'. One of the following options may be used:

```
Let c = a + b
IF c = 3 THEN
```

6.8.3 Supported Statements

The following are the statements supported by the ControlByWeb™ BASIC interpreter.

LET

The LET statement assigns a variable a value. The format is:
LET (variable) = (expression)

IF THEN, ELSE, END IF

The IF THEN statement tests the truth of a condition. The ELSE statement defines a second function if the condition is found false. In other words, if the condition is true, then a function is performed. If it is not true, a second function may be performed. The second function may or may not be necessary depending on the application. The IF THEN (ELSE) statement must always be followed with an END IF statement. The format is:

IF (variable) (=, <, >, <=, >=, <>) (expression) THEN
(Function 1)
ELSE
(Function 2)
END IF

A special Variable NAN is available to check if I/O values are valid before using them. To check if a 1-Wire temperature sensor value is valid before using it to send an email the following could be used:

IF NAN <> io.oneWireSensor1 Then
    ' send email
ELSE
    ' do something else
END IF

Note: In most BASIC interpreters, 'Function 1' (see above) may be placed after the THEN statement. This interpreter requires 'Function 1' to be put on the following line.

FOR TO, NEXT

The FOR TO statement loops a section of code a predefined number of times. The NEXT statement always follows the section of code to be looped. The format is:

FOR (variable) = (expression) TO (expression)
(code to be looped)
NEXT (variable)

DO WHILE, LOOP

The DO WHILE statement loops a section of code while a condition is found true. The LOOP statement always follows the section of code to be looped. Note that if the condition is omitted, the code will be looped without end. The format is:

DO WHILE (variable) (=, <, >, <=, >=, <>) (expression)
(code to be looped)
LOOP

If the loop is to continue indefinitely, the format would be:

DO
(code to be looped)
LOOP

Example:

```
let t0 = 100
do while t0 > 0
        ' wait
loop
```

LOG

The LOG statement causes the device to log data according to the settings specified under the *Logging setup tab.*

Note: In order to log, logging must be enabled in the Logging setup tab.

The format is: LOG

EMAIL

The EMAIL statement causes the device to send an email of the same format as that generated by other status change and alarm conditions. If an I/O variable is given as a parameter, then the subject line in the email will contain "Variable Name = Variable Value". If no I/O variable is given, then the subject of the email will be "Basic Script".

EMAIL (io.digitalIO1, etc)
Examples:
```
        EMAIL
    ...or...
        EMAIL io.analogInput1
    ...or...
        EMAIL io.input1
    ...or...
        EMAIL io.temp2
```

END

The END statement ends the main body of code.

CALL

The CALL statement is found within the main body of code, but requires the interpreter to skip to a subroutine found at the end of the program. After the subroutine is finished, the interpreter returns to the line immediately following the CALL statement. The format is:

CALL (name of subroutine)

SUB, END SUB

The SUB statement defines the beginning and name of a subroutine. The END SUB statement defines the end of the respective subroutine. Subroutine names can be up to 20 characters long and are case sensitive. The SUB and END SUB statements always must follow the END statement. The format is:

END

*'*** Subroutines Go Here ****
SUB (name of subroutine)
   (contents of subroutine)
END SUB
SUB (name of subroutine)
   (contents of subroutine)
END SUB*

REM or '

The REM or apostrophe (') statement designates remarks made by the programmer. The interpreter will disregard any characters on the line following these statements.

6.8.4 User-Defined Variables

Two types of variables are available for use in the ControlByWeb™ BASIC interpreter, user-defined variables, and predefined variables.

Up to 10 user variables may be initialized. These must be single character, lower case letters.

> *a, b, c, d, e, f, g, h, i, j*

They are always global and stored internally as floating point numbers. Variables are defined using the LET statement.

Examples:

        Let b = 5 'variable b will be set to 5
        Let d = b + 2 'variable d will be set to 7

6.8.5 Predefined Variables

The following are useful predefined variables for the ControlByWeb™ BASIC interpreter. These are useful for accessing internal values and features states.

Timer Variables

Ten timers are available for use in BASIC scripts.

> t0, t1, t2, t3, t4, t5, t6, t7, t8, t9

Timers can be set to any positive integer (or 0) by using the LET statement. As soon as a value is assigned to a timer, it will begin to count down immediately by decrementing one count every 100 ms until it reaches zero.

Examples:
        Let t3 = 1500   'set timer 3 to 150 seconds
        Let t1 = 0   'disable time 1

These timers are different than the Timer I/O that can be configured under the I/O setup tab. They are only accessible to the basic script, and their resolution is 100 ms. The Timer I/O can also be accessed from the basic script as io.timer1, etc. These timers have 1 second resolution.

Date and Time Variables

The variables 'time' and 'date' are predefined, read only variables that store the current date and time. They can be used to schedule events and activities, such as sending emails, reading temperature, or setting Outputs. The date uses the mm/dd/yyyy format. Clock time is formatted as hh:mm:ss (24-hour clock).

Example:

        If date = 01/01/2014 Then
          If time > 12:30:00 Then
            Let io.rmtRelay1 = 1
            Let io.rmtRelay2 = 0
          End If
        End If

Note: Current date and time can only be set in the Date/Time tab.

Event Variables

Data and time variables can be used to execute script events.

Up to ten date variables are available.

> *ed0*   'event date variable 0
> *…*   'ed1, ed2, ed3, ed4, ed5, ed6, ed7, ed8,
> *ed9*   'event date variable 9

The value assigned to event variables should be in the format mm/dd/yyyy. The event date variables store the number of days that have passed since January 1, 1970.

Event time variables may be used in math expressions as well as comparison statements.

Time variables have a similar naming convention.

> *et0*   'event time variable 0
> *...*   'et1, et2, et3, et4, et5, et6, et7, et8
> *et9*   'event time variable 9

Event time variables are declared in the format hh:mm:ss in 24-hour time. The event time variables store the number of seconds from the beginning of the day.

Event date and event time variables of the same number are linked. If the event time variable is incremented more than the number of seconds in a day (86400 seconds), the variable is reset to 0 and the event date variable is incremented by one. For example, if et3 rolls over to zero, ed3 will be incremented.

The following script example demonstrates defining the event variables and comparing it to the current date and time. Assume current time is April 10, 2010 at 1:30 AM and the event should occur in one hour and every hour thereafter.

Example:

```
Let a = 1
Let ed1 = 04/10/2010                'sets the event date to April 10,
                                    '2010.
Let et1 = 02:30:00                  'sets the event time to 2:30
Do While a <> 0
   If ed1 >= date Then              'Tests event date versus current date.
      If et1 >= time Then           'Tests event time versus current time.
         Let et1 = et1 + 3600       'Increments the event time by one hour
                                    '(in seconds).
                                    'Event to occur would go on this line

      End If
   End If
Loop
End
```

6.8.6 I/O Variables

The BASIC interpreter has full access to all configured I/O on the module. The I/O are treated as variables and can be read and written to the same as other variables. Depending on the ControlByWeb device, the number fixed I/O resources will vary. After I/O is added and defined they can be controlled and monitored within BASIC scripts.

BASIC scripts must reference I/O resources such as digital inputs and relays in the form of io.name where "name" is the resource name defined when the resource was added in the I/O Setup tab. If the resource name has embedded spaces, they must be removed in the io.name statement. For example, if a remote relay is named "Warehouse Fan", the fan can be turned on with the BASIC statement "LET io.warehouseFan = 1". If a resource name is changed during development and testing, the resource name in the BASIC script must be renamed to match.

The button on the BASIC setup page displays a list of all of the I/O resources currently available together with the equivalent token to be used in BASIC scripts to reference the respective I/O resource.

Relay Variables

Relay variables are represent the state of the relays and can be used to change the state of the relays. Remote relays and local relays are read and controlled the same. Below is an example on how to access the relays provided they are at the default names.

io.relay1 'relay 1
io.device1Relay1 'remote relay 1

Note: Commands can be sent to remote relays, but the states of those relays cannot be reliably read immediately. Consequently, remote relays should not be used in safety or security-critical applications.

Relay outputs can be turned ON, turned OFF, pulsed, toggled, or read in BASIC scripts. The LET statement is used to set the output state. The state options available are:

   0 – turn relay off
   1 – turn relay on
   2 – pulse relay
   5 – toggle relay

The pulse time is specified in the **Relay** setup tab.

Examples:

        Let io.relay1 = 1 'turn on relay 1
        Let io.relay1 = 0 'turn off relay 1
        Let io.relay1 = 2 'pulse relay 1
        Let io.relay1 = 5 'toggle relay 1
        Let a = io.relay2 'read the state of relay 2, state will be 0 or 1

Digital Input Variables

The digital input states can be read in BASIC scripts. Below is an example assuming the name for input 1 is left at the default.

   io.digitalInput1   'digital input 1

Example:
        Let a = io.digitalInput1 'sets 'a' equal to value of input 1

Counter Variables

The input counter can be read in BASIC scripts. Below is an example assuming the name for input 1 is left at the default.

   io.digitalInput1.count   'input 1 counter

Example:
        If io.digitalInput1.count > 5000 Then 'If input counter 1 is greater than
        5000 then
                Let io.relay1 = 1 'turn output 2 on
        End If

Analog Input Variables

The analog input scaled values can be read in BASIC scripts. Below is an example assuming the name for analog input 1 is left at the default.

> *io.analogInput1*   *'analog input 1*

Example:

```
If io.analogInput1 > 4 Then        'If Analog Input 1 is greater than 4 then
     Let io.rmtRelay1 = 2               'Pulse remote relay 1, otherwise
Else
     Let io.rmtRelay1 = 0               'remote relay 1 will be off
End If
```

Analog Output Variables

The analog outputs can be set in BASIC scripts. The LET statement is used to set the analog output. Note: the output value is subsequently processed by the Y=mX+b equation to generate the actual output of the digital to analog converter. In the example below the *User Input Max* and *User Input Min* fields of the Analog Outputs tab have been set for a 1 to 1 output relationship (no scaling). This allows the BASIC calculations and output settings to be in engineering units.

Example:

```
Let io.analogOutput1 = 0           'set analog output 1 to 0.0V
Let io.analogOutput2 = 2.5         'set analog output 2 to +2.5V
Let io.analogOutput3 = 10          'set analog output 3 to +10V
Let io.analogOutput4 = -2.5        'set analog output 4 to -2.5V
Let io.analogOutput5 = 20          'set analog output 5 to 20.0mA
```

Temperature/Humidity Sensor Variables

Each temperature sensor or humidity sensor can be read in BASIC scripts. Below is an example assuming the name for the 1-Wire sensor is left at the default.

> *io.oneWire1*   *'temp or humidity sensor 1*

Example:
```
If io.oneWire1 >= 80 Then          'If temperature sensor reads greater than 80
                                   ' degrees, then
    Let io.relay2 = 1              'Turn on relay 2
End If
```

Registers

Registers are similar to general purpose variables and can be used in the same manner. The difference between general purpose variables and registers is that registers can be accessed by the user through a web browser and modified while the script is running. Scripts can use registers as a method of retrieving input from the user or through any of the scheduled or conditional tasks. Below is an example of how to access a register assuming it is at the default name.

> *io.register1*

The following example demonstrates the BASIC script reacting to user input by creating a script that will turn **ON** a remote relay if Register 1 equals 1 and turn **OFF** if a remote relay if *Register* 1 equals 0.

Example:

```
Do
    If io.register1 = 1 Then
        Let io.device1Relay1 = 1
    Else
        Let io.device1Relay1 = 0
    End If
Loop
End
```

Registers can also be used to show numeric values, such as a count-down timer. The following code will show a count down from 10 seconds and automatically restart. The refresh rate must be configured in the *Control Page Setup* in order to see each of the count-down values.

Example:

```
let t0 = 10                          'Initialize variables and start timer
let io.register1 = 10

'Main Loop
Do
        if t0 = 0                    'when timer reaches 0 decrement counter
            let io.register1 = io.register1 - 1
            let t0 = 10
        end if

        if io.register1 < 0 'restart when counter reaches zero
            let io.register1 = 10
        end if
loop
end
```

Appendix A: Restoring Factory Default Settings

In the event that the IP address or passwords are forgotten, WebRelay-10 Plus™ may be restored to its original factory default settings.

1. Remove the DC power from the unit. This will also de-energize any relay coils.

2. Use a thin, non-conductive object (such as a toothpick) to press and hold the small button located next to the RJ-45 connector and Ethernet port. When the object is inserted, a tactile feedback can be felt as the button is pressed.

   CAUTION: DO NOT USE METAL OBJECTS FOR THIS FUNCTION.

3. While depressing the button, the green and yellow network LEDs will flash, apply power and wait for about 10 seconds before releasing the button. All settings will be back to the original factory defaults. log.txt and syslog.txt are retained.

4. Refer to **Section 2.3 Establishing Communications for Setup** to begin reconfiguration of the device.

Reset
Button

Figure Appendix A.1 - Restore Factory Defaults

Appendix B: Installing New Firmware

From time to time, updates are made to the module's firmware. As an industrial device and unlike many consumer products, firmware updates are recommended only on an as-needed basis. The firmware can be updated in the field. The procedure for updating the firmware is outlined below. Please note that it is important that this procedure is followed precisely.

Requirements

The firmware update software requires Windows 7/8/10.

Setup

1. Updating firmware will not automatically save settings, scripts, etc. Before proceeding with the firmware update, please back up all settings files and Basic scripts.

2. Contact technical support if a firmware update is needed and a download link will be provided. Only a module's image can be installed on the module, so make sure the correct image is being downloaded.

3. bootloader.exe will connect to the module using default IP address 192.168.1.2, not the address currently assigned to the module. After the update, all settings will be lost and the device will return to its default IP address of 192.168.1.2. Configure the PC to the same subnet as the IP address 192.168.1.2, such as 192.168.1.10. For instructions on doing this see *Section Establishing Communications for Setup*.

   Note: The IP address of the module will automatically be set to the default 192.168.1.2 during the update process. Since the module supports Auto Negotiation, a crossover cable is not necessary.

4. Open the bootloader.exe utility on the computer by double clicking on the downloaded file.

5. Within the ControlByWeb™ Programmer utility programmer, select File, then Open. Specify the firmware image downloaded from the ControlByWeb™ web site.

Device Upgrade Procedure

Carefully follow the following steps to put the module into bootloader mode and perform the upgrade:

1. Remove DC power from the module.

2. Using a small, non-conductive tool, press and hold the reset button.

3. While holding the reset button, apply power to the module. The LINK and ACT lights will flash. Continue to hold the reset button for the next step.

4. While holding the reset button, press the Upload Firmware button at the bottom of the ControlByWeb™ Programmer window. After the programming process begins, the reset button can be released.

5. Programming will take approximately 60 seconds, the LINK LED will stop flashing and remain lit. The module will be set to factory defaults with an IP address of 192.168.1.2.

6. Refer to *Section Establishing Communications for Setup* to reconfigure the device. Verify the new version of firmware has been installed by viewing the default setup page with a web browser (http://192.168.1.2/setup.html).

Appendix C: Accessing WebRelay-10 Plus™ Over the Internet

WebRelay-10 Plus™ can be monitored and/or controlled from a remote location over the Internet. Once WebRelay-10 Plus™ can be accessed on the local network, almost all of the settings required to provide remote access are in the router and not in WebRelay-10 Plus™. This guide is not meant to be a tutorial in router setup, but rather to provide a basic overview of remote access. For specific details, the user should refer to the instruction manual for the router on the local network. Users not familiar with basic IP networking should study one or more basic IP networking tutorials before proceeding (many tutorials are available on the Internet).

IP Addresses

Every device on the Internet is identified by a unique address called an IP (Internet Protocol) address. IP addresses are somewhat similar to mailing addresses in that they identify the precise logical location of the device on the Internet. The IP address identifies the global region down to the network and then the specific device on that network. IP addresses are globally maintained and assigned by an entity called the Internet Assigned Numbers Authority (IANA). IP addresses consist of four sets of numbers that range from 0 to 255 and are separated by a decimal. For example, 192.168.200.167 is an IP address.

Every device that is "directly" connected to the Internet uses a "public" IP address. WebRelay-10 Plus™ can be assigned a public IP address for direct connection to the Internet. Typically, a public IP address to would only be assigned to WebRelay-10 Plus™ when it is the only device on the local network. The IP address would be obtained from an Internet Service Provider (ISP).

Due to the limited number of public IP addresses, private networks can be set up with "private" IP addresses. These addresses are used within a local network and have no global designation, they are not routed on the Internet. The following address blocks are designated for private networks (where x represents decimal numbers from 0 to 255): 192.168.x.x, 10.x.x.x, and 172.16.x.x.

A Simple Local Area Network

A small Local Area Network (LAN), can be made up of two or more computers or other devices connected to an Ethernet switch. Each device on the network is assigned a unique private IP address. For example, consider a simple network that consists of a computer, an WebRelay-10 Plus™, and a WebRelay. In this example, the computer is assigned an IP address of 192.168.1.10, WebRelay-10 Plus™ has the IP address of 192.168.1.25 and a WebRelay has and IP address of 192.168.1.26. A person using the computer can access WebRelay-10 Plus™ by entering its IP address in the URL line in the browser,http://192.168.1.25. Similarly, WebRelay™ can be accessed by entering its unique private IP address in the URL line in the browser, http://192.168.1.26.

WebRelay-10 Plus™

WebRelay™

Ethernet Switch
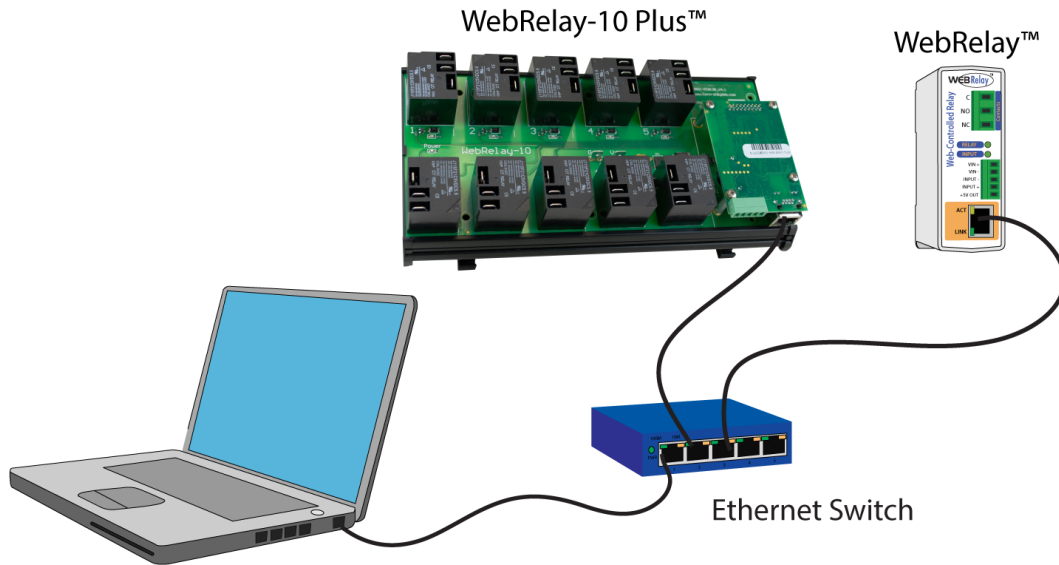
Computer

Figure Appendix C.1 - Local Area Network

A Simple LAN connected to the Internet

The LAN in the example above can be connected to the Internet by adding a router and an Internet connection. The router has two network connections. It has an Ethernet network connection that connects to the LAN, and it has another connection that connects to the Internet (this is called the WAN or Wide Area Network connection). Each network connection on the router has an IP address. In our example, the IP address on the LAN side of the router has an address of 192.168.1.1. The IP address on the WAN side of the router has an IP address that has been assigned by the Internet Service Provider such as 266.70.164.97 (note that this is not a valid IP address because each number cannot be larger than 255, but it is used in this example for illustration purposes only).

WebRelay-10 Plus™

WebRelay™

Internet

Ethernet Switch
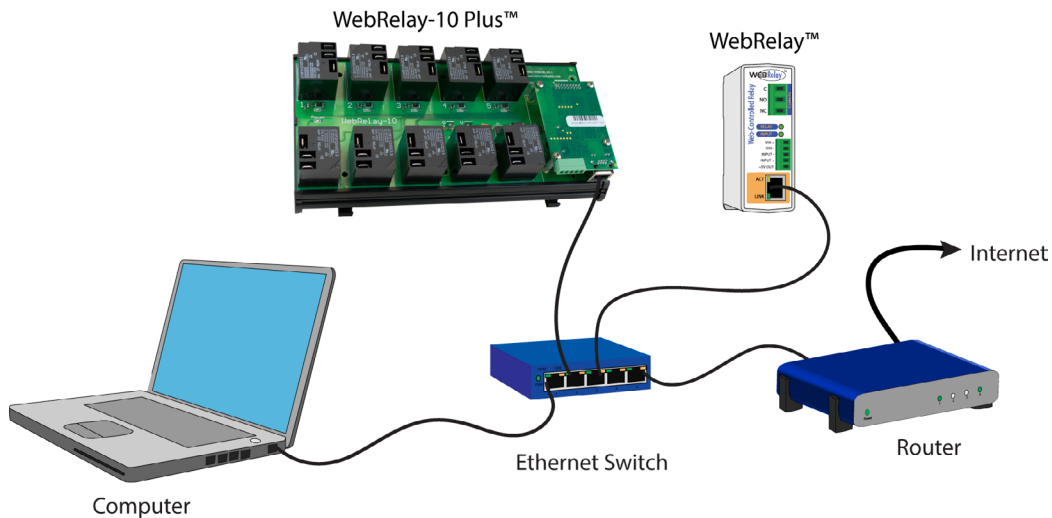
Router

Computer

Figure Appendix C.2 - LAN Connected to the Internet

In the example, when a user on the computer needs to access a server on the Internet, the computer sends the request to the router at 192.168.1.1. The router sends the request to the server on the Internet. The server does not send the response directly to the computer on the LAN, but to the router at the IP address of 266.70.164.97. The router then forwards the response to the computer. This way, all devices on the LAN share a single public IP address. This is called Network Address Translation.

Port Forwarding

The router can be configured to allow outside access to the ControlByWeb devices. All requests from the Internet to any device on the local network must use the public IP address (203.0.113.254). With only a single IP address, TCP ports are used to identify the intended device for the incoming message.

Using the mailing address analogy, the port is similar to a post office box. The IP address specifies the location, and the port specifies the specific recipient. Port numbers can be set to any number between 1 and 65535. However, many port numbers are reserved for specific applications and should be avoided. As a general rule, numbers above 8000 are safe to use. All of the ControlByWeb™ devices come from the factory with the HTTP port set to 80, which is the standard port for HTTP. In the example above, the WebRelay-10 Plus™ HTTP port will be changed to port 8000 and WebRelay™ port will be changed to 8001. Once the ports are changed in the two ControlByWeb™ devices, the router must be set up for port forwarding.

Port forwarding associates the IP address of each local device with an assigned port. In the above example, the address 192.168.1.25 for the WebRelay-10 Plus™ would be associated with port 8000. The address 192.168.1.26 for WebRelay™ would be associated with port 8001. The WebRelay-10 Plus™ would be accessed from the Internet by entering the public IP address of the router, plus the port number assigned to the WebRelay-10 Plus™ in the URL window of the browser, http://203.0.113.254:8000. All Internet requests to the router for port 8000 would be forwarded to the WebRelay-10 Plus™. Similarly, all request for port 8001 would be forwarded to WebRelay.

Note: When an HTTP request comes in to the router without the specific port specified (http://203.0.113.254), the router will handle this as a port 80 request (default HTTP port). In other words, http:// 203.0.113.254 is exactly the same as http://203.0.113.254:80.

Router configuration can vary widely. Some routers have the capability of translating the addresses and the ports, which would require no port configuration change on the ControlByWeb device. For example, the router would be configured so that messages sent to http://203.0.113.254:8000 would be forwarded to http://203.0.113.254:80, which is the default HTTP port.

An example screenshot of a router configuration is given below. This setup allows the two ControlByWeb™ devices in the above example to be accessed remotely from the Internet.

Figure Appendix C.3 - Port Range Forwarding   *Note: Routers will vary, this is an example.*

Accessing Setup Pages

After changing ports, the setup pages are accessed on a local network as described below:

```
http://(ControlByWeb Device's Local IP Address):(Port Number)/setup.html
```

For example, to access the setup pages when the port is set to 8000, the following command would be used:

```
http://192.168.1.25:8000/setup.html
```

To access the ControlByWeb™ devices from the Internet, enter the public IP address of the router plus the port number of the desired device in the following format:

```
http://(Public IP Address of Router):(Port Number of Device)/setup.html
```

Using the example above, the following line would be used to access the setup page of the device:

```
http://203.0.113.254:8000/setup.html
```

Appendix D: Log Files

WebRelay-10 Plus™ logs information to two different log files; log.txt and syslog.txt. Both log files are text files and are stored in non-volatile memory; this data will not be lost due to power failure. The log files are stored in circular buffers which write from the beginning of the allocated memory space to the end, and then repeat from the beginning (over-writing the original data). The non-volatile memory is divided into 4K byte sectors. Each time data gets erased to make room for new data, a full sector (4K bytes) is erased at one time.

Data log file: log.txt

This log file is user-configurable under the **Logging** setup tab, and stores real-world data such as temperatures and events such as relay state changes (see section **2.4.6 Logging Tab** for more information). It can be up to 3072K bytes long and is stored using a comma separated value formatting scheme.

Digital Inputs, Relays, Analog Inputs, and Sensors will only be displayed if they are selected in the **Logging** setup tab. The file is read by requesting the log.txt file from WebRelay-10 Plus™. For example, using the default IP address the following command would be used to request the log file:

```
http://192.168.1.2/log.txt
```

File format:
```
MM/DD/YYYY HH:MM:SS, Input 1, Input 2, Relay 1, Relay 2, ..., Relay 10,
Sensor 1, Sensor 2 ... Sensor 8, Source: State
```

Date and time of event:
> MM - Month (1-12)
> DD – Day (1-31)
> YYYY – Year (1970 - 2106)
> HH – Hour in 24 hour time (0 -23)
> MM – Minutes (0-59)
> SS – Seconds (0-59)

Example:
```
Date Time,I1,I2,R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,S1,S2,S3,Source
11/15/2010 11:00:46,0,1,1,1,1,0,1,0,1,0,1,1,078.0,076.4,082.7,Input02:(on)
11/15/2010 11:00:48,0,0,1,1,1,0,1,0,1,0,1,1,078.0,076.4,082.7,Input02:(off)
11/15/2010 11:01:08,0,0,0,0,0,0,1,0,1,0,1,1,078.0,076.5,082.9,Relay03:(off)
11/15/2010 11:02:04,0,0,0,0,0,1,1,0,1,0,1,1,077.7,076.4,083.4,Relay04:(on)
```

The file can then be saved using the 'Save As...' option under the 'File' menu of the web browser. If the TCP port has been changed (not port 80), the port will be required to read the file. For example, using the default IP address, and port 8000, the log file would be read as follows:

```
http://192.168.1.2:8000/log.txt
```

To erase the file, use the following parameter in the address bar:

```
http://192.168.1.2/log.txt?erase=1
```

After erasing the file, it might be necessary to refresh the page.

Note: If the control password is enabled in the setup pages, the password (control password) will be required to access this file.

System log file: syslog.txt

The syslog.txt file records various system events, which can be used for diagnostics and troubleshooting purposes.

File Format:

```
MM/DD/YYYY HH:MM:SS, (category to which message applies): (message)
```

Sample File:

```
01/02/2010 04:08:13 DEVICE: Power Up.
01/01/2010 00:41:05 DEVICE: Reset factory defaults.
```

Listed below is a description of messages that a user may see:

Category:	Message:	Description:
DEVICE	Power Up	Device has been properly powered.
	Initialize	Device is ready.
	Reset Factory Defaults	Device has been reset to factory defaults.
EMAIL	Failed DNS Lookup	Unable to lookup mail server due to an incorrect DNS setting.
	Bad Username	Email was not sent due to an incorrect user name.
	Bad Password	Email was not sent due to an incorrect password.
	Authentication Required	A user name and password are required by the mail server.
	No Response	No response from SMTP server.
CLOCK	RTC Reset	Real Time Clock has been reset.
NTP SERVER	Request Attempt	Device attempting to connect to NTP Server.
	Failed DNS Lookup	Unable to lookup NTP server name due to an incorrect DNS setting.
	Success	Device successfully connected to NTP Server.
REMOTE SERVER	Failed DNS Lookup	Unable to lookup Remote Server due to an incorrect DNS setting.
	Connection Attempt Started	Connection port with Remote Server has ben opened.
	Connection Closed	Connection port with the Remote Server has been closed.
DHCP	Address Acquired	IP address request successful, and IP address assigned.
	Lease Renewal	IP address assigned to device was renewed.
MAIN_MCU	New Firmware Loaded	New firmware has been loaded to the device.

This file is read by requesting the syslog.txt file. For example, using the default IP address the following command would be used:

```
http://192.168.1.2/syslog.txt
```

Note: The setup user name and password are required to access this file.

If the TCP port has been changed (not port 80), the port will be required to read the file. For example, using the default IP address, and port 8000, the log file would be read as follows:

```
http://192.168.1.2:8000/syslog.txt
```

To erase the file, use:

```
http://192.168.1.2/syslog.txt?erase=1
```

Appendix E: External Server and Remote Services

Note: The following methods are supported by WebRelay-10 Plus™; however, Xytronix Research & Design, Inc. does not provide or support custom third party applications, or external web servers.

Accessing WebRelay-10 Plus™ with Custom Software or Third Party Applications

WebRelay-10 Plus™ provides a programming interface that allows applications running on a computer to have full access to it. Custom applications can send commands to WebRelay-10 Plus™ for monitoring and control. They can also be used to provide a custom user interface, provide access to multiple units in a single screen, and allow for automation, logging, etc.

Using an External Web Server

Rather than accessing WebRelay-10 Plus™ directly from a computer, an external web server can be used. The term "external" web server is used here to mean a separate web server (such as Apache or IIS) that is not the web server built into WebRelay-10 Plus™. In this scenario, users access custom web pages that reside on the external web server and the external web server communicates with WebRelay-10 Plus™. This means users would not communicate directly with WebRelay-10 Plus™.

Using an external web server to provide a custom user interface is beneficial because multiple WebRelay-10 Plus™ devices (or other ControlByWeb™ products) can be integrated into a single control page or set of pages so they appear as part of a larger system rather than individual devices. In other words, the user may not be aware that he/she is using multiple ControlByWeb™ devices, but rather the user sees a custom control page that shows the entire system. In addition, the use of an external web server allows programmers to create custom user interfaces that take advantage of the additional resources typically available on larger web servers, including more memory and various web programming languages.

There are two approaches that an external server can use to communicate with WebRelay-10 Plus™ and other ControlByWeb™ devices, Direct Server Control and Remote Services.

Direct Server Control

The first approach is for the external server to create a TCP connection whenever it needs to access WebRelay-10 Plus™. In this case, the external server opens the connection, sends commands and/or reads the device, and closes the connection.

This method is ideal when the web server and all of WebRelay-10 Plus™ devices are on the same network (without routers between them). In this case, the server can communicate with WebRelay-10 Plus™ directly and securely since data never has to leave the local network. When the server and WebRelay-10 Plus™ are on different networks, routers must be configured to allow appropriate access. If a public network is used, such as the Internet, security precautions should be considered.

Remote Services

The second approach is for WebRelay-10 Plus™ to initiate a connection using the "Remote Services." The settings under the Advanced Network tab in the setup pages will enable the WebRelay-10 Plus™ to open a TCP connection with an external server. Once the connection is open, the external server can send commands and/or read the device. The external server can leave the connection open (so that it never closes) or close the connection.

"Remote Services" is ideal for installations where the server and WebRelay-10 Plus™ are installed on different networks. This is especially useful when each WebRelay-10 Plus™ is installed on a separate private network. For example, if the user doesn't control the network connections where WebRelay-10 Plus™ is installed, "Remote Services" would initiate a TCP connection over the Internet with the control computer. Since WebRelay-10 Plus™ initiates the connection, the control computer doesn't have to know the IP address of WebRelay-10 Plus™. This means that WebRelay-10 Plus™ can be installed using DHCP. In addition, no special router configuration is required. This makes the network installation of WebRelay-10 Plus™ very simple, and since no incoming ports need to be opened in the router, security is not compromised. See the **Advanced Network** Tab for more information.

WebRelay-10 Plus™ can be configured to establish a connection when triggered by an event, such as an Outlet state changing. This is done by setting the Remote Service/SNMP checkbox in the setup tabs.

When an event occurs and a connection is open, the state.xml file is sent.

If a connection is not open and Remote Services is enabled, a connection will be established by sending the connection string. Once the connection is established state.xml will be sent.

Connection String

With Remote Services enabled, a connection attempt will be made periodically according to the Connection Interval setting in the **Advanced Network** setup tab. The Connection String is a user-defined character string configured in the **Advanced Network** tab.

The connection string is also sent at the same interval once the connection is open. The external server is responsible for closing the connection when it is done.

Appendix F: Specifications

Power Requirements:

- Voltage: 9-28VDC
- Max Current: 477mA - 1391mA

Relay Contacts:

- Number of Relays: 10
- N.O.
  - 40A @ 240VAC resistive
  - 30A @ 277VAC General Purpose
  - 2hp @ 250VAC
- N.C.
  - 30A @ 240VAC, 30VDC resistive
  - 20A @ 277VAC General Purpose
  - 1-1/2 hp @ 250VAC
- Contact Type: SPDT (Form 1C)
- Load Type: General Purpose
- Contact Resistance: < 30 milliohms initial
- Contact Material: AgSnO2
- Electrical Life: 100K cycles (Typical)
- Mechanical Life: 10M cycles (Typical)
- Environmental Rating: Over voltage Category II,
- Pollution Degree 2
- Relay Modes: ON/OFF or Pulse
- Pulse Timer Duration: 0.1 to 86,400 Seconds (1-day)
- Relay Connections: 1/4" Tab Terminals

Digital Inputs:

- Number of Inputs: 2
- Type: Non-Isolated
- Voltage Range: 0-5VDC
- Current: 47K Pullup
- Minimum Hold Time: 20ms
- Input Isolation: Non-Isolated
- Input Functions: Counters, Email Alerts, SNMP Traps
- Maximum Count: 32-bit
- Max Count Rate: 25Hz
- Edge Trigger: Rising, Falling or Both

Temperature Sensors:

- Maximum Number of Sensors: 16
- Type: Dallas Semiconductor DS18B20
- Temperature Range: -67°F to 257°F (-55°C to +125°C)
- Accuracy: ±0.5°C (from -10°C to +85°C)
- Sensor Functions: Monitor Temperature, Log Temperature, Email Alerts, SNMP Traps
- Humidity Type: Xytronix Model X-DTHS-P sensor
- Humidity Range: 0-100% RH
- Accuracy: ±2%
- Max Cable Length: 600 feet (180m) maximum combined cable length

Real-Time Clock:

- Manual or NTP(Network Time Protocol) setup
- NTP Sync Period: Once, Daily, Weekly, On Powerup
- Auto Daylight Savings Adjustment
- Battery (capacitor) Power Backup

Capacitor Power Backup:

- Backup Functions: Retain Real-Time Clock, 1 Register, 2 Counters, 16 Relay States
- Backup Duration: 2 weeks minimum

Network:

- Type: 10/100 Base-T Ethernet Port
- Setup: Static or DHCP IP address configuration

Connectors:

- Accessory Connector: 5-Position, 3.81mm terminal spacing, Removable. For 16-28 AWG wire
- Network: 8-pin RJ-45

LED Indicators:

Number of LEDs: 13 - Power on, Relay coil energized 1-10, Network linked, Network activity

Physical:

- Size:
    - 9.125in (232mm) wide
    - 4.25in (126mm) tall
    - 2.425in (62mm) deep
- Weight: 22 oz (626 grams)
- Enclosure Material: PVC
- Enclosure Flame Rating: UL94 V1
- Operating Temperature: -40°F to 150°F (-40°C to 65.5°C)

Protocols:

- HTTP(S), XML, Modbus, JSON, SNMP, Remote Services, and Remote Monitoring

Logging:

- Log File Size: 3072K (Max 50688 logs)
- Storage: Nonvolatile Flash
- Buffer Architecture: Circular Buffer
- Log data can be periodically read and stored on a computer

Password Settings:

- Password protection on setup page: Yes
- Password protection on control page: Optional
- Password Encoding: Base 64
- Password Length: 6-18 Characters

Electromagnetic Compliance:

- IEC CISPR 22, CISPR 24
- FCC 47CFR15 (Class B)
- EN55024 ITE Immunity (2010)
- EN55022 Emissions (2010)

Product Safety Compliance:

- IEC 61010-1

Appendix G: Warranty

This Xytronix Research & Design, Inc. product is warrantied against defects in material and workmanship for a period of five years from the date of shipment. During the warranty period, Xytronix Research & Design, Inc. will, at its option, either repair or replace products that prove to be defective. This warranty is extended to the original purchaser of the equipment only.

For warranty service or repair, customer must contact Xytronix Research & Design, Inc. technical support (support@ControlByWeb.com) and obtain a Return Authorization number (RA#). Before issuing an RA#, a support technician will work with customer to try to resolve the issue without returning the product. If technician determines that product must be returned for service an RA# will be issued. Next, the product must be properly packaged and returned to Xytronix Research & Design, Inc. with the RA# clearly marked on the package. The purchaser shall prepay all charges for shipping to Xytronix Research & Design, Inc. For warranty repairs of products less than one year old, Xytronix Research & Design, Inc. will pay the shipping charges to return the product to the purchaser as long as the product is shipped within the continental United States. If the product is shipped outside of the continental United States or the product was shipped more than one year earlier, the purchaser shall pay all shipping charges both ways.

Limitation

The foregoing warranty shall not apply to defects or damage resulting from improper use or misuse, unauthorized repair, tampering, modification, improper connection, or operation outside the electrical/environmental specifications for the product. Further, the warranty does not cover damage from Acts of God, such as lightning, fire, flood, hurricanes and tornadoes. This warranty does not cover damage to property, equipment, direct, indirect, consequential, or incidental damage (including damage for loss of business profit, business interruption, loss of data, and the like) arising out of the use or misuse of this product.

UNDER NO CIRCUMSTANCES WILL THE LIABILITY OF XYTRONIX RESEARCH & DESIGN, INC. TO THE PURCHASER OR ANY OTHER PARTY EXCEED THE ORIGINAL PURCHASE PRICE OF THE PRODUCT, REGARDLESS OF THE FORM OF THE CLAIM. No other warranty is expressed or implied. Xytronix Research & Design, Inc. specifically disclaims the implied warranties or merchantability and fitness for a particular purpose. Some jurisdictions may not allow the exclusion of limitation of liability for consequential or incidental damage.

Appendix H: FCC Statement

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

- This device may not cause harmful interference.
- This device must accept any interference received, including interference that may cause undesired operation.

Warning

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause interference to radio communications. There is no guarantee, however, that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into a relay on a circuit different from where the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Notice

Changes or modification not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

Appendix I: Trademark and Copyright Information

This document is Copyright ©2013-2023 by Xytronix Research & Design, Inc. All rights reserved.

WebRelay-10 Plus™,WebRelay, ControlByWeb™, and Xytronix Research & Design™ are trademarks of Xytronix Research & Design™, Inc. 2005-2023.

All other trademarks are the property of their respective owners.

All parts of this product and design including but not limited to firmware, hardware design, schematics, PCB layout, concept, graphics, users manual, etc., are property of Xytronix Research & Design, Inc. ©2005-2023. WebRelay-10 Plus™ may not be opened, disassembled, copied or reverse-engineered.

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying or scanning, for any purpose other than the personal use by the purchaser of this product. Xytronix Research & Design, Inc., assumes no responsibility for any errors that may appear in this document.

Whereas reasonable effort has been made to make the information in this document as useful and accurate as possible, Xytronix Research & Design, Inc. assumes no responsibility for the application, usefulness, or completeness of the information contained herein. Under no circumstance will Xytronix Research & Design, Inc. be responsible or liable for any damages or losses including direct, indirect, special, incidental, or consequential damages or losses arising from either the use of any information contained within this manual or the use of any products or services referenced in this manual.

Xytronix Research & Design, Inc. reserves the right to change any product's features, specifications, documentation, warranties, fee schedules, and conditions at any time and without notice.

Appendix J: Mechanical Information

Voltage Input
9-28VDC
*(Locations may vary
on older modules)*