

# WebRelay-Dual™

## Users Manual



Revision: 1.2 - July 5, 2007

Covers: X-WR-2R1-2I5-5  
X-WR-2R1-2I5-I  
X-WR-2R1-2I24-I  
X-WR-2R1-2I5-E

**CONTROL** by **WEB**™  
[www.ControlByWeb.com](http://www.ControlByWeb.com)

a division of...  
**Xytronix Research & Design, Inc.**  
North Logan, Utah, USA

## Contents

Trademark and Copyright Information

Warranty

FCC Statement

Installation Guidelines (Read Before Installing)

Section 1: Introduction

1.1 Features

1.2 WebRelay-Dual™ Models Available

1.3 Connectors & Indicators

1.4 Example Configurations and Applications

1.4.1 Basic WebRelay-Dual™ Configuration

1.4.2 Single WebRelay-Dual™ Device (Voltage and/or IP Network Control)

1.4.3 Two WebRelay-Dual™ Units Extends Digital Signals (No Computer Required for Operation)

1.4.4 Multiple WebRelay-Dual™ Operation

Section 2: Installation and Setup

2.1 Mounting

2.1.1 Wall Mounting

2.1.2 DIN-Rail Mounting

2.2 Connection

2.2.1 Power Supply Connection

2.2.2 Network Connection

2.2.3 Relay Connection

2.2.4 Optically-Isolated Input Connection

2.3 Establishing Communications for Setup

2.3.1 Option 1: Assign a temporary IP address to WebRelay-Dual™

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

2.3.3 Open Configuration Web Page

2.4 Web-Based Setup

2.4.1 Main Setup Page

2.4.2 Network Setup Page

2.4.3 Password Setup Page

2.4.4 Relay/Input Setup Pages

2.4.5 Events

2.4.6 Control Page Setup

Section 3: Operation

3.1 Browser Operation

3.2 XML Operation

3.2.1 Controlling The Unit Through state.xml

3.2.2 Monitoring Events Through eventX.xml

3.2.3 Calibrating The Real Time Clock Through rtc\_cal.xml

3.3 Configuring Events Through Get Requests

3.3.1 Specifying the Event to be Modified

3.3.2 Modifying the Event Settings

### 3.4 Modbus Operation

- 3.4.1 Read Coils (Modbus Function Code 01 (0x01))
- 3.4.2 Read Discrete Inputs (Modbus Function Code 02 (0x02))
- 3.4.3 Read Holding Registers (Modbus Function Code 03 (0x03))
- 3.4.4 Write Single Coil (Modbus Function Code 05 (0x05))
- 3.4.5 Write Multiple Coils (Modbus Function Code 15(0x0F))
- 3.4.6 Write Multiple Registers (Modbus Function Code 16 (0x10))

Appendix A: Restoring Factory Default Settings

Appendix B: Specifications

Appendix C: Mechanical Information

Appendix D: Power Failure

Appendix E: Open TCP Legal Notice

## Trademark and Copyright Information

This document is Copyright ©2005-2007 by Xytronix Research & Design, Inc. All rights reserved. WebRelay-Dual™ and ControlByWeb™ are Trademarks of Xytronix Research & Design, Inc. 2005-2007.

Portions of the software used in WebRelay-Dual™ are open source and appropriate copyright and legal notices are listed at the end of this manual. All other parts of the software are property of Xytronix Research & Design, Inc. ©2005-2007. The hardware design, schematics, PCB layout, concept, graphics, users manual, etc., are property of Xytronix Research & Design, Inc. ©2005-2007. WebRelay-Dual™ may not be opened, dis-assembled, 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 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.

## Warranty

This Xytronix Research & Design, Inc. product has a warranty against defects in material and workmanship for a period of one year 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, the product must be properly packaged, and returned to Xytronix Research & Design, Inc. The purchaser shall prepay all charges for shipping to Xytronix Research & Design, Inc., and 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 United States. If the product is shipped outside of the United States, the purchaser shall pay all shipping charges, duties, and taxes.

### 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 Acts of God, such as 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.

## 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 (Class A for POE models) 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. However, there is no guarantee 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 an outlet on a circuit different from that to which 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.

## Installation Guidelines (Read Before Installing)

- Do not open the WebRelay-Dual™ enclosure. This will void the warranty.
- This unit must be installed by qualified personnel.
- This unit must not be installed directly outdoors.
- 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.

Notes about security:

By design, WebRelay-Dual™ is very secure. It does not support terminal or file transfer programs such as telnet, ftp, ssh, etc. This means that it is not possible for someone to 'break in' to WebRelay-Dual™ and access other devices on your local network. WebRelay-Dual™ does not support remote firmware updates which means that it is not possible for someone to remotely install malicious software. The simplicity of WebRelay-Dual™ makes it a very secure device. As with any device to be installed on a network, there are some security precautions that should be observed. If WebRelay-Dual™ is installed on the Internet, it is recommended that passwords be enabled for the control page. Make sure secure passwords are used. Passwords should be at least 8 characters in length and should be a combination of upper case letters, lower case letters, and numbers. Don't use passwords that would be easy to guess. For additional security, a firewall may be used to limit access only to selected IP addresses. Another option may be to set up a Virtual Private Network (VPN) between the network where WebRelay-Dual™ resides and the client machine (web browser, second WebRelay-Dual™, etc.).

## Section 1: Introduction

WebRelay-Dual™ is an Ethernet based, high performance, mini I/O module. It has two low-signal relays, two optically-isolated inputs, and a real-time clock. WebRelay-Dual™ 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-Dual™ using a web browser or by using custom applications. Computers, PLCs, or automation controllers may control and monitor WebRelay-Dual™ without user intervention. This works by sending text commands over the network and reading XML status pages from WebRelay-Dual™, or by using Modbus/TCP protocol. WebRelay-Dual™ can also be operated by using the optically isolated inputs. The optically isolated inputs can be configured to turn the relays on/off, pulse the relays, toggle the relays, latch the relays (for alarming), etc. The inputs can also be used to control a relay at a remote location or to simply monitor the status of digital signals. WebRelay-Dual™ also has a real-time clock that can be used to turn the relays on and off at preset times.

WebRelay-Dual™ has many applications and is only limited by imagination. 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. It can also be used to monitor switches, sensors, and control voltages.

### 1.1 Features

WebRelay-Dual™ is very simple but has many features. Some features include...

- No programming required.
- Easy to use as a stand alone device or as part of a large control system.
- Built-in web server provides simple, web based configuration and control.
- Control page can be customized with appropriate text and buttons
- 10/100 Ethernet connectivity.
- Two 1 Amp / 24 Volt relays.
- Two optically-isolated inputs can be used for status monitoring, local relay control, or remote relay control.
- Real-time clock allows relays to be programmed to turn on/off at preset times. Up to six one-time or repeated events can be set up.
- Removable terminal connector (included) simplifies wiring and service.
- XML status pages provides an easy way to interface with custom applications.
- Modbus/TCP protocol support provides inter-interoperability with devices/software from other manufacturers.
- Password protection.
- Selectable TCP ports.
- On/Off or pulse modes.
- Input high timer automatically keeps track of time that input 1 is high (XML readable only)

### 1.2 WebRelay-Dual™ Models Available

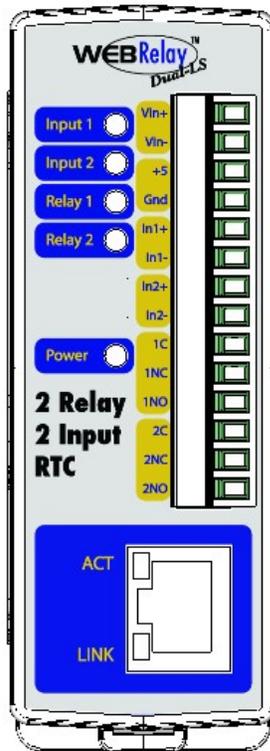
WebRelay-Dual™ is currently available in four models. There are only two differences between the four models. One difference is the power supply requirements and the other difference is the input voltage range on the optically-isolated inputs.

Part Number	Power Supply Requirements	Optically-Isolated Input Voltage Range
X-WR-2R1-2I5-5	5VDC	3-11VDC
X-WR-2R1-2I5-I	9-28VDC	3-11VDC
X-WR-2R1-2I24-I	9-28VDC	11-28VDC
X-WR-2R1-2I5-E	Power Over Ethernet or 5VDC	3-11VDC

### 1.3 Connectors & Indicators

WebRelay-Dual™ has a 14-position removable terminal connector and an Ethernet connector. The terminal connector is used to provide power to the module, provide connection to the relay contacts, provide connection to the optically-isolated inputs, and to provide a 5V output voltage that can be used to connect dry-contacts to the inputs.

There are seven LED indicators on WebRelay-Dual™. One LED is labeled Power, which is illuminated (green) whenever the module is powered. Two LEDs are labeled Input 1 and Input 2. These are illuminated (green) whenever a voltage is applied to the corresponding input. Two LEDs are labeled Relay1 and Relay2. These are illuminated (green) whenever the corresponding relay coil is energized. When the relay coil is energized, the NO (Normally Open) contact is closed and the NC (Normally Closed) contact is open. The other two LEDs are located on the Ethernet connector. They are labeled LINK and ACT. LINK is illuminated (green) when the module is properly connected to an Ethernet network. This LED must be illuminated before any communications can take place with the module. The ACT LED flashes when activity is detected on the network.

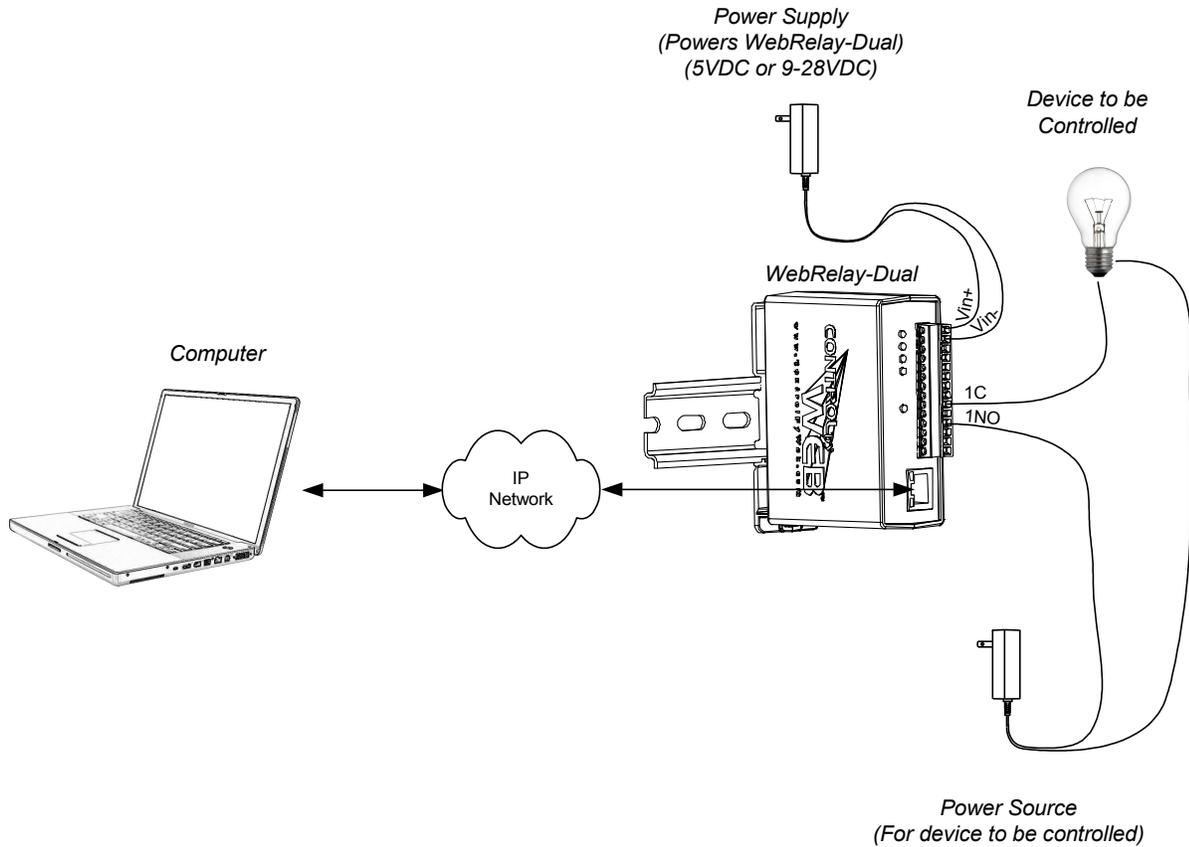


## 1.4 Example Configurations and Applications

WebRelay-Dual™ is very versatile and can be used in many applications. Several basic installation schemes are illustrated in this section.

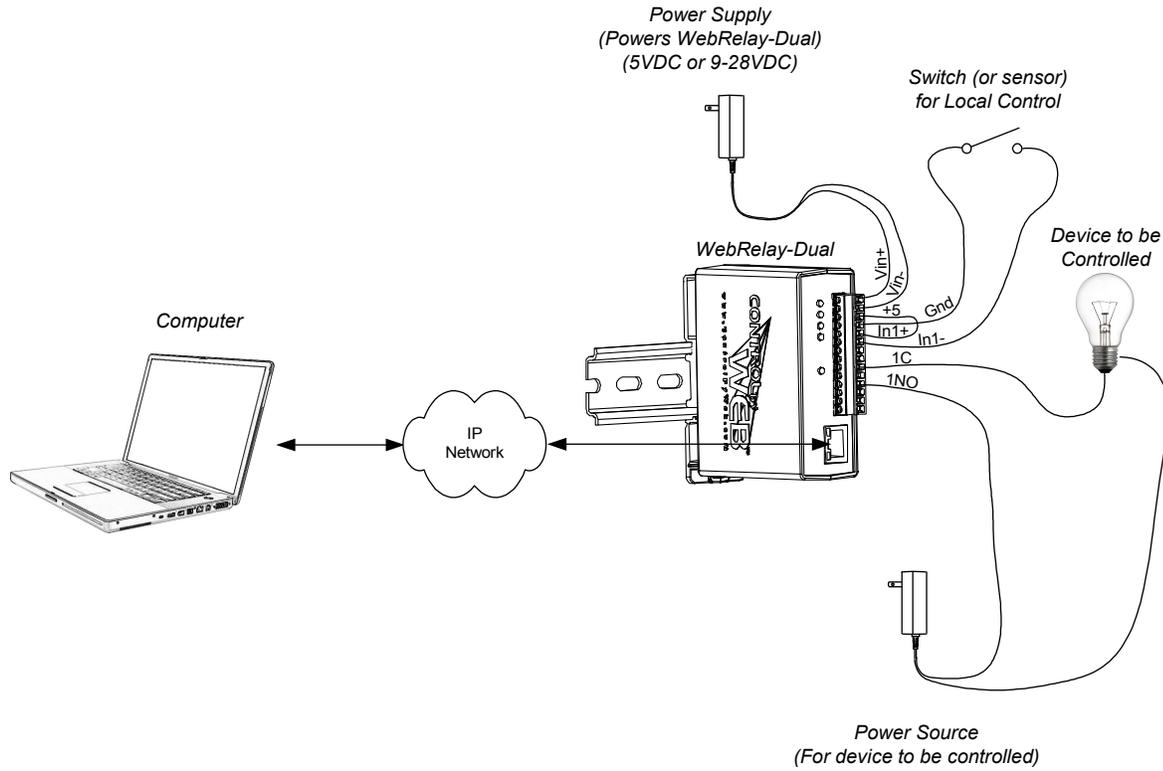
### 1.4.1 Basic WebRelay-Dual™ Configuration

The illustration below shows a simple example of using WebRelay-Dual™ to control a device over an IP network. The device to be controlled is wired in series with the relay contacts. This configuration is used for control of a device using a web browser or a control script/application running on a computer. Note that WebRelay-Dual™ has two internal relays so a second device could be controlled separately.



**1.4.2 Single WebRelay-Dual™ Device (Voltage and/or IP Network Control)**

This configuration offers both network and local control. In the example, the load shown (light) can be controlled locally using a mechanical switch, or remotely over the IP network. The mechanical switch controls a 5 volt signal that is applied to the optically-isolated input. The 5 volt source comes from the +5 and Gnd terminals. Each time the 5 volt signal changes (turns on or off), WebRelay-Dual™ changes the state of the relay.



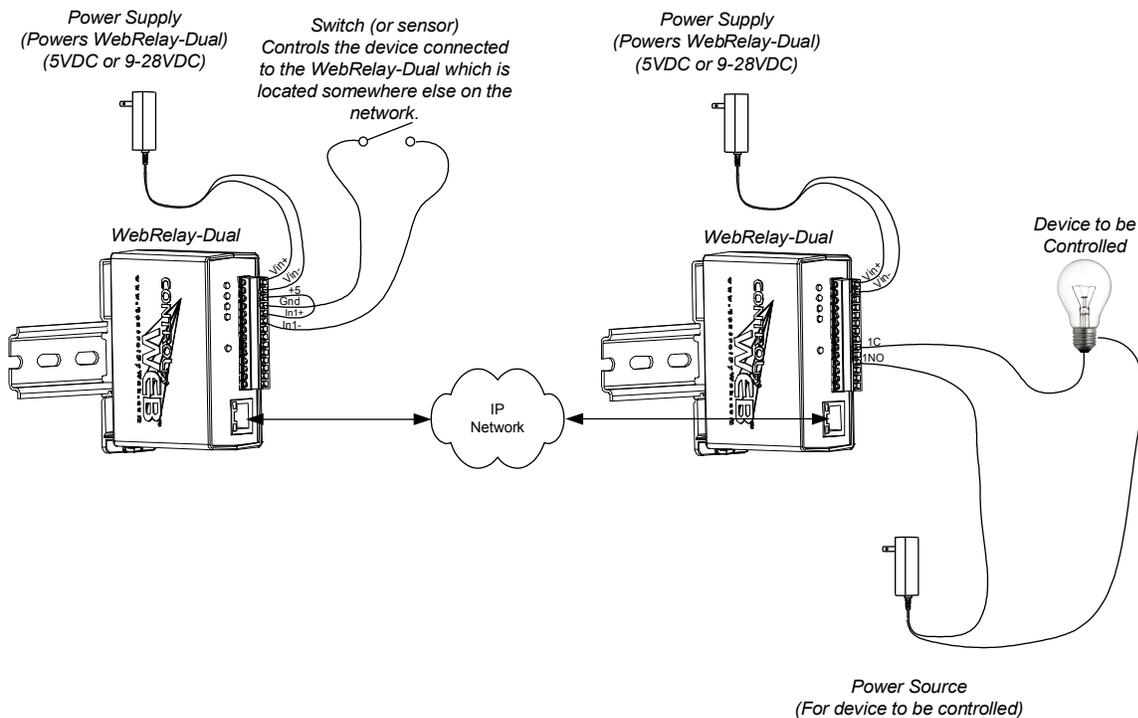
**1.4.3 Two WebRelay-Dual™ Units Extends Digital Signals (No Computer Required for Operation)**

This configuration is used to control devices remotely without a browser, or to simply extend a digital signal to a remote location on the network. A control voltage applied to one unit controls the relay contacts of another unit at a remote location. WebRelay-Dual™ can send and receive messages to/from other ControlByWeb™ devices such as WebRelay™, WebRelay-Quad™, the DAQ Series™ Five Input Module, etc.

This configuration has many uses. Two examples are briefly described below.

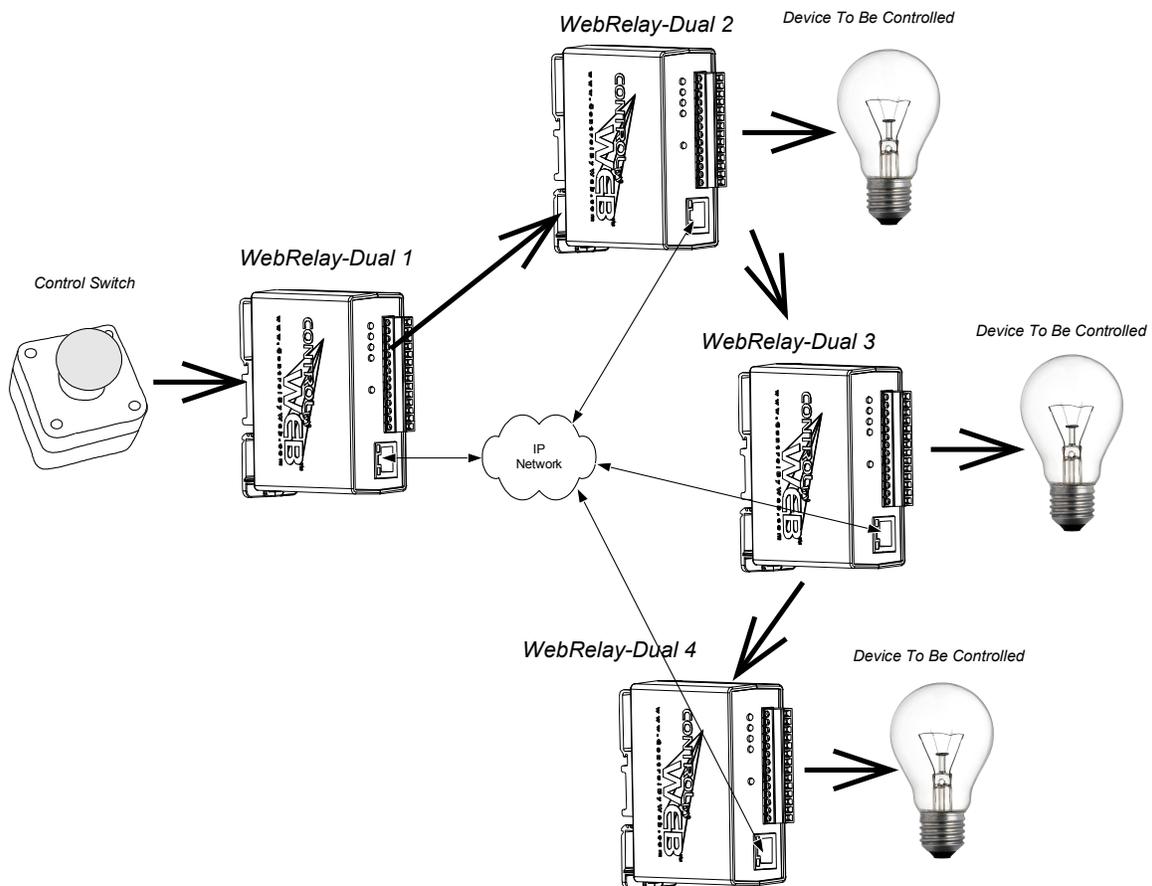
For security applications, the switch in the example could be replaced with a sensor, and the device to be controlled could be a buzzer or chime. A business owner, for example, could set up a sensor at his/her office and a chime at his/her home. The chime at the home would announce each time the sensor detects entry to the office (which may be miles away). When the chime indicates office entry late at night, the business owner can take appropriate action.

Another way this configuration is useful is to extend the output of a PLC (Programmable Logic Controller) in a factory to control motors, lights, etc., in another building. Digital outputs from a PLC are connected to the inputs of one WebRelay-Dual™, and the load devices (motors, lights, etc.) is connected to the relay contacts of a second WebRelay-Dual™ (loads larger than 1 Amp at 28V must use an external relay). This allows any PLC to control devices across an IP network. This configuration is also useful to connect digital signals (on/off) from a remote location to the input of a PLC.



#### 1.4.4 Multiple WebRelay-Dual™ Operation

For some applications, it is useful to for one WebRelay-Dual™ to control multiple WebRelay-Duals at remote locations. There is not a way to configure a single WebRelay-Dual™ to control multiple WebRelay-Duals directly, but it is possible to set up multiple WebRelay-Duals in a daisy-chain mode so that one WebRelay-Dual™ controls a second WebRelay-Dual™ which, in turn controls a third WebRelay-Dual™ and so on. In the illustration below, each WebRelay-Dual™ is numbered (1, 2, 3, and 4). A mechanical switch controls WebRelay-Dual™ 1, and WebRelay-Dual™ 1 controls WebRelay-Dual™ 2, which controls WebRelay-Dual™ 3, which controls WebRelay-Dual™ 4. This daisy-chain mode is set up by configuring the Relay options in WebRelay-Duals 1, 2 and 3 to “*remote command equals local relay*”. Note that the relay contacts of each WebRelay-Dual™ will be set to the same state as the contacts of the WebRelay-Dual™, that is configured to control it and not necessarily to the state of the switch. In other words, if the switch is 'ON' but a user changes the state of WebRelay-Dual™ 3 to 'OFF' by changing the relay in the web page, WebRelay-Dual™ 4 will be changed to the 'OFF' state also. Note also that in this example, WebRelay-Duals 2, 3, and 4 are each controlling load devices and WebRelay-Dual™ 1 is not, but a load could be connected to WebRelay-Dual™ 1 as well.



## Section 2: Installation and Setup

Installation consists of mounting WebRelay-Dual™, connecting to an IP network, providing power, configuring via a web browser, and wiring relay contacts to the device that will be controlled. Optionally, control signals may be connected to the optically-isolated inputs.

### 2.1 Mounting

WebRelay-Dual™ can be wall mounted or DIN rail mounted. It should be mounted in a clean dry location where it is protected from the elements. Ventilation is recommended for installations where ambient air temperature is expected to be high.

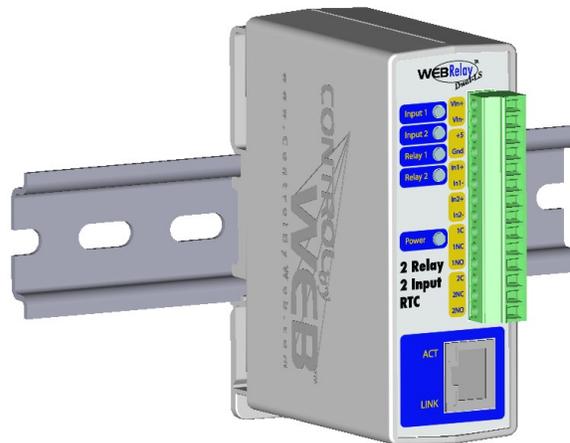
#### 2.1.1 Wall Mounting

Mount WebRelay-Dual™ to a wall by using two #8 screws. Attach the screws to the wall vertically spaced exactly 2.5 inches apart. The head of the screws should be about 1/10 inch away from the wall. See Appendix C for mechanical details.



#### 2.1.2 DIN-Rail Mounting

WebRelay-Dual™ can be mounted to a standard (35mm by 7.55mm) DIN rail. Attach WebRelay-Dual™ to the DIN rail by hooking the top hook on the back of the enclosure to the DIN rail and then snap the bottom hook into place. Remove WebRelay-Dual™ from DIN rail using a flat-head screwdriver. Insert the screwdriver into the notch in the release tab and pry against the enclosure to release the bottom hook.



## 2.2 Connection

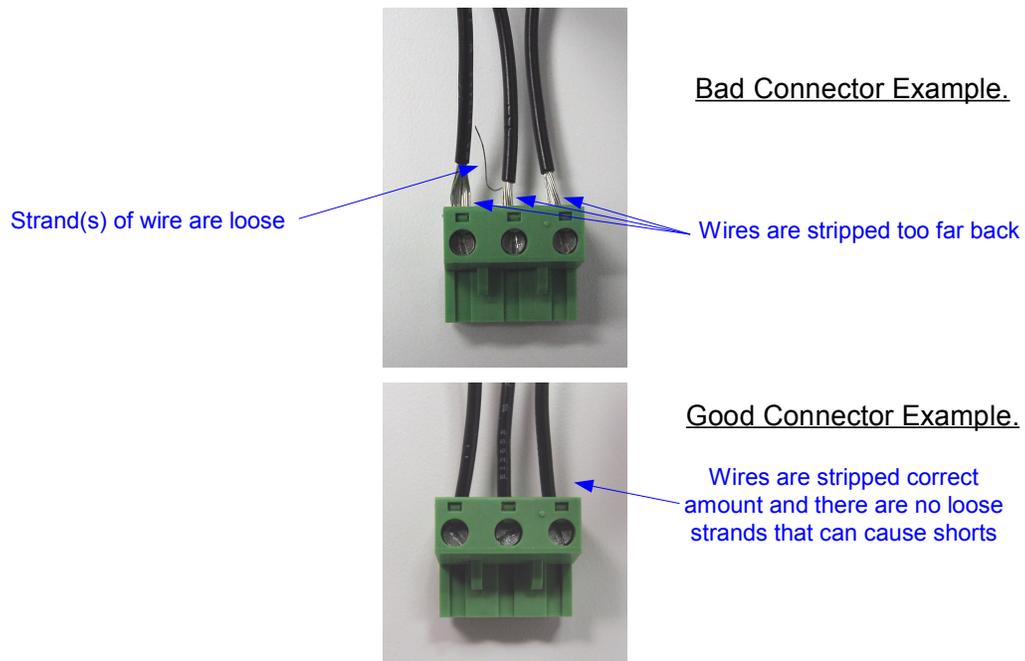
CAUTION: MAKE SURE POWER IS SHUT OFF BEFORE WIRING!

Two removable terminal connectors are provided for simple wiring to WebRelay-Dual™. The correct wiring procedure is as follows;

1. Make sure power is turned off.
2. Remove terminal connector from WebRelay-Dual™ and make wiring connections to the terminals.
3. Reconnect terminal connectors.
4. Apply power.

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

IMPORTANT: MAKE SURE WIRES ARE PROPERLY ATTACHED TO THE TERMINALS!



**14-pin connector pinout:**

Pin	Description
VIn+	Power supply input. Connect to the positive side of the appropriate power supply. DO NOT EXCEED MAXIMUM POWER SUPPLY VOLTAGE.  <u>Model X-WR-2R1-2I5-5</u> 5VDC Power supply  <u>Models X-WR-2R1-2I5-I and X-WR-2R1-2I24-I</u> 9~28VDC Power supply  <u>Model X-WR-2R1-2I5-E</u> Power Over Ethernet model. These units do not require a power supply to be connected to this pin when the unit is connected to an 802.3af compliant network. Alternatively, this model can be powered using a 5VDC power supply.
VIn-	Vin- Negative power supply input.
+5	+5VDC Out. This voltage is provided as a convenient voltage source for the optically-isolated inputs . This is used, for example, when dry contacts (such as a switch) are needed to control the input state. This can be used by connecting the INPUT- directly to Gnd, and then connecting this terminal to INPUT+ through a switch. Do not use this output for other purposes. Note that this voltage is active for all models but is only useful for models with 3-12V optical inputs.
Gnd	Negative side of the 5VDC output voltage.
In1+	This is the positive side of the optically-isolated input 1.
In1-	This is the negative side of the optically-isolated input 1.
In2+	This is the positive side of the optically-isolated input 2.
In2-	This is the negative side of the optically-isolated input 2.
1C	Relay 1 Common Contact
1NC	Relay 1 Normally Closed Contact
1NO	Relay 1 Normally Open Contact
2C	Relay 2 Common Contact
2NC	Relay 2 Normally Closed Contact
2NO	Relay 2 Normally Open Contact

**2.2.1 Power Supply Connection**

WebRelay-Dual™ requires power for its internal logic circuits. Connect appropriate power supply to the VIn+ and VIn- terminals. Alternatively, WebRelay-Dual™ units with the Power Over Ethernet option may be powered through the Ethernet connection instead of using an external power supply.

Multiple WebRelay-Dual™ 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 specifications for current requirements for a specific model number).

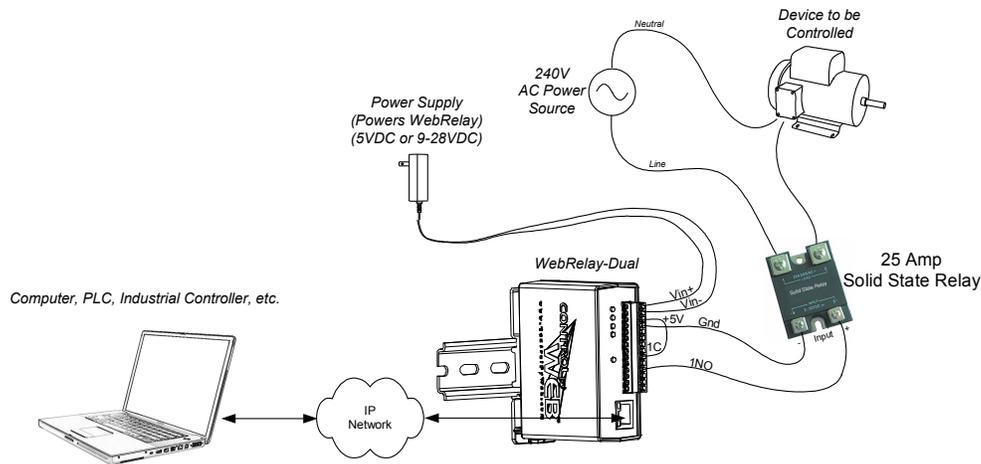
WebRelay-Dual™ units with the Power Over Ethernet option may be connected to an 802.3af compliant Ethernet port instead of connecting an external power supply. In this case, WebRelay-Dual™ is powered through the network port. In 802.3af compliant networks, a 48 Volt power source is injected into the Ethernet line. This power source is provided by an 802.3af compliant hub, switch, or power injector which may be located in a utility closet which could be a distance away from WebRelay-Dual™. This option is very useful for installations where local power is not available. The power injector may inject the 48 Volt power source through the data lines or the spare lines in the Ethernet cable. **DO NOT USE A POWER INJECTOR THAT APPLIES A VOLTAGE TO BOTH THE DATA LINES AND THE SPARE LINES.**

### 2.2.2 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-Dual™ 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-thru” cable should be used.

### 2.2.3 Relay Connection

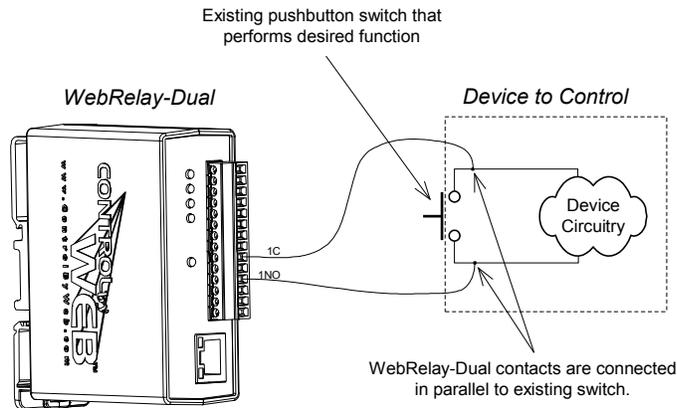
The relay contacts internally connect directly to the terminal connector. Direct access to Common, Normally Open, and Normally Closed contacts are provided. The relay contacts may be wired in series with the power source for the load (device to be controlled) as long as the load does not exceed the maximum current and voltage of the relay contacts.



For loads greater than 1 Amp, an external relay may be used. The illustration below shows how a 20-Amp motor can be controlled using an external relay. In the example, WebRelay-Dual™ controls the external relay and the external relay controls the load.

Many of the illustrations in this manual show WebRelay-Dual™ contacts wired in series with a power source which allows the power for the device to be switched on and off remotely. In many cases, however, it is useful to control other functions and not just power. One way to do this is to connect relay contacts in parallel with pushbutton switches that control a device. An example is useful to illustrate this. Suppose you have a gate controller system that has a pushbutton switch that opens a gate when pressed. The button that controls the gate is located on one side of the building, but you want to be able to open the gate from a room on the other side of the building. Since the building is pre-wired for networking, it is a simple task to accomplish this goal by connecting the WebRelay-Dual™ contacts in parallel with the push-button. An illustration of this is shown below. In doing this, users can now operate the gate with the existing pushbutton, or users can operate the gate from another room using a web browser. Another option (not shown here), is to connect another pushbutton switch to the inputs of a second WebRelay-Dual™ device in a room across the building. The second device could be configured to send control messages to the first device each time the new pushbutton is pressed. As a result, there is a second pushbutton that could

operate the gate.



### 2.2.4 Optically-Isolated Input Connections

The optically-isolated inputs can be used to control the internal relays, control remote relays (over the network), or simply to monitor the state of a discrete device. To use these inputs, connect a small DC control voltage directly to the input and set up the function of the input using the configuration pages. A current limiting resistor is provided internally, so no external resistors are required as long as the maximum input voltage is not exceeded. If an AC signal, or a signal greater than rated input voltage needs to be detected by WebRelay-Dual™, use a signal conditioner to convert the signal to a DC signal within the input range. Signal conditioners can be made of discrete components or can be purchased at industrial automation distributors.

To connect an AC voltage to the input of WebRelay-Dual™, a diode (or bridge rectifier) and a capacitor can be used to convert the AC signal to a DC signal. To connect a voltage to the input of WebRelay-Dual™ that is greater than the input is rated, a resistor of the appropriate value and power rating can be used to reduce the voltage.

The formulas to calculate external resistor values is provided below.

For ControlByWeb™ products with a 3-12VDC input range...

$$R = (V-9)/0.020$$

V = Desired Vin

R = External resistor value required. The external resistor must also be rated with a minimum power value (minWatts=(V-9)x0.020)

For ControlByWeb™ products with a 11-28VDC input range...

$$R = (V-28)/0.014$$

V = Desired Vin

R = External resistor value required. The external resistor must also be rated with a minimum power value (minWatts=(V-28)x0.014)

Example (to connect a 24VDC signal voltage to WebRelay-Dual™ with a 3-11VDC input range):

$R = (24-9)/0.020=750$  Ohm resistor. The resistor must be at least  $(24-9) \times 0.020 = 0.3$  Watts so use a ½ Watt resistor.

## 2.3 Establishing Communications for Setup

WebRelay-Dual™ is set up using a web browser. The first task is to establish communications between a computer and the WebRelay-Dual™ device so that the browser-based configuration can begin. To do this, the computer and WebRelay-Dual™ 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 WebRelay-Dual™ so that they are on the same network. The first way (Option 1), is to change the IP address of WebRelay-Dual™ to an address that is on the same network as the computer. The second way (Option 2) is to change the IP address of the computer to an address that is on the same network that WebRelay-Dual™ is set to by default.

### 2.3.1 Option 1: Assign a temporary IP address to WebRelay-Dual™

This option is used to TEMPORARILY assign an IP address to WebRelay-Dual™ without the need to change the IP address of the configuration computer. Note that WebRelay-Dual™ will only use this IP address as long as power is maintained. Once power is lost and restored, WebRelay-Dual™ 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 page using the browser.

#### To assign the temporary IP address...

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

#### Windows:

Open a Command Prompt (on Windows XP, select START, then RUN, then type “cmd”).

Type...

```
arp -s {new IP address} {serial number of WebRelay-Dual™ }
```

Note: IP address format is: xxx.xxx.xxx.xxx

Serial number format is: ss-ss-ss-ss-ss-ss

For example, to set a WebRelay-Dual™ device (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
```

#### Linux/Unix:

Open a terminal, change to root user (su -, then enter root password).

Type...

```
arp -s {new IP address} {serial number of WebRelay-Dual™ }
```

Note: IP address format is: xxx.xxx.xxx.xxx

Serial number format is: ss:ss:ss:ss:ss:ss

For example, to set a WebRelay-Dual™ device (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
```

### Mac OS X

Open a terminal,

Note that the terminal is in the “Utilities” directory which is in “Applications” directory.

```
type
```

```
sudo arp -s {new IP address} {serial number of WebRelay-Dual™ }
```

Note: Administrator password is required.

IP address format is: xxx.xxx.xxx.xxx

Serial number format is: ss:ss:ss:ss:ss

For example, to set a WebRelay-Dual™ device (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.

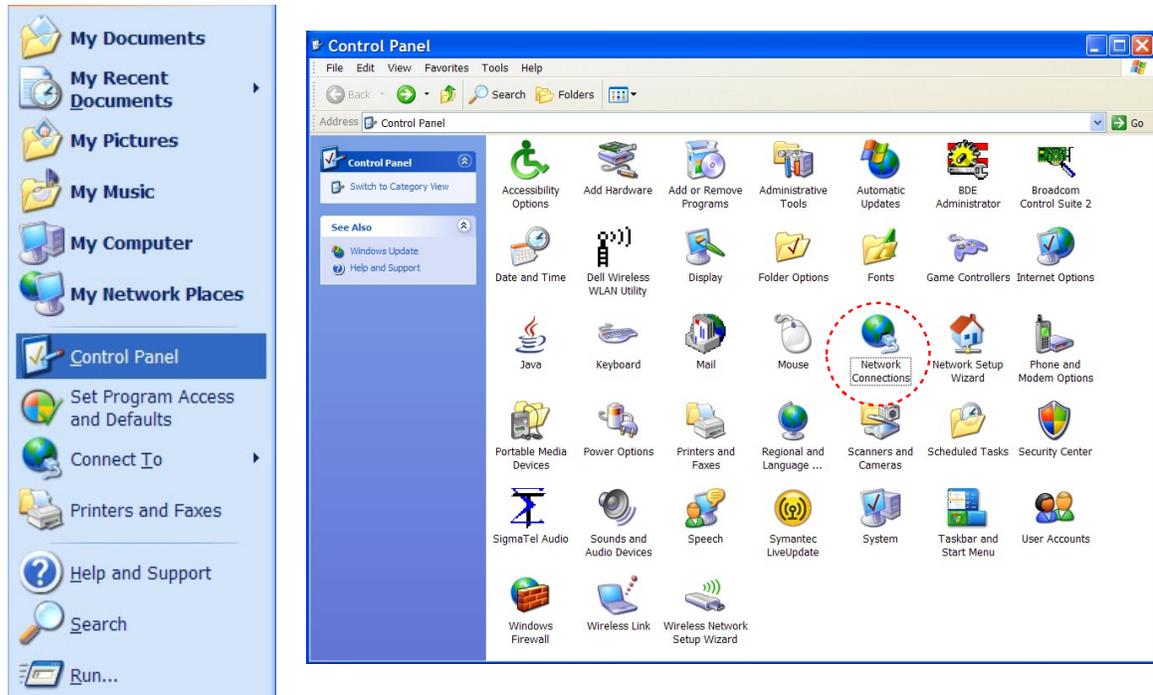
```
ping -s 102 10.10.10.40
```

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

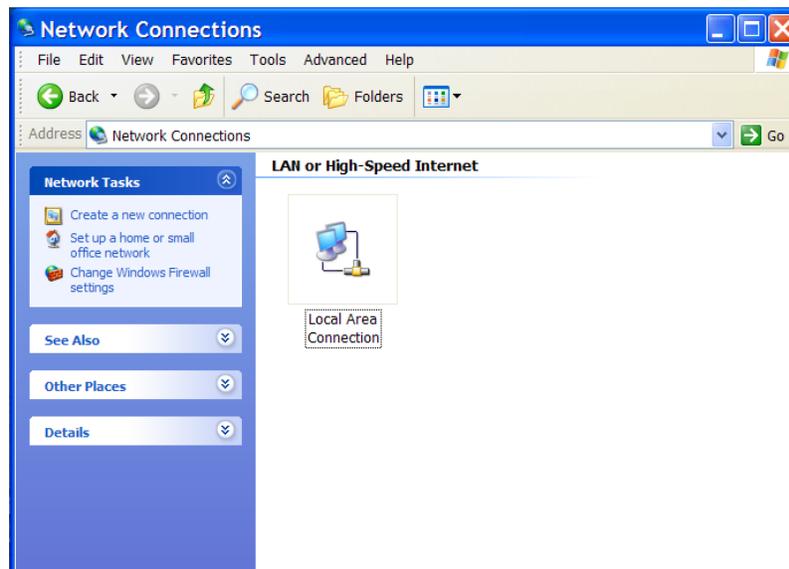
If the first option above is not used, you can use this option to communicate with WebRelay-Dual™ . By default, WebRelay-Dual™ comes from the factory with an IP address of 192.168.1.2. Communications with WebRelay-Dual™ may be established by assigning an IP address to the configuration computer that is on the same network as WebRelay-Dual™ (for example the configuration computer could be assigned to 192.168.1.5) .

Instructions for changing the IP address of the computer that will be used for WebRelay-Dual™ configuration are given here. Note that these instructions are specifically for computers with the Windows XP operating system. For setup using other operating systems, refer to the appropriate users manual.

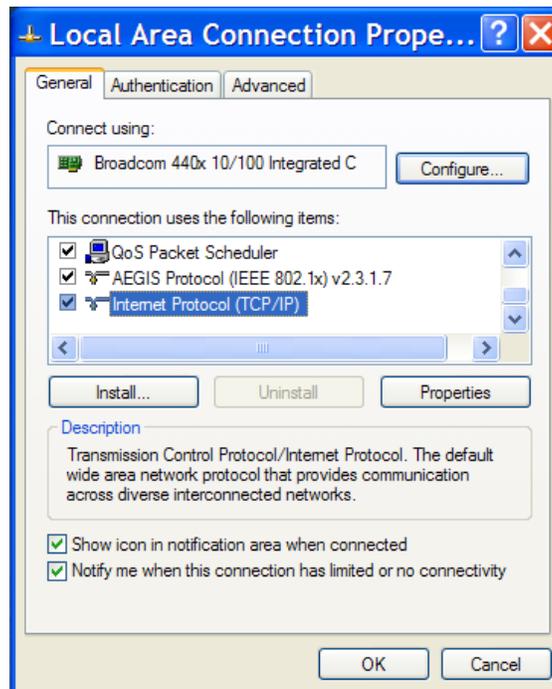
Step 1: Open the control panel by clicking on the start menu and then clicking on Control Panel. (Note that control panel shown is in “Classic View”. If control panel is in “Category View” select the “Classic View” option before proceeding.)



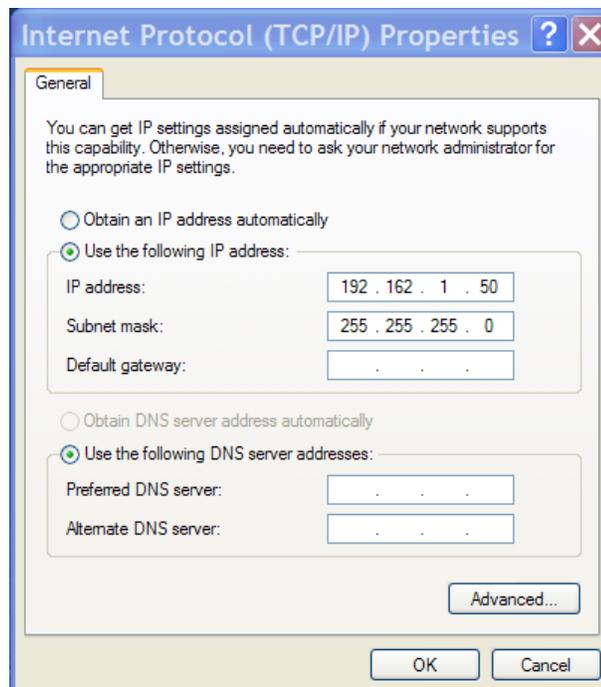
Step 2: Double click on the icon labeled Network Connections. The following menu will pop up.



Step 3: Right click on the icon labeled Local Area Connection. Another menu will appear. Select the option at the bottom of the menu labeled Properties. The Local Area Connection Properties window will appear.



Step 4: On the Local Area Connection Properties page scroll down to Internet Protocol (TCP/IP), select it, and then click the button labeled properties.



Step 5: Before making any changes to the network settings, write down the current settings so that they can be restored once WebRelay-Dual™ is configured. Next, 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. Leave the

default gateway field blank. Click OK to apply the new settings.

### 2.3.3 Open Configuration Web Page

Once the network is set up, open the configuration setup page by typing the following URL into the browser: <http://192.168.1.2/setup.html> (note that if option 1 above was used for initial configuration, replace the IP address given here with the newly assigned IP address). A password is required to change any parameters. The default password is 'webrelay' (do not include quotes, password is case sensitive).

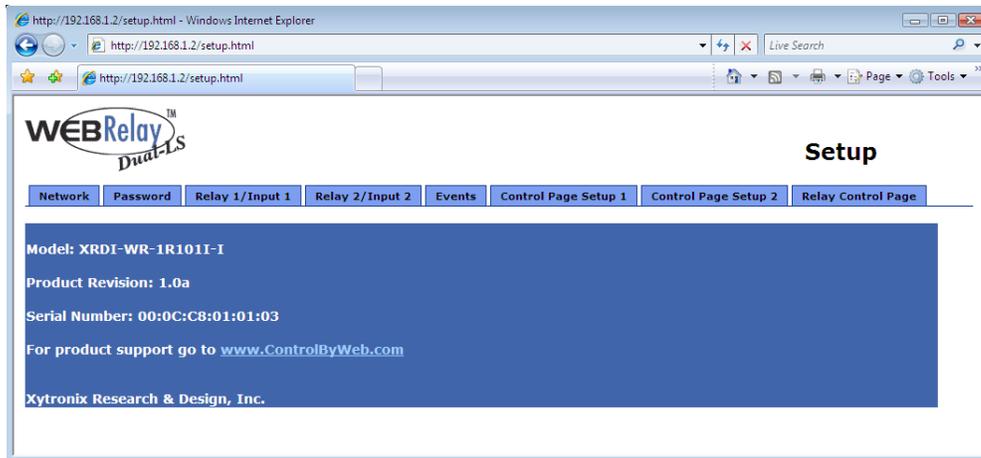
## 2.4 Web-Based Setup

WebRelay-Dual™ is fully configurable through HTML 4.0 compliant web browsers such as Internet Explorer and Mozilla Firefox. It's easy to use tab based menu system has been designed to allow the unit to be configured easily. Note that in this chapter, the default IP address of 192.168.1.2 is used in all examples. If the IP address has been changed, substitute the new IP address for the address shown in the examples.

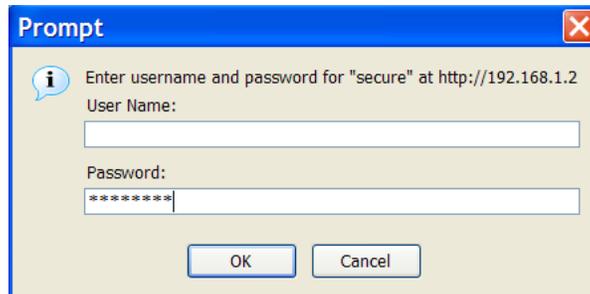
Before proceeding, make sure a network connection has been established between the computer and WebRelay-Dual™. This is done by typing the following URL into the web browser: <http://192.168.1.2/setup.html>. Another way to check communications is to ping WebRelay-Dual™ (from the command prompt (type ping 192.168.1.2)). Each setup page is described below.

### 2.4.1 Main Setup Page

This is the initial page that appears when the URL <http://192.168.1.2/setup.html> is entered into the web browser. It provides basic information about the WebRelay-Dual™ unit.



Navigating between setup pages is done by clicking on the tabs at the top of the page. All setup pages require a password. The default password is 'webrelay' (no quotes, all lower case) and no username is required.



Each setup page has a "Submit" button and a "Reset" button at the bottom of the page. After entering the desired parameters into each page, the "Submit" button must be pressed before any parameters will be saved. If a mistake is made in entering the parameters, the "Reset" button may be used to restore all parameters on the page to their current settings. The "Reset" button is only effective before the "Submit" button is pressed.

#### 2.4.2 Network Setup Page

The network parameters are changed on this page. Note that if multiple WebRelay-Dual™ units are used on the same network, install one unit at a time and set the IP address of each unit before connecting the next unit to the network. This avoids having multiple WebRelay-Dual™ units installed on the network with the same factory default IP address at the same time. It may be necessary to clear the arp cache each time you swap WebRelay-Dual™ units on the network (this is because each unit has the same default IP address but different mac address). This is done by typing `arp -d` in the command prompt of a Windows computer (`arp -d -a` as super user on Apple OSX). Also note that the unit must be power-cycled (power disconnected, then reconnected) before network settings take effect. No other setup page requires power-cycling for the settings to take effect.

##### 1. IP Address:

WebRelay-Dual™ requires a static IP address. This is a unique address that identifies WebRelay-Dual™ on the network. Dynamic IP address assignment is not supported. The lack of dynamic IP addressing support is intentional because a dynamically changing the IP address would make it difficult for a client to access the web server built into WebRelay-Dual™. The IP address is specific to the network where WebRelay-Dual™ will be installed, and must be obtained from the network administrator.

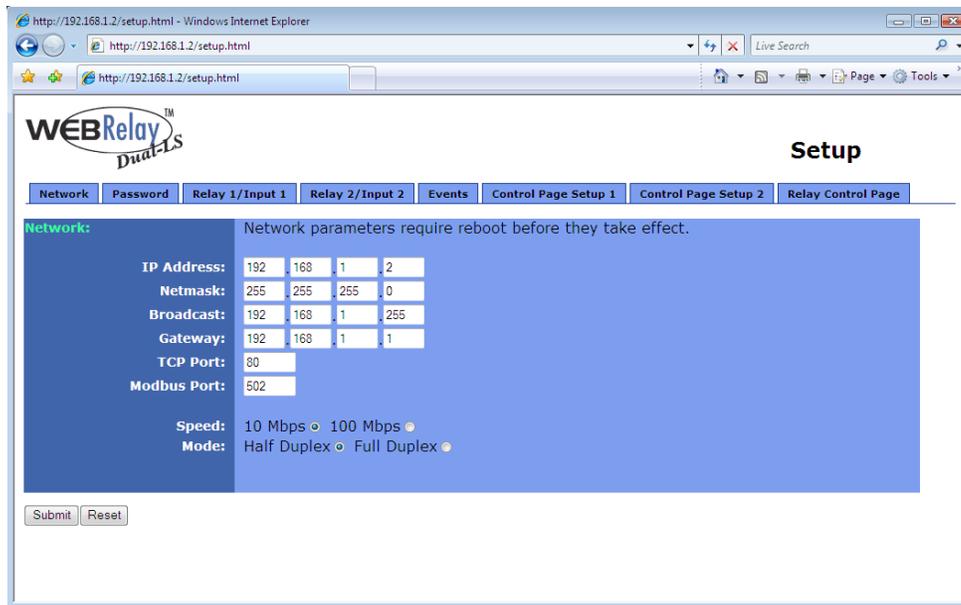
This guide is not meant to be a tutorial on IP addressing, however a few comments about IP addressing are given here.

If WebRelay-Dual™ will be used over the Internet, the IP address must be a routable address assigned by the upstream Internet Service Provider (ISP).

In cases where the ISP only provides a single routable IP address for the entire network (this is typical with ISPs such as cable providers), a proxy server (or gateway router) may be used. A proxy server allows multiple devices to connect to the Internet using a single routable IP address. Many small routers from LinkSys, Dlink, and Netgear perform proxy server functions. If a proxy server is used, WebRelay-Dual™ will not be accessible from the Internet until the proxy server is properly configured (forward proper port to WebRelay-Dual™). This information is mentioned for convenience but details of setting up a configuration such as this is beyond the scope of this manual.

If WebRelay-Dual™ is used on a private network only and is NOT used over the Internet, a routable IP address is not necessary. This may be the case when WebRelay-Dual™ is used to control (or monitor) a device in another room or a nearby building.

If WebRelay-Dual™ will be installed on a simple, private network that does not connect to the Internet, the default IP address may be used as long as no other device on the network uses the same address. If multiple WebRelay-Dual™ units are installed on the same network, each unit must have its own unique IP address. For example, WebRelay-Dual™ comes from the factory with a default IP address of 192.168.1.2. If multiple units are used, change the IP address for each unit (192.168.1.3, 192.168.1.4, 192.168.1.5 etc.).



2. Netmask:

This specifies the size of the local network. This must be obtained from the network administrator. By default, the netmask is set to 255.255.255.0.

3. Broadcast:

This specifies the broadcast address. This must be obtained from the network administrator. By default, this is set to 192.168.1.255.

4. Gateway:

This specifies the IP address of the gateway router. This must be obtained from the network administrator. By default, this is set to 192.168.1.1.

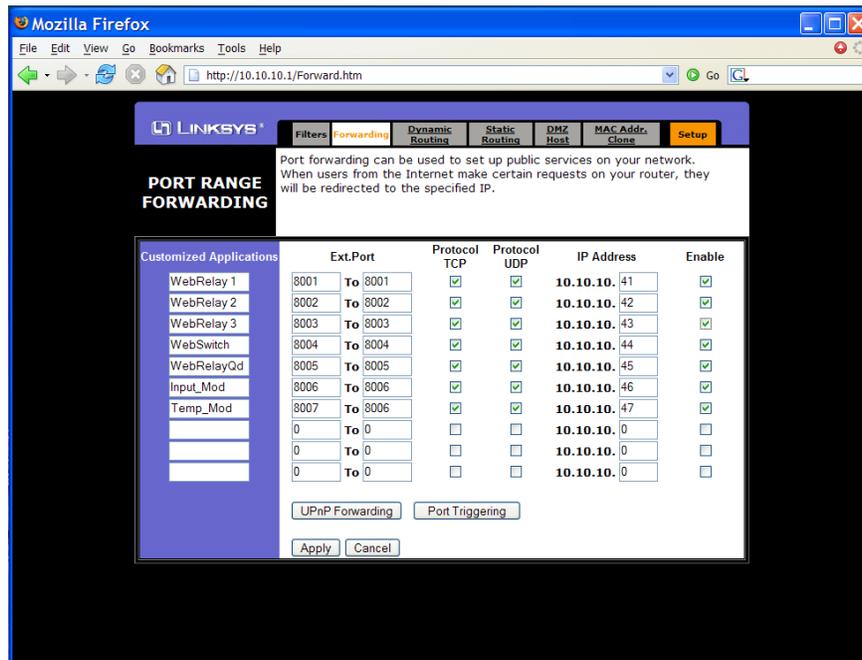
5. TCP Port:

This specifies the TCP port used for communications with WebRelay-Dual™. By default, the port is set to 80 which is the standard http port. It is recommended that the port not be changed without an understanding of TCP/IP and ports.

Changing the port can be useful for accessing multiple WebRelay-Dual™ devices which are installed behind a gateway router on a private network that uses non-routable IP addresses (192.168.x.x, 10.x.x.x, and 172.16.x.x through 172.31.x.x are non-routable or private IP addresses). In this case, each WebRelay-Dual™ unit would be assigned a different port (for example 8000, 8001, 8002, etc). The gateway router would be set up to

forward all traffic for each of the assigned ports to the IP address of the WebRelay-Dual™ unit which uses that port. The WebRelay-Dual™ units could then be accessed from outside the private network by entering the IP address of the gateway and the port for the desired WebRelay-Dual™ unit. Note that whenever any port is assigned other than port 80, all communications with that WebRelay-Dual™ device must include the port. For example, if WebRelay-Dual™ is assigned port 8000, access to the setup page would require the following URL to be entered; <http://192.168.1.2:8000/setup.html>.

An example screenshot of a gateway router configuration is given below. This setup allows seven ControlByWeb™ products to be accessed on a private network behind a gateway router. Note that this screenshot is simply an example of a typical router setup page. Routers will vary.



In the example, the seven ControlByWeb™ products are assigned IP addresses of 10.10.10.41 to 10.10.10.47. The first device labeled WebRelay-Dual 1 has an IP address of 10.10.10.41 and is assigned port 8001. Note that in the WebRelay-Dual™ setup page (under Network settings tab), TCP Port 8001 must be specified for this device. The second device, labeled WebRelay-Dual 2 has an IP address of 10.10.10.42 and is assigned port 8002. This WebRelay-Dual™ must have its TCP Port set to 8002 in its network settings page. This pattern continues through 10.10.10.77 which is assigned the port 8007. To access the ControlByWeb™ units from the Internet, enter the IP address of the gateway plus the port number of the desired device.

6. Modbus Port:

This specifies the port used for Modbus/TCP communications with WebRelay-Dual™. By default this is set to port 502 which is the standard Modbus port. For users not familiar with Modbus, Modbus is a messaging structure protocol used in industrial manufacturing control and automation. It is an open protocol and offers interoperability with software and devices from other manufacturers. If Modbus is not used, this setting can be ignored. Modbus communications are disabled whenever the control password is enabled.

7. Speed:

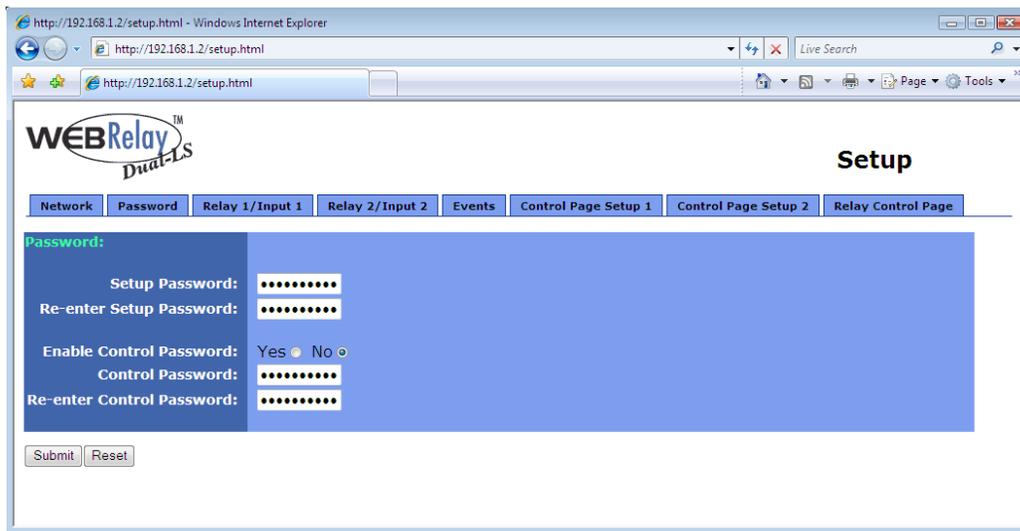
This option sets the data rate of the Ethernet port. Both 10Mbps and 100Mbps can be selected. The 100Mbps option offers faster communications but the amount of data to and from WebRelay-Dual™ is so small that users will not likely notice much of a difference. When WebRelay-Dual™ is set to 10Mbps, it draws less power and runs a little cooler which translate into a longer product life. IT IS RECOMMENDED THAT THIS SETTING BE LEFT AT 10MBPS UNLESS THE USER HAS A SPECIFIC REASON TO USE 100MBPS.

8. Mode:

This option allows the Ethernet port to be set to Half Duplex or Full Duplex. Legacy Ethernet operates in Half Duplex mode which means that devices can either send data or receive data, but not both at the same time. Full duplex means that devices can send and receive data at the same time.

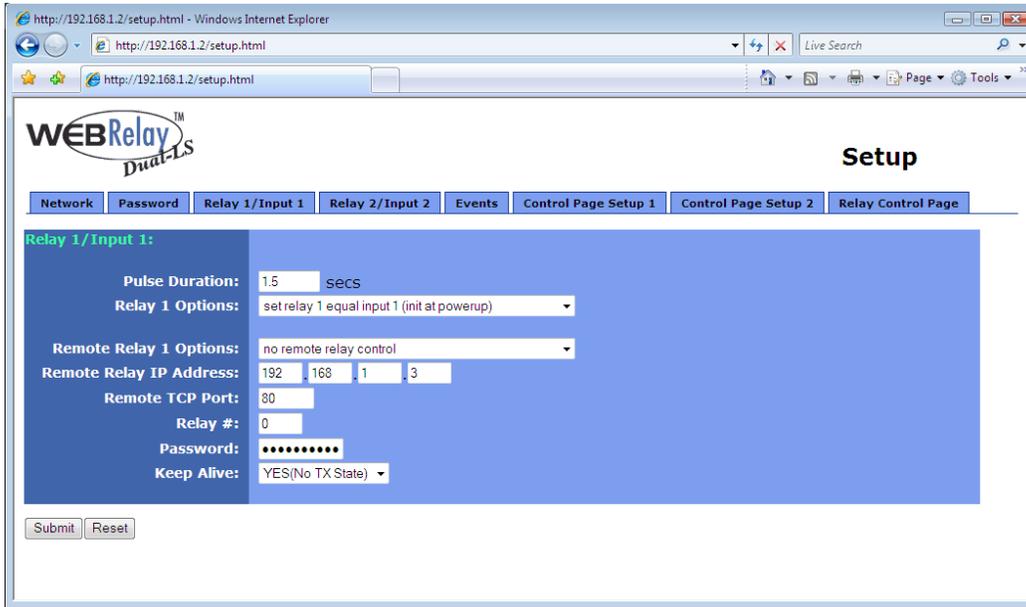
### 2.4.3 Password Setup Page

The password setup page is used to change and enable passwords. A password is required for the setup pages but is optional for the control page. The password is enabled or disabled for the control page by using the *Yes* or *No* radio buttons. Enabling the control page password also enables the requirement for a password when reading/writing XML pages. Enabling the control page password also disables the ability to communicate with WebRelay-Dual™ using Modbus. Note that when the password is changed, the password may be displayed in the browser's history in clear text. It is advisable to clear the browser's history after setting the password. Passwords may be up to 10 characters.



### 2.4.4 Relay/Input Setup Pages

These pages are used to set up the function of the relays and the control inputs. Note that in this manual, the relays are considered to be *on* when the relay coil is energized. When the relay coil is *on*, the normally open contacts are closed and the normally closed contacts are open. When the relay coil is *off*, the coil is not energized and the normally closed contacts are closed and the normally open contacts are open. Also note that in this manual the input is considered to be *on* when a voltage is applied to the input. The LEDs on the front panel of WebRelay-Dual™ will illuminate whenever the relay or input are in the *on* state.



#### 1. Pulse Duration:

When WebRelay-Dual™ receives a command (from over the network or through an input change) to *pulse* the relay, this is the time in seconds that the relay will pulse *on*. This time can be set from 0.1 seconds to 86,400 seconds (1 day). Note that each time WebRelay-Dual™ receives a pulse command, the relay will be turned *on* (if not *on* already) and the pulse timer will be reset. If WebRelay-Dual™ receives a pulse command and a second pulse command comes in before the pulse timer has expired, the relay will remain *on* and the pulse timer will be reset so the duration of the pulse will be extended. If pulse commands are issued continuously so that the pulse timer is not allowed to expire, the relay will remain *on* until after the last pulse command was sent (to be precise, the relay will go off at the time of the last command plus the Pulse Duration time). The pulse command can be canceled before the pulse duration time has expired by issuing any other command. The factory default setting for this is 1.5 seconds.

#### 2. Relay Options:

The optically-isolated input can be configured to control the relay. The ‘Relay Options’ setting is used to specify if and how the input affects the local relay. The options in the drop down list are described below. Note that in all cases, input changes will only affect the relay at the time the input is changed. Once the change has taken place the state of the input has no effect on the state of the relay. In other words, if the input causes the relay to go *on*, a user may turn the relay off from the web browser, even if the input is still *on*.

**-no local relay control-** The input has no effect on the local relay.

**-set relay equal input-** When a voltage is applied to the input, the relay goes *on*. When a voltage is removed from the input, the relay goes *off*.

**-set relay equal input (init at powerup)-** When a voltage is applied to the input, the relay goes *on*. When a voltage is removed from the input, the relay goes *off*. In addition, the relay will automatically be set to match the state of the input when WebRelay-Dual™ is powered on. This is the factory default setting.

**-set relay opposite of input-** When a voltage is applied to the input, the relay goes *off*. When a voltage is removed from the input, the relay goes *on*.

**-set relay opposite of input (init at powerup)-** When a voltage is applied to the input, the relay goes *off*. When a voltage is removed from the input, the relay goes *on*. In addition, the relay will automatically be set to the opposite of the state of the input when WebRelay-Dual™ is

- powered on.
- latch relay when input on (reset via web)-** When a voltage is applied to the input, the relay goes *on*. Once the relay is on it will stay *on* (unaffected by input changed) until it is turned *off* via the web (or command) by a user or WebRelay-Dual™ is powered off. This option is useful for alarming functions.
  - latch relay when input off (reset via web)-** When a voltage is removed from the input, the relay goes *on*. Once the relay is *on* it will stay *on* (unaffected by input changed) until it is turned off via the web (or command) by a user or WebRelay-Dual™ is powered off. This option is useful for alarming functions.
  - latch relay when input changes (reset via web)-** When a voltage is applied to the input or removed from the input, the relay goes *on*. Once the relay is *on* it will stay *on* (unaffected by input changes) until it is turned off via the web (or command) by a user or WebRelay-Dual™ is powered off. This option is useful for alarming functions.
  - toggle relay when input on-** When a voltage is applied to the input, the relay will change states. When a voltage is removed from the input, the relay will remain in its current state.
  - toggle relay when input off-** When a voltage is removed from the input, the relay will change states. When a voltage is applied to the input, the relay will remain in its current state.
  - toggle relay when input changes-** When a voltage is applied to the input or removed from the input the relay will change states.
  - pulse relay when input on-** When a voltage is applied to the input, the relay will pulse for the Pulse Duration specified. The voltage may remain on or be removed and the pulse time will not be affected. If the voltage is applied, removed, and then re-applied before the end of the pulse time, the Pulse Duration timer will be re-started (it will re-start each time a voltage is applied) and the relay will continue to remain *on* until the timer expires.
  - pulse relay when input off-** When a voltage is removed from the input, the relay will pulse for the Pulse Duration specified. The voltage may remain off or be re-applied and the pulse time will not be affected. If the voltage is removed, re-applied, and then removed again before the end of the pulse time, the Pulse Duration timer will be re-started (it will re-start each time a voltage is removed) and the relay will continue to remain *on* until the timer expires.
  - pulse relay when input changes-** When a voltage is applied to or removed from the input, the relay will pulse for the Pulse Duration specified. If the state of the voltage changes before the end of the pulse, the timer will be re-started (it will re-start each time the voltage changes) and the relay will continue to remain *on* until the timer expires.

### 3. Remote Relay Options:

The optically-isolated inputs can be configured to control a WebRelay-Dual™ device (as well as other ControlByWeb™ products) at a remote location. The 'Remote Relay Options' setting is used to specify if and how the input affects the remote relay. The options in the drop down list are described below. Note that in all cases, input changes will only affect the remote relay at the time the input is changed. Once the change has taken place the state of the input has no effect on the state of the remote relay. In other words, if an input causes the remote relay to go on, a user may turn the remote relay off from the web browser, even if the local input is still on.

- no remote relay control-** The input has no effect on the remote relay.
- remote command equals input-** When a voltage is applied to the input, a command is sent to the remote WebRelay-Dual™ to set the relay state to *on*. When a voltage is removed from the input, a command is sent to the remote WebRelay-Dual™ to set the relay state to *off*.
- remote command opposite of input-** When a voltage is applied to the input, a command is sent to the remote WebRelay-Dual™ to set the relay state to *off*. When a voltage is removed from the input, a command is sent to the remote WebRelay-Dual™ to set the relay state to *on*.
- send on command when input on (no off command)-** When a voltage is applied to the input, a command is sent to the remote WebRelay-Dual™ to set the relay state to *on*. When a voltage is removed from the input, no commands are sent to the remote WebRelay-Dual™.
- send on command when input off (no off command)-** When a voltage is removed from the input,

- a command is sent to the remote WebRelay-Dual™ to set the relay state to *on*. When a voltage is applied to the input, no commands are sent to the remote WebRelay-Dual™.
- send off command when input off (no on command)-** When a voltage is removed from the input, a command is sent to the remote WebRelay-Dual™ to set the relay state to *off*. When a voltage is applied to the input, no commands are sent to the remote WebRelay-Dual™.
- send off command when input on (no on command)-** When a voltage is applied to the input, a command is sent to the remote WebRelay-Dual™ to set the relay state to *off*. When a voltage is removed from the input, no commands are sent to the remote WebRelay-Dual™.
- send pulse command when input on-** When a voltage is applied to the input, a command is sent to the remote WebRelay-Dual™ to pulse the relay. When a voltage is removed from the input, no commands are sent.
- send pulse command when input off-** When a voltage is removed from the input, a command is sent to the remote WebRelay-Dual™ to pulse the relay. When a voltage is applied to the input, no commands are sent.
- send pulse command when input changes-** When a voltage is applied to the input or removed from the input, a command is sent to the remote WebRelay-Dual™ to pulse the relay.
- send toggle command when input on-** When a voltage is applied to the input, a command is sent to the remote WebRelay-Dual™ to toggle the relay (change the remote relay to the state opposite of its current state). When a voltage is removed from the input, no commands are sent.
- send toggle command when input off-** When a voltage is removed from the input, a command is sent to the remote WebRelay-Dual™ to toggle the relay (change the remote relay to the state opposite of its current state). When a voltage is applied to the input, no commands are sent.
- send toggle command when input changes-** When a voltage is applied to the input or removed from the input, a command is sent to the remote WebRelay-Dual™ to toggle the relay (change the remote relay to the state opposite of its current state).
- remote command equals local relay-** This option is used to “daisy-chain” ControlByWeb™ units. In this mode, commands are sent to the remote WebRelay™ based upon the state of the local relay and not the local input. When the local relay changes state to *on*, a command is sent to the remote WebRelay™ to set the relay state to *on*. When the local relay changes state to *off*, a command is sent to the remote WebRelay™ to set the relay state to *off*.

#### 4. Remote Relay IP Address:

When any option in the ‘Remote Relay Options’ setting is selected other than ***no remote relay control***, commands are sent over the IP network to this address as the control voltage input changes.

#### 5. Remote TCP Port:

When any option in the ‘Remote Relay Options’ setting is selected other than ***no remote relay control***, commands are sent over the IP network to this TCP port on the remote WebRelay-Dual™ as the control voltage input changes.

#### 6. Relay #:

When any option in the ‘Remote Relay Options’ setting is selected other than ***no remote relay control***, commands are sent to the remote device with this relay offset. This option is used to identify the specific relay within the device that will be controlled. If the remote device is a single WebRelay™, this option should be set to 0. If the remote device has multiple relays (such as WebRelay-Dual™ and WebRelay-Quad™) this option should be set to the relay number that will be controlled (1 or 2 for WebRelay-Dual™ and 1,2,3 or 4 for WebRelay-Quad™).

For a more technical description of this option, commands sent to remote WebRelay™ units look like this...

<http://192.168.1.2/state.xml?relayState=1>

When a 0 is entered into this field, the above format is used which is required for standard WebRelay™ units.

Commands sent to a remote WebRelay-Dual™ or WebRelay-Quad™ look like this...

http://192.168.1.2/state.xml?relay1State=1

http://192.168.1.2/state.xml?relay2State=1

...

http://192.168.1.2/state.xml?relay4State=1

In this case, each of the four relays can be addressed as illustrated. The address of the specific relay on the WebRelay-Quad™ is given in this field.

For custom computer applications, the ‘Relay #’ option may be useful to monitor input changes as they happen. This works by pointing the ‘Remote Relay IP Address’ above, to a computer instead of another WebRelay-Dual™ device. Each time the input changes, a message is sent to the computer indicating the change. The computer can identify the device sending the message by the ‘Relay #’. Note that when remote relay commands are sent to a computer instead of a WebRelay-Dual™, the computer must appear to this WebRelay-Dual™ as another WebRelay-Dual™ device which may require some advanced programming.

This field may be set from 0 to 255.

#### 7. Password:

When any option in the ‘Remote Relay Options’ setting is selected other than **no remote relay control**, this password is included with commands that are sent to the remote WebRelay-Dual™ unit. If the remote WebRelay-Dual™ unit has the control password option enabled, the password for that unit must be entered here for the remote commands to have any effect.

#### 8. Keep Alive:

When any option in the ‘Remote Relay Options’ setting is selected other than **no remote relay control**, the Keep Alive option may be turned on to keep the TCP connection to the remote WebRelay-Dual™ unit alive. If the connection is alive, commands sent to the remote unit will change the state of the remote relay almost instantly (with a good network connection). If the connection is not alive, commands sent to the remote unit will be delayed a few seconds while the connection is being setup. There are three options available in this field.

I. **NO**: When this option is selected, the TCP connection between the two units will be terminated if there is a time gap (more than about 50 seconds) between commands. Once the connection is closed, the two WebRelay-Dual™ units must re-establish a connection before the next command can be sent. This means that when the input voltage toggles, it may take a few seconds for the remote relay to respond. In many cases, this delay may not be a problem. This option uses the least network bandwidth (and is the most secure) because communications only take place when the input voltage changes..

II. **YES(No TX State)**: When this option is selected, a message is sent to the remote unit every 50 seconds to keep the TCP connection alive. The message does not contain the state of the local input. If power is lost and the control voltage changes while the unit is not powered, the new control voltage status will not be sent to the remote unit when the power is restored. The state will not be updated until the next time the control input changes.

III. **YES(TX State)**: When this option is selected, a message is sent to the remote unit every 50 seconds to keep the TCP connection alive. This message will contain the current state of the input. By sending the state of the unit, changes to the control voltage input will eventually be sent to the remote unit, even if the change occurred during a power loss.

Transmit state is only valid for the ‘Remote Relay Options’ *remote command equals input*, *remote command opposite of input* and *remote command equals local relay*. If any of the other ‘Remote Relay Options’ are selected, the state will not be transmitted.

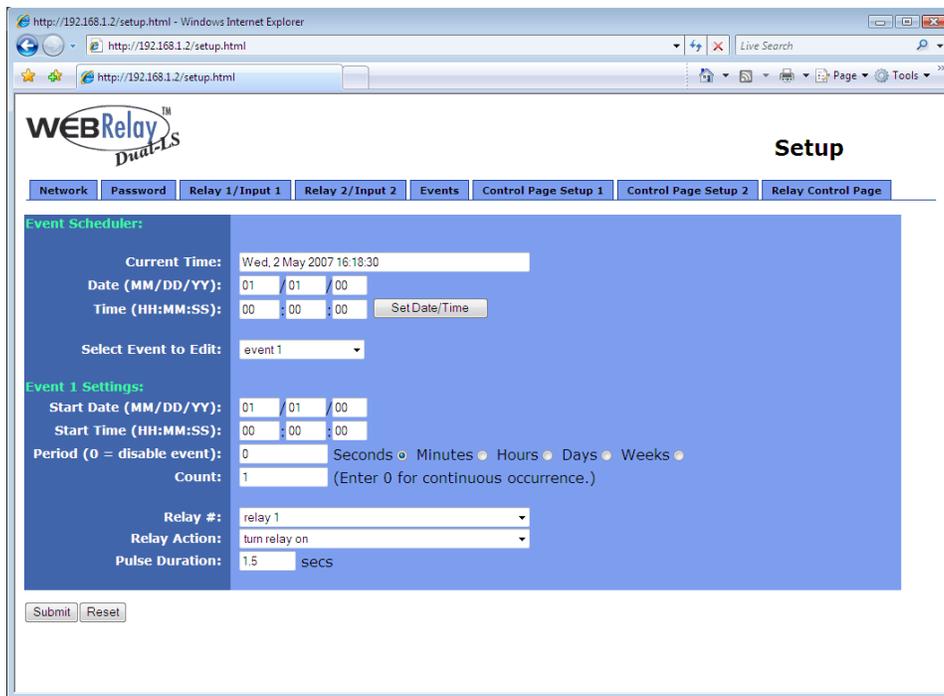
Note that when units are set up to require a control password, this password must be sent each time the state is updated. When the option is selected to transmit the state of the input each time the keep alive message is sent, the password will be frequently transmitted over the network. This may not be a problem, but the user should be aware that the more often the password is sent, the more opportunity hackers would have to capture the password, decode it, and potentially use it to control the remote relay (especially in public networks such as the Internet).

### 2.4.5 Events

In addition to controlling the relays remotely over the network or by using a control voltage, events can be programmed to turn the relays on and off at specific times. Up to six events can be created. Each event can occur one time or multiple times.

#### 1. Current Time:

WebRelay-Dual™ has an internal real-time clock. This field displays the day, date, and time of day that is set in the internal clock. Time is displayed in 24-hour mode. The date is automatically adjusted for leap years but the time will not automatically change for daylight savings time changes.



#### 2. Date:

This field is used to set the current date in the internal clock. To set the clock, enter the date in the date field, the time in the time field, and press Set Date/Time.

#### 3. Time:

This field is used to set the current time in the internal clock. To set the clock, enter the date in the date field, the time in the time field, and press Set Date/Time.

4. Event:

This drop-down box is used to select the event to view and/or edit. Events 1-6 can be selected. The seven fields below display settings for the selected event.

5. Start Date (MM/DD/YY):

This field specifies the date that this event will take place (or begins).

6. Start Time (HH:MM:SS):

This field specifies the time that this event will take place (or begins). Time is entered as 24 hour time.

7. Period (0=disable event):

When the event is configured to occur more than one time (any entry other than **1** in the Count field below), the Period field specifies the time between events. For long events (pulse relay), this is the time interval between the time the event begins and the next time the event begins. This time can be specified in Seconds, Minutes, Hours, Days, or Weeks by selecting the appropriate radio button to the right of this field. If 0 is entered in this field, the event will be disabled.

8. Count:

This field specifies the number of times the event will occur. If **0** is entered, the event will be continuous.

9. Relay #:

This field specifies which relay will be affected by the event. This can be set to **relay 1** or **relay 2**.

10. Relay Action:

This drop down box specifies which action will take place when the event occurs. This can be set to **turn relay on**, **turn relay off**, **pulse relay**, or **toggle relay**.

11. Pulse Duration:

When events are scheduled to occur, and the Relay Action field above is set to **pulse relay**, this field specifies the time (in seconds) that the relay will be pulsed. This can be set from 0.1 seconds to 86,400 seconds (1 day).

### 2.4.6 Control Page Setup

The following three parameters only appear on Control Page Setup 1

1. Main Header Text:

This text will be displayed in the main header area of the control page. This field can be up to 25 characters in length.

2. Auto Refresh Page:

Web pages traditionally display static information. The WebRelay-Dual™ control page, however, displays information that is dynamic. Each time the control page is loaded to the browser, it displays a snapshot of the current status of the unit. If the state of the unit changes, the information on the control page will be outdated unless the page is re-loaded. Whenever a user changes the state of a relay via the web browser, the web page will automatically reload the control page to display current information. If, however, the relay state changes due to a change in the control voltage input or due to a command sent from another machine, the relay state will not be updated and the control page information will be obsolete. In addition, when the optically-isolated inputs change, the control page will be obsolete until the page is refreshed. WebRelay-Dual™ can not force the web page to be updated when the input changes. The 'Auto Refresh Page' option will cause the control page to continually update its contents by setting a timer in the web page that causes it to be reloaded at a specified time interval.

3. Duration:

If the 'Auto Refresh Page' option is set to **Yes**, this field specifies the time interval in seconds that the page

---

will be refreshed. It can be set from 1 to 32 seconds.

*The following parameters appear on both Control Setup Page 1 and Control Setup Page 2*

#### 4. Relay Description:

This text is used to describe the function of the relay on the control page. This text will not appear if the 'Display Relay Status' is set to *No*, the 'On/Off Buttons' is set to *No*, and the 'Pulse Button' is set to *No*. This field can be up to 20 characters in length.

#### 5. Display Relay Status:

When this option is set to *Yes*, the relay status will be displayed on the control page. If it is set to *No*, the relay status will not be displayed.

#### 6. Status ON Color:

When the 'Display Relay Status' option is set to *Yes*, this field specifies the color that will be displayed when the relay is *on* (coil energized). Options are Green, Red, Yellow, and Blue.

#### 7. Status ON Text:

When the 'Display Relay Status' option is set to *Yes*, this field specifies the text that will be displayed when the relay is *on* (coil energized).

#### 8. Status OFF Color:

When the 'Display Relay Status' option is set to *Yes*, this field specifies the color that will be displayed

when the relay is *off* (coil not energized).

9. Status OFF Text:

When the 'Display Relay Status' option is set to **Yes**, this field specifies the text that will be displayed when the relay is *off* (coil not energized).

10. ON/OFF Button:

When this option is set to **Yes** the ON/OFF button to control the relay will be displayed on the control page. This allows the user to toggle the relay state. When this option is set to **No** the ON/OFF button to control the relay will not be displayed on the control page.

11. ON/OFF Button Label:

When the 'ON/OFF Button' option is set to **Yes**, this field specifies the text that will be displayed in the button.

12. Pulse Button:

This option allows the user to include a pulse button to control the relay. If the pulse button is included, the relay will *pulse* for the time specified in the 'Pulse Duration' field in the 'Relay/Input' page each time the button is pressed.

13. Pulse Button Label:

When the 'Pulse Button' option is set to **Yes**, the text entered here will appear in the pulse button on the control page.

14. Display Input Status:

When this option is set to **Yes** the status of the optically-isolated input will be displayed on the control page. Note that this input can be used to control a local relay, a remote relay, or be read using the XML page no matter how this option is set.

15. Input Description:

This is the text that will be displayed next to the input status when the 'Display Input Status' is enabled. It can be up to 20 characters in length.

16. Input ON Color:

When 'Display Input Status' is set to **Yes**, this field specifies the color that will be displayed when the input is *on* (voltage applied to input).

17. Input ON Text:

When 'Display Input Status' is set to **Yes**, this field specifies the text that will be displayed when the input is *on* (voltage applied to input).

18. Input OFF Color:

When 'Display Input Status' is set to **Yes**, this field specifies the color that will be displayed when the input is *off* (voltage is not applied to input).

19. Input OFF Text:

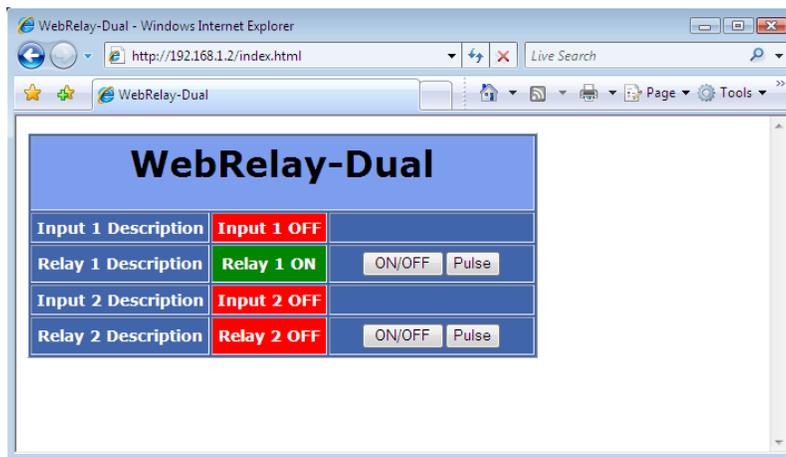
When 'Display Input Status' is set to **Yes**, this field specifies the text that will be displayed when the input is *off* (voltage is not applied to input).

## Section 3: Operation

WebRelay-Dual™ can be operated by applying a voltage to the optically-isolated input, using a web browser, by sending text commands to an XML status/control page, or by sending Modbus/TCP requests. Using a control voltage was described throughout previous sections and will not be described here.

### 3.1 Browser Operation

Once the unit is set up, the control page may be accessed by typing the following URL into the web browser: <http://192.168.1.2> (Note that if the IP address was changed, replace the default IP address shown with the new address that was assigned. Note also that if any port is used other than port 80, the port must also be included in the request. For example, to access a unit on port 8000 the following command would be used: <http://192.168.1.2:8000> ) The new control page will appear. A control page with default settings is shown below.



### 3.2 XML Operation

Custom computer applications may be created to monitor and control WebRelay-Dual™ without using a web browser. There are three xml pages that can be used to monitor and control WebRelay-Dual™. The three xml pages are `state.xml`, `eventX.xml`, and `rtc_cal.xml`.

#### 3.2.1 Controlling The Unit Through `state.xml`

The state of the inputs and relays can be monitored by sending a request to port 80 (or port specified in setup) for the XML page `state.xml`. The relays can be controlled by sending GET requests to the same page on port 80 (or port specified in setup). This can be demonstrated by entering commands into the URL line of a web browser.

Request the current state: <http://192.168.1.2/state.xml>

This will return the following XML page:

```
<?xml version="1.0" encoding="utf-8" ?>
- <datavalues>
  <input1state>0</input1state>
  <input2state>0</input2state>
  <relay1state>0</relay1state>
  <relay2state>0</relay2state>
  <input1hightime>0.00</input1hightime>
</datavalues>
```

The tags <input1state>, <input2state>, <relay1state>, <relay2state>, and <input1hightime> indicate the current state of the WebRelay-Dual™ unit. Values for each tag are described below.

<input1state>	0=off (voltage not applied to input one) 1=on (voltage applied to input one)
<input2state>	0=off (voltage not applied to input two) 1=on (voltage applied to input two)
<relay1state>	0=off(coil one off) 1=on (coil one energized)
<relay2state>	0=off(coil two off) 1=on (coil two energized)
<input1hightime>	This field indicates the amount of time that has elapsed, in seconds, since a voltage has been applied to input one. When the voltage is removed the counter stops. When a voltage is reapplied to input one the counter resets and begins counting again. This counter is also cleared whenever WebRelay-Dual™ loses power.

Commands can be sent to WebRelay-Dual™ which control the relay and change its state. Commands are sent using a variable called relayXState (X is replaced by 1 for relay1 or 2 for relay2). A few examples of using relayXState are given here.

Turn relay one 'off': <http://192.168.1.2/state.xml?relay1State=0>  
 Turn relay one 'on': <http://192.168.1.2/state.xml?relay1State=1>  
 Turn relay two 'off': <http://192.168.1.2/state.xml?relay2State=0>  
 Turn relay two 'on': <http://192.168.1.2/state.xml?relay2State=1>

All relayXState options are contained in the following list.

relayXState	0=Turn relay off 1=Turn relay on 2=Pulse relay 5=Toggle the state of the relay
-------------	---

When the pulse command is sent to WebRelay-Dual™, the relay will pulse for the time specified in the setup page. It is possible, however, to specify a pulse time that is different than the pulse time configured in the setup page. This is done by sending an additional variable called pulseTime along with the desired pulse time.

The pulseTime variable does not change the pulse time specified in the setup page and it is not stored or recorded. The pulseTime variable only changes the pulse duration for the single pulse initiated by that command. In other words, you must issue the pulseTime for each pulse command that differs from the preset pulse time. For example, suppose the pulse time is set to 1.5 seconds in the configuration pages. To issue one 1.5 second pulse, three five second pulses, and a 1.5 second pulse once again to relay one, you would issue the following commands...

<a href="http://state.xml?relay1State=2">state.xml?relay1State=2</a>	This will pulse relay one for the preset time (1.5 seconds)
<a href="http://state.xml?relay1State=2&amp;pulseTime1=5">state.xml?relay1State=2&amp;pulseTime1=5</a>	This will pulse relay one for 5 seconds.
<a href="http://state.xml?relay2State=2">state.xml?relay2State=2</a>	This will pulse relay two for the preset time (1.5 seconds)
<a href="http://state.xml?relay2State=2&amp;pulseTime2=5">state.xml?relay2State=2&amp;pulseTime2=5</a>	This will pulse relay two for 5 seconds

(Note that `http://192.168.1.2/` would be included on all commands above)

Note that when commands are sent to WebRelay-Dual™, its current state is returned in the form of an XML page. The commands can also be sent without having WebRelay-Dual™ return the XML page. This is accomplished by adding the `noReply` field as follows.

Turn relay one 'on' without returning state: `http://192.168.1.2/state.xml?relay1State=1&noReply=1`

Turn relay one 'off' without returning state: `http://192.168.1.2/state.xml?relay1State=0&noReply=1`

Password :

If the control password is enabled in the WebRelay-Dual™ unit and the `state.xml` page is requested through a browser, the browser will prompt the user for the password. If the XML request is sent from another application and not a browser, the html request will need to contain the password encoded using the base 64 encoding scheme. The html request header without the password looks like this:

```
GET /state.xml?relay1State=1&noReply=1 HTTP/1.1 (Ends with two \r\n)
```

The html request header with the password looks like this:

```
GET /state.xml?relay1State=1&noReply=1 HTTP/1.1(\r\n here)
Authorization: Basic bm9uZTp3ZWJyZWxheQ== (Ends with two \r\n)
```

where `bm9uZTp3ZWJyZWxheQ==` is the base 64 encoded version of the user name and password  
none:webrelay

**A utility is provided at <http://www.ControlByWeb.com/encoder> that can be used to encode the password. The utility is used by simply typing the string `username:password` into the website and pressing `encode`.**

### 3.2.2 Monitoring Events Through `eventX.xml`

The relays can be configured to turn on, turn off, or pulse at specified times. Instructions to change the relay state at a given time (or periodically) are called events. WebRelay-Dual™ supports six events and these events can be monitored through an XML page called `eventX.xml` (X is the event number 1-6). This can be demonstrated by entering commands into the URL line of a web browser.

Request the current state of event one: `http://192.168.1.2/event1.xml`

This will return the following XML page:

```
<?xml version="1.0" encoding="utf-8" ?>
  - <event1>
      <active>yes</active>
      <currentTime>05/07/07 08:20:20</currentTime>
      <nextEvent>05/07/07 08:20:30</nextEvent>
      <period>10 s</period>
      <count>0</count>
      <relay>1</relay>
      <action>on</action>
      <pulseDuration>1.5 s</pulseDuration>
    </event1>
```

For each event, the following tags are displayed in the XML page: `<eventX>`, `<active>`, `<currentTime>`, `<nextEvent>`, `<period>`, `<count>`, `<relay>`, `<action>`, and `<pulseDuration>`. Each of the tags are described below.

---

<eventX>	All of the XML tags for each event are encapsulated between opening and closing tags called eventX. The X in the tag represents the event number (1-6).
<active>	This field indicates whether or not the event is active. The text <i>yes</i> indicates that the event is active and the event will occur at the next event time (given that the next event time is in the future). The text <i>no</i> indicates that the event is inactive and event will not take place.
<currentTime>	This field indicates the current date and time of the unit. The date and time are given as <b>MM/DD/YY HH:MM:SS</b> and is in 24 hour format.
<nextEvent>	This field indicates the next time the event is scheduled to occur. The format for the date and time is <b>MM/DD/YY HH:MM:SS</b> . Note that if the event is not active, this field will display the start date and time as entered in the setup pages for the event.
<period>	For events that occur more than one time, this field indicates the period of the event (time between event occurrences). The unit of time is indicated after the value (seconds (s), minutes (m), hours (h), days (d), or weeks (w)). If this field is set to 0 then the event has been disabled.
<count>	This field indicates the number of remaining times the event will occur. If this field equals zero and the event is active then the event will occur continuously forever.
<relay>	1 = event action will apply to relay number one 2 = event action will apply to relay number two
<action>	on (relay will turn on when event occurs) off (relay will turn off when event occurs) pulse (relay will pulse when event occurs) toggle (relay will toggle state when event occurs)
<pulseDuration>	When the event causes the relay to pulse, this field indicates the time (in seconds) that the relay will pulse.

### 3.2.3 Calibrating The Real Time Clock Through rtc\_cal.xml

The real time clock in WebRelay-Dual™ is a crystal based time keeping circuit which offers enough accuracy for many applications. If the unit is going to be used in extreme temperatures, the real time clock can be calibrated in order to obtain more accuracy at those temperatures. Changing the calibration settings of the real time clock can be done through a request to the XML page rtc\_cal.xml.

Request the current calibration byte of the real time clock: [http://192.168.1.2/rtc\\_cal.xml](http://192.168.1.2/rtc_cal.xml)

This will return the following XML page:

```
<?xml version="1.0" encoding="utf-8" ?>
- <datavalues>
  <cal_val>6</cal_val>
</datavalues>
```

The current calibration byte is given between the <cal\_val> tags. To determine what the calibration byte should be for a specific environment, set the clock in the Events setup page to the correct time using an accurate time source. After 30 days check the time of the module again in the Events setup page. Find the difference in seconds between the clock and the module. This value can be used to determine the calibration byte using the following two equations. If the difference is a positive value use:

$$N = \text{INT}((S*32)/337)$$

where N equals the calibration byte, S equals the number of seconds difference between the clock and the unit, and INT indicates that the outcome should be rounded to a whole number. If the difference is a negative value use:

$$N = \text{INT}((S*32)/169)$$

Once a calibration byte has been determined it can then be set using the following xml command:

```
rtc_cal.xml?cal_val=N
```

where N is the calibration value determined above.

Example: Using the above method it was determined that the difference between an accurate clock and WebRelay-Dual™ units time after a 30 day period was a negative 30 seconds. The WebRelay-Dual™ units time had increased by 30 seconds compared to the clock over a 30 day period. Using the equation for a negative difference we come up with a calibration byte of 6. Six is the calibration value that will cause WebRelay-Dual™ real time clock to slow down enough to make it accurate relative to our comparison clock. This value can then be used to calibrate the unit using the following request:

```
rtc_cal.xml?cal_val=6
```

(Note that <http://192.168.1.2/> or the current IP address, would be included on all commands above)

Password :

Unlike the other XML pages, rtc\_cal.xml deals with a setup option of the WebRelay-Dual™ unit and therefore uses the same password as the setup pages. If rtc\_cal.xml is requested through a browser, the browser will prompt the user for the password. If the XML request is sent from another application and not a browser, the html request will need to contain the password encoded using the base 64 encoding scheme. The html request header without the password looks like this:

```
GET /rtc_cal.xml?cal_val=3 HTTP/1.1 (Ends with two \r\n)
```

The html request header with the password looks like this:

```
GET /rtc_cal.xml?cal_val=3 HTTP/1.1(\r\n here)
Authorization: Basic bm9uZTp3ZWJyZWxheQ== (Ends with two \r\n)
```

where bm9uZTp3ZWJyZWxheQ== is the base 64 encoded version of the username and password  
none:webrelay

**A utility is provided at <http://www.ControlByWeb.com/encoder> that can be used to encode the password. The utility is used by simply typing the string username:password into the website and pressing encode.**

### 3.3 Configuring Events Through Get Requests

Custom computer applications may be created to change event settings without using a web browser and the default setup pages. There are two server side scripts that can be sent HTTP GET requests in order to change event settings: eventRef.srv, and event.srv. Requesting one of these server side scripts can be done in the same way the control page is requested. The only difference is that event parameters can be sent to the unit through the request as well. This means that these requests can be accomplished by directly typing them into the address bar of a web browser, or by sending them through a custom computer application.

### 3.3.1 Specifying the Event to be Modified

There are six individual events that can be modified on the WebRelay-Dual. Before modifying any of the settings for an event, that event needs to be selected for modification. In order to select an event for modification the eventRef.srv script should be called, and the event number should be passed to the script. For example, to select event number five for modification we would issue the following request:

```
http://192.168.1.2/eventRef.srv?eventNumber=4
```

In the example above, notice that we have specified the number four for the eventNumber parameter even though we wanted to select event number five for modification. WebRelay-Dual numbers events starting with zero on up to five. So, event five will be event four and so on when selecting the event. Also note that when the request is sent, we get the events setup page as a response. In order to stop WebRelay-Dual from sending back the events setup page as a response we can append noReply=1 to the end of the request as follows:

```
http://192.168.1.2/eventRef.srv?eventNumber=4&noReply=1
```

When a request is sent using the noReply=1 parameter and value pair only acknowledges will be received by the request. It is up to the client to close the connection with the server after the request is sent.

Once an event has been selected for modification, this event will remain selected until a new event has been specified. During this time all other event parameters that are changed will pertain to the selected event. Changing the event for modification can also be achieved through the Events setup page by selecting an event from the drop down box for the field **Select Event to Edit**.

### 3.3.2 Modifying the Event Settings

Once an event has been selected for modification, we can send GET requests to the event.srv script to change the event settings. There are eight settings that can be modified for each of the six events on WebRelay-Dual: start date, start time, period, units, count, relay number, relay action, and pulse duration. Following is a description of each setting and how it can be changed through the use of GET requests to the event.srv script. Normally all of these parameters can be sent in one request even though the examples show individual parameters being changed only. Also, the response of a request to the event.srv script will normally return the main setup page. To avoid getting this response, responses can be inhibited by sending a noReply=1 parameter and value pair. Appended to the end of the request using the & sign. If this method is used, the client application is responsible for closing the connection with the server.

**Start Date:** The start date is defined by the month, day, and year in which the event will occur for the first time. To specify these values using a get request use the parameters smonth, sday, and syear. For example, to change the start date of an event to January 5, 2010 send the following get request to event.srv:

```
http://192.168.1.2/event.srv?smonth=01&sday=05&syar=10
```

**Start Time:** The start time is defined by the hour, minute, and second in which the event will occur for the first time. To specify these values using a get request use the parameters shour, sminute, and ssecond. For example, to change the start time of an event to 1:15 and 30 seconds in the afternoon send the following get request to event.srv:

```
http://192.168.1.2/event.srv?shour=13&sminute=15&ssecond=30
```

Note: The hour is specified in 24 hour time.

**Period and Units:** The period and units settings determines the time period between the subsequent occurrences of an event. If the period setting is set to zero the event will be disabled. These fields can be changed using the parameters freq and units. For example, to set the period of the event to 20 seconds the following GET request could be issued:

`http://192.168.1.2/event.srv?freq=20&units=0`

**Note:** In order to specify the units for the period use the following lookup table to determine the correct value to send for the units parameter.

Units	Value
Seconds	0
Minutes	1
Hours	2
Days	3
Weeks	4

**Count:** This setting determines how many times the event will occur. If this setting is set to zero, the event will occur forever at time intervals specified by the period and units settings. This field can be changed using the parameter count. For example, to create an event that will occur 20 times issue the following GET request:

`http://192.168.1.2/event.srv?count=20`

**Relay Number:** Each event can be setup to control one of the two relays on the WebRelay-Dual. To specify which relay the event will control use the whichRelay parameter. For example, to create an event that performs an action on relay number two send the following GET request:

`http://192.168.1.2/event.srv?whichRelay=1`

**Note:** In the example above notice that we have specified the number one for the whichRelay parameter even though we wanted to select relay number two. WebRelay-Dual numbers relays starting with zero on up to one. So, relay one will be relay zero and so on when selecting relays.

**Relay Action:** Each event can be setup to perform a different action upon execution. To specify which action will occur when the event executes use the relayAction parameter. For example, to setup an event which will turn a relay on, send the following GET request:

`http://192.168.1.2/event.srv?relayAction=0`

**Note:** In order to specify the relay action use the following lookup table to determine the correct value to send for the relayAction parameter.

Relay Action	Value
Turn Relay On	0
Turn Relay Off	1
Pulse Relay	2
Toggle Relay	3

**Pulse Duration:** When the action of an event is set to pulse the relay, the amount of time the relay pulses is determined by the pulse duration. To specify the pulse duration for an event use the eventPulseDur paramter. For example, to setup an event which will pulse a relay for 10 seconds issue the following GET request.

```
http://192.168.1.2/event.srv?eventPulseDur=10
```

Note: The pulse duration for events is always specified in seconds. The valid range for pulse field is 0.1 to 86400 seconds.

#### **Password :**

If eventRef.srv or event.srv is requested through a browser, the browser will prompt the user for the password used in the setup pages. If the request is sent from another application and not a browser, the GET request will need to contain the setup password encoded using the base 64 encoding scheme. The HTTP request header without the password looks like this:

```
GET /eventRef.srv?eventNumber=1 HTTP/1.1 (Ends with two \r\n)
```

The GET request header with the password looks like this:

```
GET / eventRef.srv?eventNumber=1 HTTP/1.1(\r\n here)
Authorization: Basic bm9uZTp3ZWJyZWxheQ== (Ends with two \r\n)
```

where bm9uZTp3ZWJyZWxheQ== is the base 64 encoded version of the username and password  
none:webrelay

**A utility is provided at <http://www.ControlByWeb.com/encoder> that can be used to encode the password. The utility is used by simply typing the string username:password into the website and pressing encode.**

### **3.4 Modbus Operation**

WebRelay-Dual™ can be controlled (and read) using Modbus/TCP protocol. This provides a standard means of using WebRelay-Dual™ in conjunction with devices and software from other manufacturers. This section contains the information necessary to communicate with WebRelay-Dual™ using Modbus/TCP. This 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>.

It is important to note that when the control password in WebRelay-Dual™ is enabled, Modbus/TCP communications are disabled. This is because Modbus/TCP does not provide a mechanism for password protection. Make sure the control password is disabled (default) before using Modbus with WebRelay-Dual™.

WebRelay-Dual™ functions as a Modbus server (slave). Client (master) devices open a connection with WebRelay-Dual™ on port 502 (unless another modbus port is selected) and sends commands or requests to read the state of the relay, read the input, or change the relay state. When WebRelay-Dual™ receives a command, it will perform the desired function and return a response. The following commands are available:

Read Coils (Modbus function 01) - read the status of the relays (and/or the inputs).  
 Read Discrete Inputs (Modbus function 02) - read the status of the inputs  
 Read Holding Registers (Modbus function 03) – read back how long input one remained high (voltage applied most recently)  
 Write Single Coil (Modbus function 05) - change the relay state  
 Write Multiple Coils (Modbus function 15) – change one or more relay states  
 Write Multiple Registers (Modbus function 16) - pulse one or more relays

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. WebRelay-Dual™ has two TCP sockets available for Modbus/TCP which allows two connections to be open at one time. Requests for more than two open connections will be rejected.

#### 3.4.1 Read Coils (Modbus Function Code 01 (0x01))

This function returns the state of the relay coils. In addition, the optically isolated inputs can be read using this function which allows full state of WebRelay-Dual™ to be read in a single transaction.

Valid Starting Address and Quantity of coils combinations

Valid starting address can fall in the range of 0x0000 and 0x0004. Valid quantities can also fall in the range of 0x0000 and 0x0004. The only restriction is that the sum of the starting address and the quantity of coils must not be greater than 0x0004. For example, if you want to read 0x0004 bits, two coils and two inputs, then the starting address must be 0x0004. If you want to read the inputs only you may read 0x0002 bits and use the starting address of 0x0002. On the other hand, you may want to read the second relay state only. This can be achieved by starting at address 0x0001 and using a quantity of 0x0001.

##### Request

Modbus/TCP:

Transaction identifier (2 Bytes): 0x0001  
 Protocol identifier (2 Bytes): 0x0000  
 Length (2 Bytes): 0x0006  
 Unit identifier (1 Byte): 0xff

Modbus:

Function code (1 Byte): 0x01 (read coils)  
 Starting address (2 Bytes): 0x0000 (0x0000 to 0x0004)  
 Quantity of coils (2 Bytes): 0x0004 (0x0001 to 0x0004)

Character String Example:

```
char read_coils_mb_request[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x06, 0xff, 0x01, 0x00, 0x00, 0x00,
0x04 };
```

##### Response

Modbus/TCP:

Transaction identifier (2 Bytes): 0x0001  
 Protocol identifier (2 Bytes): 0x0000  
 Length (2 Bytes): 0x0004  
 Unit identifier (1 Byte): 0xff

Modbus:

Function code (1 Byte): 0x01 (Read coils)

Byte count (1 Byte): 0x01

Coil status (1 Byte): 0x00 to 0x0F (bit 0 = relay state 1, bit 1 = relay state 2, bit 3 = input state 1, bit 4 = input state 2)

Data Byte (hex)	LS Data Bits (binary)	Input 2 State	Input 1 State	Relay 2 State	Relay 1 State
0x00	0000	Low	Low	Off	Off
0x01	0001	Low	Low	Off	On
0x02	0010	Low	Low	On	Off
0x03	0011	Low	Low	On	On
0x04	0100	Low	High	Off	Off
0x05	0101	Low	High	Off	On
0x06	0110	Low	High	On	Off
0x07	0111	Low	High	On	On
0x08	1000	High	Low	Off	Off
0x09	1001	High	Low	Off	On
0x0A	1010	High	Low	On	Off
0x0B	1011	High	Low	On	On
0x0C	1100	High	High	Off	Off
0x0D	1101	High	High	Off	On
0x0E	1110	High	High	On	Off
0x0F	1111	High	High	On	On

Modbus Error:

Function code (1 Byte): 0x81

Exception code (1 Byte): 0x01 or 0x02

Exception codes:

0x01 - Function code not supported

0x02 - Incorrect starting address / quantity of outputs combination

Character String Example:

```
char read_coils_mb_response[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x04, 0xff, 0x01, 0x01, 0x01};
```

### 3.4.2 Read Discrete Inputs (Modbus Function Code 02 (0x02))

This function returns the state of the optically-isolated inputs.

Valid Starting Address and Quantity of coils combinations

Valid starting address can fall in the range of 0x0000 and 0x0001. Valid quantities can fall in the range of 0x0001 and 0x0002. The only restriction is that the sum of the starting address and the quantity of coils must not be greater than 0x0002. For example, if you want to read 0x0002 bits, both inputs, then the starting address must be 0x0000. On the other hand, you may want to read the second input only. This can be achieved by starting at address 0x0001 and using a quantity of 0x0001.

Request

Modbus/TCP:

Transaction identifier (2 Bytes): 0x0001

Protocol identifier (2 Bytes): 0x0000  
 Length (2 Bytes): 0x0006  
 Unit identifier (1 Byte): 0xff

**Modbus:**

Function code (1 Byte): 0x02 (read discrete input)  
 Starting address (2 Bytes): 0x0000 (0x0000 or 0x0001)  
 Quantity of inputs (2 Bytes): 0x0001 (0x0001 or 0x0002)

**Character String:**

```
char read_inputs_mb_request[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x06, 0xff, 0x02, 0x00, 0x00, 0x00,
0x01 };
```

**Response**

**Modbus/TCP:**

Transaction identifier (2 Bytes): 0x0001  
 Protocol identifier (2 Bytes): 0x0000  
 Length (2 Bytes): 0x0004  
 Unit identifier (1 Byte): 0xff

**Modbus:**

Function code (1 Byte): 0x02 (Read discrete input)  
 Byte count (1 Byte): 0x01  
 Input status: 0x00 (input off), 0x01 (input on)

Data Byte (hex)	LS Data Bits (binary)	Input 2 State	Input 1 State
0x00	0000	Low	Low
0x01	0001	Low	High
0x02	0010	High	Low
0x03	0011	High	High

**Modbus Error:**

Function code (1 Byte): 0x82  
 Exception code (1 Byte): 0x01 or 0x02

**Exception codes:**

0x01 - Function code not supported  
 0x02 - Incorrect starting address / quantity of outputs combination

**Character String:**

```
char read_inputs_mb_response[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x04, 0xff, 0x02, 0x01, 0x01 };
```

**3.4.3 Read Holding Registers (Modbus Function Code 03 (0x03))**

This function is used to read the amount of time that has elapsed, in seconds, since a voltage has been applied to input one. This value will continually increment as long as a voltage is applied. As soon as the voltage is removed it will stop. The remaining value will indicate the amount of time that elapsed while the voltage was applied. If a voltage is reapplied to input one, this register will reset to zero and begin the timing process again.

Note that the time is provided in IEEE 754 floating point format. The four data bytes are treated as two individual big endian 16-bit words but the least significant word is sent first. In other words, the 32-byte floating point number represented as ABCD is sent as CDAB. In the example shown below, input one has had a voltage applied to it for 5 seconds. Five seconds is represented using a floating point number of 40A00000. As you can see, it is transferred as 000040A0.

## Request:

## Modbus/TCP

Transaction identifier (2 Bytes): 0x0001

Protocol identifier (2 Bytes): 0x0000

Length (2 Bytes): 0x0006

Unit identifier (1 Byte): 0xff

## Modbus

Function code (1 Byte): 0x03 (read holding registers)

Output address (2 Bytes): 0x0014 (only valid register address is 0x0014 for this function)

Word Count (2 Bytes): 0x0002 (only valid word count is 0x0002 for this function)

```
char read_registers_mb_request[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x06, 0xff, 0x03, 0x00, 0x14, 0x00, 0x02};
```

## Response:

## Modbus/TCP

Transaction identifier (2 Bytes): 0x0001

Protocol identifier (2 Bytes): 0x0000

Length(2 Bytes): 0x0007

Unit identifier (1 Byte): 0xff

## Modbus

Function code (1 Byte): 0x03 (read holding registers)

Byte Count (1 Byte): 0x04 (should always be 0x04)

Data (4 bytes): 0x000040A0 (IEEE 754 Floating Point Number Least Significant Word First. 0X40A00000 = 5.0 seconds)

## Modbus Error:

Function code (1 Byte): 0x83

Exception code (1 Byte): 0x01 or 0x02

## Exception codes:

0x01 - Function code not supported

0x02 - Incorrect starting address / quantity of outputs combination

```
char read_registers_mb_response[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x07, 0xff, 0x03, 0x04, 0x00, 0x00, 0x40, 0xA0 };
```

**3.4.4 Write Single Coil (Modbus Function Code 05 (0x05))**

This function allows you to turn relays one and two on and off one at a time.

## Request:

## Modbus/TCP

Transaction identifier (2 Bytes): 0x0001

Protocol identifier (2 Bytes): 0x0000

Length (2 Bytes): 0x0006

Unit identifier (1 Byte): 0xff

## Modbus

Function code (1 Byte): 0x05 (write coil)

Output address (2 Bytes): 0x0000 or 0x0001 (relay 1 = 0x0000, relay 2 = 0x0001)

Output value (1 Byte): 0xff - turn relay on, 0x00 - turn relay off

Padding (1 Byte): 0x00

```
char write_coil_mb_request[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x06, 0xff, 0x05, 0x00, 0x00, 0xff, 0x00};
```

## Response:

**Modbus/TCP**

Transaction identifier (2 Bytes): 0x0001

Protocol identifier (2 Bytes): 0x0000

Length(2 Bytes): 0x0006

Unit identifier (1 Byte): 0xff

**Modbus**

Function code (1 Byte): 0x05 (write coil)

Output address (2 Bytes): 0x0000

Output value (1 Byte): 0xff - relay on, 0x00 - relay off

Padding (1 Byte): 0x00

**Modbus Error:**

Function code (1 Byte): 0x85

Exception code (1 Byte): 0x01 or 0x02

## Exception codes:

0x01 - Function code not supported

0x02 - Incorrect starting address / quantity of outputs combination

```
char write_coils_mb_response[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x06, 0xff, 0x05, 0x00, 0x00, 0xff, 0x00 };
```

**3.4.5 Write Multiple Coils (Modbus Function Code 15(0x0F))**

This function allows you to control relays one and two simultaneously using one command.

**Valid Starting Address and Quantity of coils combinations**

Valid starting address can fall in the range of 0x0000 and 0x0001. Valid quantities can fall in the range of 0x0001 and 0x0002. The only restriction is that the sum of the starting address and the quantity of coils must not be greater than 0x0002. For example, if you want to control both relays, then the starting address must be 0x0000. On the other hand, you may want to control the second relay only. This can be achieved by starting at address 0x0001 and using a quantity of 0x0001.

**Request:****Modbus/TCP**

Transaction identifier (2 Bytes): 0x0001

Protocol identifier (2 Bytes): 0x0000

Length (2 Bytes): 0x0008

Unit identifier (1 Byte): 0xff

**Modbus**

Function code (1 Byte): 0x0F (write coils)

Output address (2 Bytes): 0x0000 or 0x0001 (relay 1 = 0x0000, relay 2 = 0x0001)

Bit count (2 Bytes): 0x0001 or 0x0002 (control one relay or both relays)

Byte count (1 Byte): 0x01 (always 0x01)

Output value (1 Byte): 0x00 through 0x03 (bit 0 = relay 1, bit 1 = relay 2)

```
char write_coils_mb_request[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x08, 0xff, 0x0F, 0x00, 0x00, 0x00, 0x02, 0x01, 0x03};
```

**Response:****Modbus/TCP**

Transaction identifier (2 Bytes): 0x0001

Protocol identifier (2 Bytes): 0x0000

Length(2 Bytes): 0x0006

Unit identifier (1 Byte): 0xff

**Modbus**

Function code (1 Byte): 0x0F (write coils)  
 Output address (2 Bytes): 0x0000 or 0x0001 (relay 1 = 0x0000, relay 2 = 0x0001)  
 Bit Count (2 Bytes): 0x0001 or 0x0002 (control one relay or both relays)

**Modbus Error:**

Function code (1 Byte): 0x8F  
 Exception code (1 Byte): 0x01 or 0x02

**Exception codes:**

0x01 - Function code not supported  
 0x02 - Incorrect starting address / quantity of outputs combination

```
char write_coils_mb_response[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x06, 0xff, 0x0F, 0x00, 0x00, 0x00, 0x02 };
```

**3.4.6 Write Multiple Registers (Modbus Function Code 16 (0x10))**

This is used to pulse the relays for a specified time. When WebRelay-Dual™ receives this command, it immediately turns the specified relay coil on (if not on already) and starts the pulse timer. The pulse time is specified in the register value field and the time can range from 0.1 seconds to 86400 seconds (1 day). When the pulse time expires, the relay coil will be turned off. If a pulse time command is sent with a value greater than 86400, the pulse timer will be set to 86400. If a pulse time command is sent with a value less than 0.1, the pulse timer will be set to 0.1. If any commands are sent to WebRelay-Dual™ (commands may be modbus, xml, html, or electrical input) before the pulse timer has expired, the pulse timer will be canceled immediately and the new command will be executed.

**Valid address/quantity combinations**

Writing to these registers requires that the address be in the range of 0x10 and 0x13. Also, the address must be evenly divisible by 2. In other words, only addresses 0x10 and 0x12 are valid addresses. When the address 0x10 is used, both relays can be sent pulse times (8 bytes), or just the first relay can be sent a pulse time (4 bytes.) If the address 0x12 is used, only relay 2 can be sent a pulse time (4 bytes.)

Note that the pulse time is provided in IEEE 754 floating point format. The four data bytes are treated as two individual big endian 16-bit words but the least significant word is sent first. In other words, the 32-byte floating point number represented as ABCD is sent as CDAB. In the example shown below, the relay will pulse for 5 seconds. Five seconds is represented using a floating point number of 40A00000. As you can see, it is transferred as 000040A0.

**Request:****Modbus/TCP**

Transaction identifier (2 bytes): 0x0001  
 Protocol identifier (2 bytes): 0x0000  
 Length (2 bytes): 0x000f  
 Unit identifier (1 byte): 0xff

**Modbus**

Function code (1 Byte): 0x10 (Write Multiple Registers)  
 Starting address (2 Bytes): 0x0010, or 0x0012  
 Number of registers (2 Bytes): 0x0002, or 0x0004  
 Byte count (1 Byte): 0x04, or 0x08  
 Registers value (4 Bytes or 8 Bytes): 00 00 40 A0 00 00 40 A0 (pulse both relays for 5 seconds)

```
char write_registers_mb_request = {0x00, 0x01, 0x00, 0x00, 0x00, 0x0f, 0xff, 0x10, 0x00, 0x10, 0x00, 0x04, 0x08, 0x00, 0x00, 0x40, 0xA0, 0x00, 0x00, 0x40, 0xA0};
```

## Response:

## Modbus/TCP

Transaction identifier (2 Bytes): 0x0001

Protocol identifier (2 Bytes): 0x0000

Length (2 Bytes): 0x0006

Unit identifier: 0xff

## Modbus

Function code (1 Byte): 0x10 (Write Multiple Registers)

Starting address (2 Bytes): 0x0010 or 0x0012

Number of registers(2 Bytes): 0x0002 or 0x0004

## Modbus Error:

Function code (1 Byte): 0x90

Exception code (1 Byte): 0x01 or 0x02

## Exception codes:

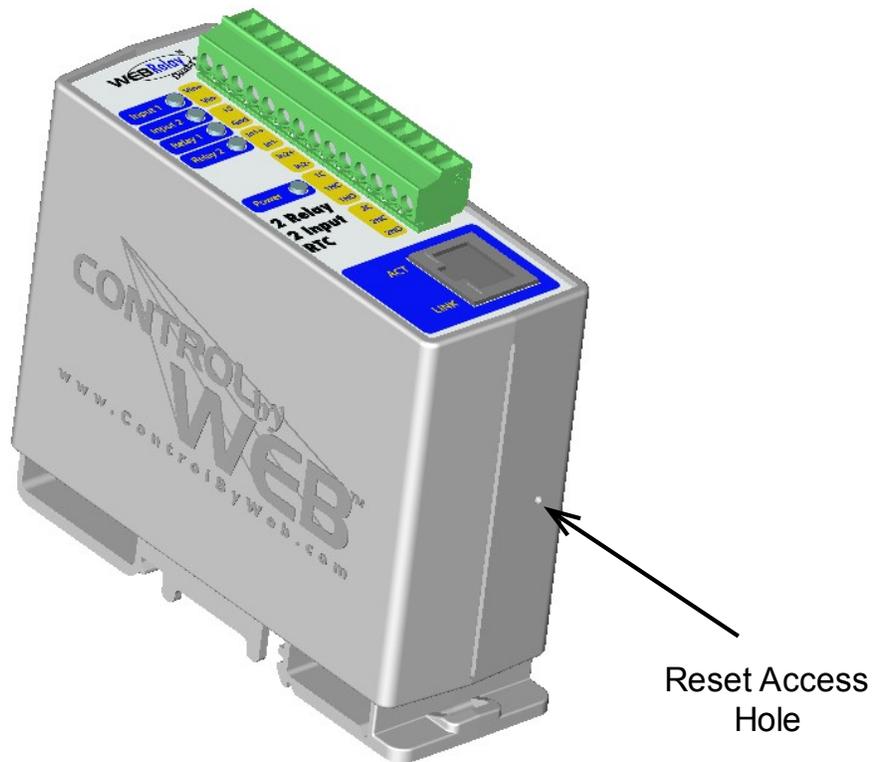
0x01 - Function code not supported

0x02 - Incorrect starting address / byte count combination

```
char write_registers_mb_response[] = {0x00, 0x01, 0x00, 0x00, 0x00, 0x06, 0xff, 0x10, 0x00, 0x10, 0x00, 0x04 };
```

## Appendix A: Restoring Factory Default Settings

In the event that the IP address or passwords are forgotten, WebRelay-Dual™ may be restored to its original factory default settings. To do this, first remove the power from the unit. Next, insert a thin object (such as a toothpick) through the small hole in the bottom of the unit to press the small button that is located inside the unit. When the object is inserted, a tactile feel can be detected as the button is depressed. While holding the button down, apply power and wait for about 10 seconds. After about 10 seconds, release the button. Now all settings will be back to the original factory defaults. CAUTION: DO NOT USE METAL OBJECTS FOR THIS FUNCTION.



## Appendix B: Specifications

### Power Requirements:

#### Model X-WR-2R1-2I5-5

Voltage: 5VDC  $\pm$  5%

Current: See table below for typical values

	10Mbps Network	100Mbps Network
No Relays or Inputs On	130mA	277mA
All Relays and Inputs On	204mA	394mA

#### Model X-WR-2R1-2I5-I or X-WR-2R1-2I24-I

Voltage: 9-28VDC

Current: See table below for typical values

	10Mbps Network	100Mbps Network
9V No Relays or Inputs On	89mA	165mA
9V All Relays and Inputs On	170mA	251mA
12V No Relays or Inputs On	66mA	124mA
12V All Relays and Inputs On	127mA	189mA
24V No Relays or Inputs On	36mA	66mA
24V All Relays and Inputs On	68mA	99mA

#### Model X-WR-2R1-2I5-E

POE Class 1 (0.44 to 3.84 Watt). Optionally can be powered with external 5VDC power supply.

**I/O:** 2 Optically Isolated Inputs, 2 Relay Output

### Relay Contacts:

Contact Form: SPDT

Contact Material: Ag (Au clad)

Max Voltage: 28VAC

Max Current: 1A

Min Load: 1mA @ 5VDC

Relay Control Options: ON/OFF or Pulsed

Pulse Timer Duration: 100ms to 86400 Seconds (1 day)

Accuracy of pulse timer: 99.99%

### Optically Isolated Inputs:

Input Voltage: 3-12VDC (5V Input models)

Input Current: 4.7mA – 25mA (5V Input models)

Input Voltage: 11-28VDC (24V Input models)

Input Current: 4.9mA – 13.4mA (24V Input models)

Input Isolation: 1500V

Input Functions: Monitor, Local Relay Control, Remote Relay Control

Input to Local Relay Settings: equal, invert, toggle, latch, pulse, none

Input to Remote Relay Settings: equal, invert, toggle, latch, pulse, none

**Network:** 10/100 Base-T Ethernet

**Network Setup:** static IP address assignment, TCP port selectable

**Connectors:**

Power/Relay Contacts/Inputs: 14-position, removable terminal strip, 3.81mm terminal spacing  
(Replacement part number, Phoenix Contact 1803691 )

Network: 8-pin RJ-45 socket

**LED Indicators: 4**

- Power on
- Relay coils (1-2) engaged
- Inputs (1-2) on
- Network linked
- Network activity

**Physical:**

Operating Temperature: -20° to 70°C ( -4°-158°F)

Size: 1.38in (35.1mm) wide X 3.77in (95.8mm) tall X 3.07 in(78.0mm) deep

Weight: 4.3oz (122 grams)

Enclosure Material: Lexan 940 (UL94 V0 flame rated)

**Password Settings:**

Password protection on setup page: Yes

Password protection on control page: Optional

Password Encoding: Base 64

Max password length: 10 characters

**Regulatory Compliance:**

Electromagnetic Compliance:

IEC CISPR 22, CISPR 24

FCC 47CFR15 (-I and -5 Models class B, POE Model Class A)

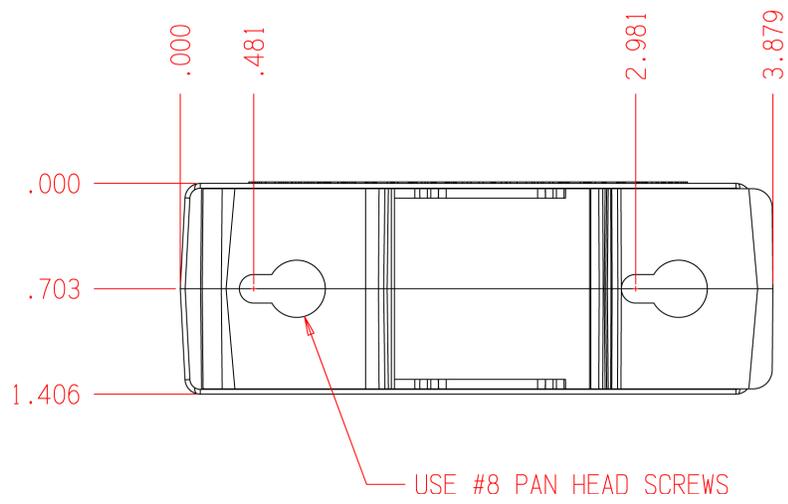
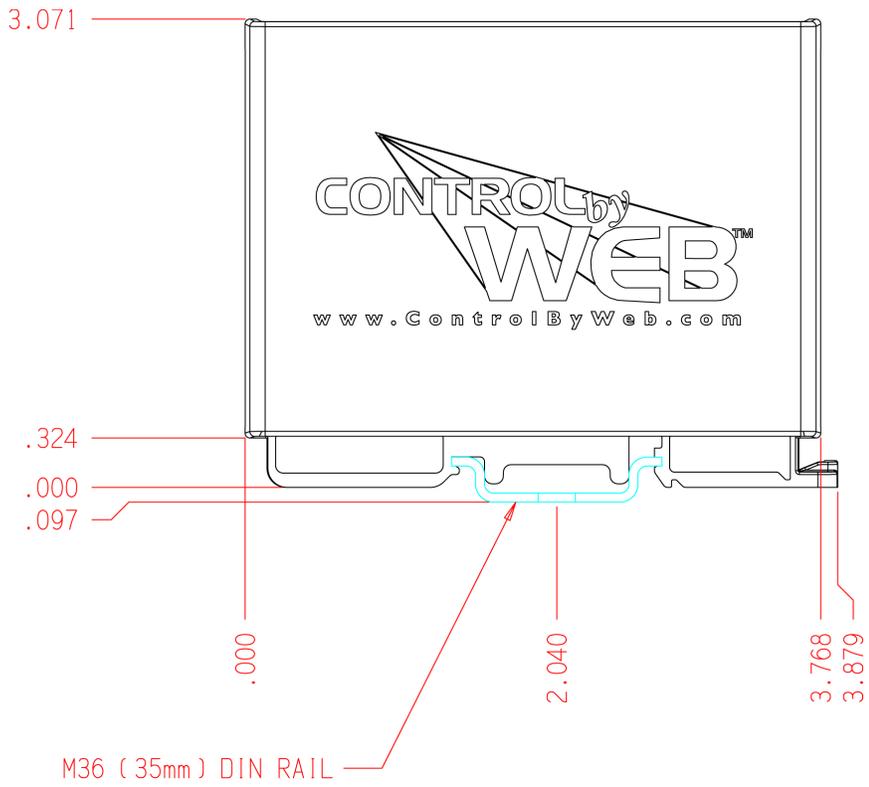
EU EN55024, EN55022

Product Safety:

IEC 60950-1 / EN 60950-1



### Appendix C: Mechanical Information



## Appendix D: Power Failure

WebRelay-Dual™ stores all configuration data in non-volatile memory so user configuration will not be lost when power is removed from the device.

The real-time clock circuit includes a very large capacitor that keeps the clock running when main power is removed. The capacitor charges when the unit is powered and will keep the clock circuit running for a minimum of several days when power is removed. Unlike battery backups, the capacitor does not have to be replaced on a regular basis.

When power is applied to WebRelay-Dual™, the state of the relays depend upon the following conditions...

First, if the Relay Options is set to either of the settings that include *init at powerup* (for example *set relay equal input (init at powerup)*). The relay state will be set to the appropriate state according to the input. This setting takes the highest precedence.

Second, if the Relay Options is not set to either of the settings that include *init at powerup*, WebRelay-Dual™ will evaluate the events. If the most recent event is set to the Relay Action of *turn relay on*. The relay will be turned on. Any other settings (including *pulse relay*, or *toggle relay* will not cause the relay to turn on).

If the Relay Options is not set to either of the settings that include *init at powerup*, and the Relay Action of the most recent events are not set to *turn relay on* the relay coil will remain off.

## Appendix E: Open TCP Legal Notice

Portions of the software used in WebRelay-Dual™ are open source. The appropriate notices are listed below. All other parts of the software are property of Xytronix Research & Design, Inc. ©2005-2006.

Copyright (c) 2000-2002 Viola Systems Ltd.  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:  
"This product includes software developed by Viola Systems (<http://www.violasystems.com/>)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "OpenTCP" and "Viola Systems" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [opentcp@opentcp.org](mailto:opentcp@opentcp.org).
5. Products derived from this software may not be called "OpenTCP", nor may "OpenTCP" appear in their name, without prior written permission of the Viola Systems Ltd.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL VIOLA SYSTEMS LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

OpenTCP is the unified open source TCP/IP stack available on a series of 8/16-bit microcontrollers, please see <http://www.opentcp.org>.

For more information on how to network-enable your devices, or how to obtain commercial technical support for OpenTCP, please see <http://www.violasystems.com/>.