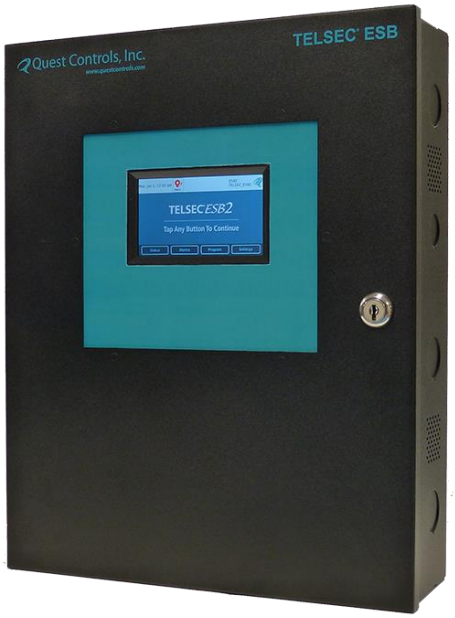
Quest Controls
Intelligent Solutions. Controlled.

TELSEC® ESB2/ESB2x

User Manual



Rev 3.5 10/08/25

CRITICAL APPLICATIONS DISCLAIMER

QUEST PRODUCTS ARE NOT INTENDED OR AUTHORIZED FOR USE IN ANY APPLICATION THAT REQUIRES FAIL-SAFE OPERATION, OR ANY SYSTEM THAT IS RESPONSIBLE FOR PROVIDING DETECTION, MONITORING OR CONTROL OF CRITICAL SAFETY, LIFESAVING OR LIFE-SUPPORT CONDITIONS, FUNCTIONS OR OPERATIONS (INCLUDING BUT NOT LIMITED TO FIRE DETECTION, ALARMING, OR SUPPRESSION), OR ANY APPLICATION WHEREIN A FAILURE OR MALFUNCTION OF THE PRODUCT MAY LEAD TO SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE, PERSONAL INJURY OR DEATH.

Lithium Battery Notice

Replace Battery with Panasonic CR2032 or equivalent Lithium Battery (Quest PN 250034), rated at 3 Vdc. Use of another battery may present a risk of fire or explosion.

CAUTION - Battery may explode if mistreated. **Do not** recharge, disassemble or dispose of in fire.

FCC Compliance Statement

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

FCC Caution: Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment.

COPYRIGHT NOTICE

Copyright © 2020-25 by Quest Controls, Inc. (Quest). The material discussed in this publication is the proprietary property of Quest. Quest retains all rights to reproduction and distribution of this publication.

Specifications are subject to change without notice.

Revision History

Date	Rev #	Modifications	By:
10/08/25	3.5	Added process to return to Factory Default	K. Nickel
10/21/24	3.4	Added Appendix F – IP protocols and ports used by product.	K. Nickel
07/31/24	3.3	Updated the Shell Level access for the new status commands	K. Nickel
11/3/23	3.2	Correct section 11.2.4.11 for proper format of Boolean operators	K. Nickel
05/09/23	3.1	Added section 11.2.5 using the comment function in equations	K. Nickel
09/16/22	3.0	Updates for features in Build #14 -e and Build#4-t: • Password rules for http, SSH/Telnet and SNMPv3 • TL1 type addition for alarms • Support for 8 Time Schedules	K. Nickel
04/20/22	2.1	Minor update to specification section	K. Nickel
10/26/21	2.0	Updated to support new features and web page organization in Build # 10	K. Nickel
03/5/21	1.3	Corrected “waitUntil” function Section 11.2.4.3 Updated install drawings for power connectors	K. Nickel J. Sarnie
01/15/21	1.2	Minor grammatical fixes, added additional equation examples	K. Nickel
12/18/20	1.1	Added SNMP tables, upload/download file format and updated ESB2 photos.	K. Nickel
10/17/20	1.0	Initial release of manual	K. Nickel J. Sarnie

Table of Contents

1 Introduction 1
2 Product Description 1
2.1 Specifications 5
3 Installing the ESB2 7
3.1 Choosing a Location 7
3.2 Mounting the ESB2 to the Wall 7
3.3 Powering the ESB2 8
3.4 Connecting Inputs, Outputs and Peripherals 10
3.5 TELSEC ESB2 Modbus 13
4 Touchscreen Operation 17
4.1 Status Menu 17
4.2 Alarm Status 18
4.3 Program Menu 20
4.4 Settings Menu 21
5 Communicating with the ESB2 22
5.1 Connecting to the System 22
5.2 Web Page Navigation 23
6 System Status 24
6.1 Input Status 24
6.2 Modbus Status 25
6.3 Output Status 26
6.4 Control Point Status 28
6.5 Alarm Status 31
6.6 Active Alarm Log 32
6.7 Historical Alarm Log 32
6.8 Log Data 34
7 Bypassing 35
7.1 Bypass Inputs 35
7.2 Bypassing Digital Outputs 36
7.3 Bypass Analog Outputs 37
8 Configuring the ESB2 38
8.1 System Setup 38
8.2 Alarm Dispatch Settings 45

8.3 HTTP User Administration 47
8.4 Advanced Settings 48
9 Programming the System 51
9.1 Input Programming 51
9.2 Output Programming 54
9.3 Analog Output Programming 55
9.4 Modbus Programming 56
9.5 QMbus Modules 61
9.6 Alarm Programming 62
9.7 Date Schedules 68
9.8 Time Schedules 68
10 Shell Commands 72
10.1 Help Command 73
10.2 show alarms states 73
10.3 show alarms history 73
10.4 show inputs points history 73
10.5 show inputs points 74
10.6 show outputs digitals points history 74
10.7 show outputs analog points 74
10.8 show outputs digital points 75
10.9 show equation setpoints 75
10.10 show modbus points history 75
10.11 show modbus points 75
10.12 show modbus ports addrs points 76
10.13 show modbus ports addrs status 76
10.14 show modbus ports points 76
10.15 show modbus ports status 77
10.16 show system info 77
10.17 Set IP Command 77
10.18 Set MTU Command 78
10.19 Set httppswd Command 78
10.20 Set snmpmgr Command 78

10.21 Upgrade URL 79
10.22 Checking for Upgrades 79
10.23 Resting the System 79
10.24 Ping Command 80
10.25 Exit & Logout Command 80
11 Equation Programming 81
11.1 Defining Equation points 81
11.2 Creating Equations 83
11.3 Equation Examples 93
12 File upload format 96
12.1 Section Formats 96
12.2 Uploading config files 101
13 SNMP Sets & Gets 102
13.1 Public SNMPv2 102
13.2 Input table 102
13.3 Alarm point table 102
13.4 Modbus point table 103
13.5 Output table 104
13.6 Active alarm point table 104
13.7 Analog output table 105
13.8 Setpoint table 105
13.9 Scalar values 105
14 Troubleshooting 106
14.1 Power Up 106
14.2 Communication 106
14.3 Sensor Reading 106
14.4 Control Point Failure 106
14.5 Alarm Notification Failure 107
14.6 Battery 107
14.7 Modbus Communication and Status 107
14.8 Return to Factory Default 108
Appendix A - SNMP Trap Format 109
Appendix B - RADIUS Server Configuration 110

Appendix C – SD Card Update 111
Appendix D – Email Alarm Format 112
Appendix E – TL1 Alarms and Commands 113
Appendix F – Network Settings and Supported Protocols 116
Contact Information 117

1 Introduction

Congratulations on your purchase of the TELSEC ESB2 Controller or TELSEC ESB2x Controller!

The TELSEC ESB2 and TELSEC ESB2x (both models herein referred to as the TELSEC ESB2 unless otherwise specified), are state-of-the-art electronic monitor and control systems that provide you with an integrated surveillance solution for monitoring, control and management of your critical facilities. Acting as your smart eyes and ears at the remote site, the TELSEC ESB2 is capable of performing your HVAC and lighting control and monitoring, environmental monitoring, power monitoring, generator monitoring, equipment monitoring, intrusion alarming and door access control --- all in ONE product.

This manual is intended to provide the information needed to install, program and operate the TELSEC ESB2 and to highlight the TELSEC ESB2's features, specifications, and applications. Please contact us at Quest if you have product questions or suggestions to improve this Manual.

2 Product Description

The TELSEC ESB2 is an intelligent, integrated surveillance solution to monitor, alarm and control all environmental equipment. The system can be programmed to monitor, automate and optimize many facility functions. These include but are not limited to lighting schedules, power monitoring and environmental controls such as HVAC optimization, intelligent lead/lag control, exhaust fans, make up air units and standalone free cooling systems. The TELSEC ESB2 allows for on-site monitoring with a color touchscreen or remote interrogation from any maintenance or alarm center through an Ethernet connection.

Figure 1 TELSEC ESB2 (left) and TELSEC ESB2x (right)

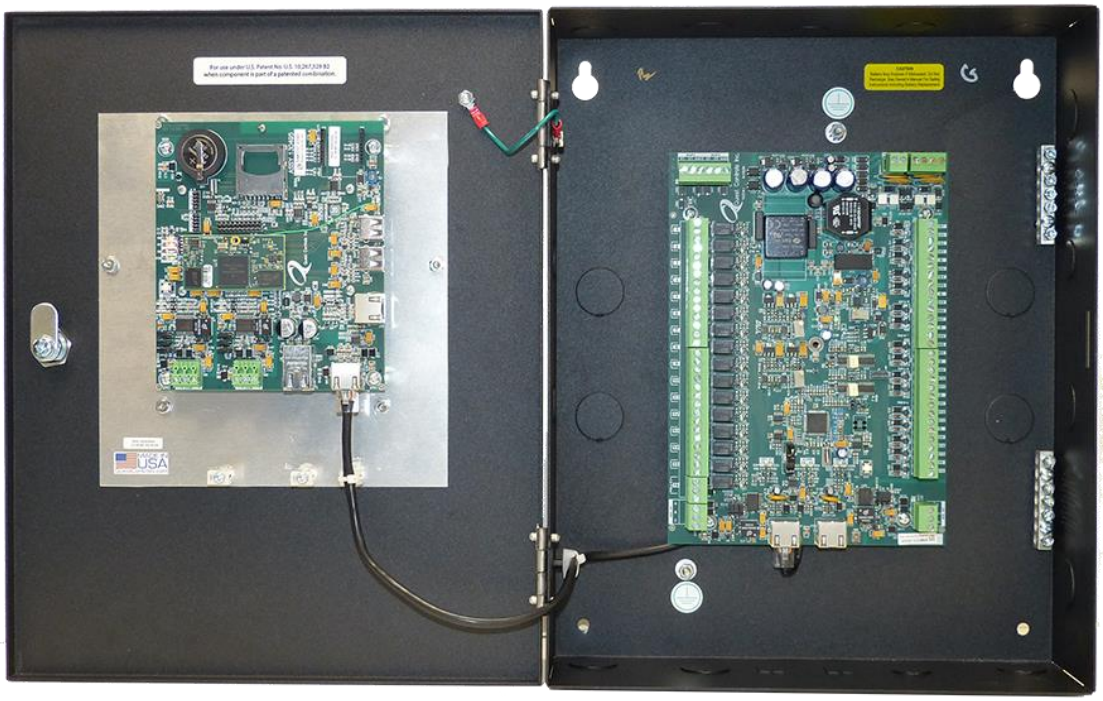


Figure 2 TELSEC ESB2 Inside Panel

Figure 3 TELSEC ESB2x Inside Panel

Figure 4 Optional Single Expansion Panel

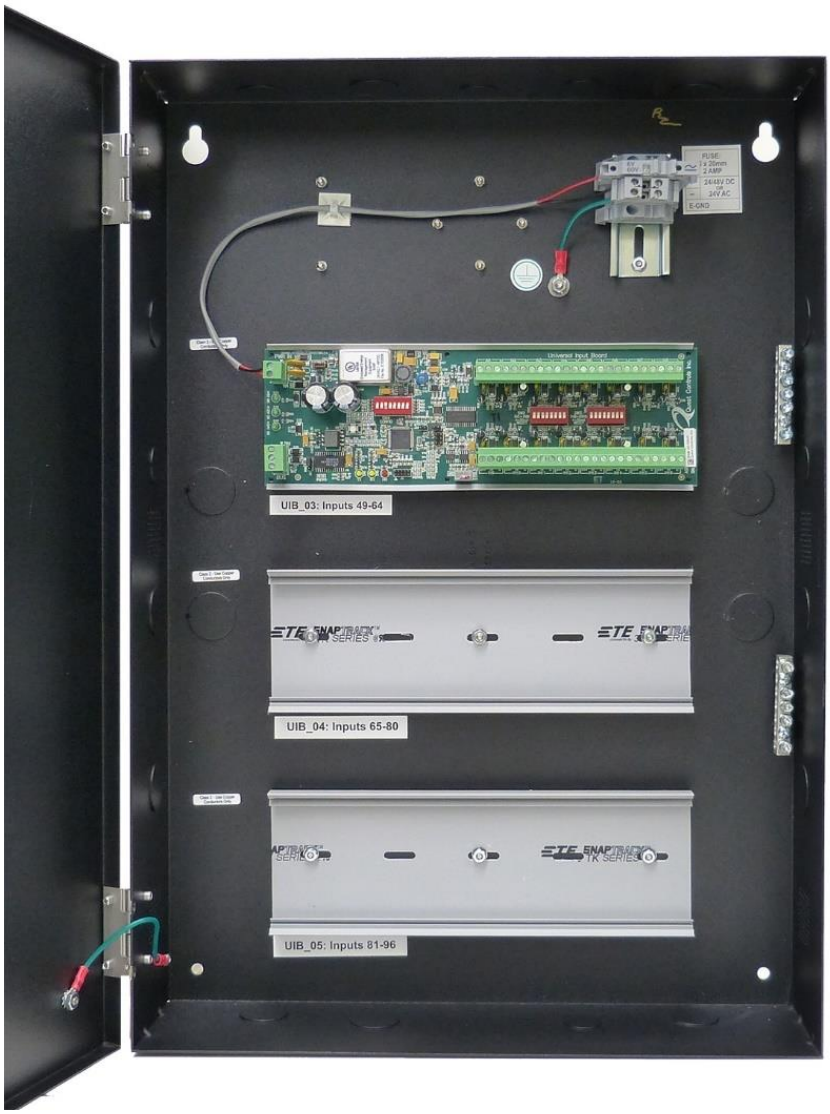


Figure 5 Optional Three Board Expansion Panel

2.1 Specifications

Table 1 ESB2 Power, Physical and Environmental Specifications

Specification	Description
Input Power	
TELSEC ESB2	24 VAC or +/-20-60 VDC, current draw 0.5 amps at 24 VDC
TELSEC ESB2x	24 VAC or +/-20-60 VDC, current draw 0.5 amps at 24 VDC
Physical	
TELSEC ESB2 Dimensions Weight	 12.25"W x 15.25"H x 3.92"D (311mmW x 387mmH x 92mmD) 14 lbs. (6.4 kg)
TELSEC ESB2x Dimensions Weight	 12.25"W x 22"H x 3.5"D (387mmW x 559mmH x 89mmD) 22 lbs. (9 kg)
Mounting	Wall mount
Front Panel	Color Touch Screen LCD 4.3" diagonal with 24-bit TrueColor. WQVGA resolution (480 x 272)
Environmental	
Operating Temperature	-20F to 120F (-29C to 82C)
Storage Temperature	-40F to 158F (-40C to 70C)
Humidity	0 to 95% RH, non-condensing
Internal Battery	Retains memory during power failure events. Long life lithium, 10-year shelf life, 1.5 years under load
Certifications	UL, cUL, FCC 47 CFR Part 15B, ISED ICES-003 Class A
Warranty	One (1) year

c UL US LISTED

Table 2 ESB2 Input and Output Specifications

Specification	Description
Inputs and Outputs	
Inputs	16 Universal inputs, expandable to 272, in 16 input increments (part# 130455 for Universal Input Expansion Board) Configurable for one of the following: 0 – 6 VDC 0 – 20 mA Dry contacts Wet contacts 0-60 VDC Thermistors (accuracy ±1°F (±0.5°C))

Analog Outputs (optional)	2 analog outputs on board, expandable to 66, in 8 output increments. (optional part # 130456 for Analog Output Expansion Board- AOB) Configurable for one of the following: 0 – 10 VDC 4 – 20 mA
Digital Outputs	16 digital outputs, expandable to 144, in 8 output increments. (part# 130454 for Digital Output Expansion Board - DOB)
Contacts & Contact Ratings: ESB2 Digital Outputs Digital Output Board	 16 Form A, 1.0 amp/30VDC, 0.3 amps/60 VDC, 0.5 amps/125 VAC 8 Form C, 10 amps at 28 VDC, 8.5 amps at 120 VAC
I/O Terminal Type	2-piece pull-off terminals for all field wiring

Table 3 ESB2 Communications Specifications

Specification	Description
Communications Interface	
Ethernet Network	10/100 Base-T Ethernet
Application	Communications to/from the ESB2 and the IP network switch
Connector/Cable Length	RJ-45, 330 ft. maximum cable length (100 M)
Protocols	IPv6, IPv4, HTTP, HTTPS, RADIUS Password Authentication, NTP, SMTP (email), SSHv2, Telnet, SNMP v1, v2c and v3 for Gets, Sets and traps/informs
RS-485 Serial Data (3)	
Application	Used for communicating with ESB2 Input/ Output expansion modules. Two RS-485 ports for Modbus devices
Connector	3 position screw terminals

3 Installing the ESB2

3.1 Choosing a Location

The TELSEC ESB2 wall mount enclosure is designed to mount to the wall of any type of facility. The unit should be mounted at eye level in a well-lit, protected area free from excessive vibration and moisture, usually in a maintenance room or other non-public location near the circuit breaker panel. Leave sufficient area around the enclosure to allow for routing of conduit and cable or adding expansion panels as necessary.

3.2 Mounting the ESB2 to the Wall

1. Mark and drill mounting holes for screws and anchors (not included) to mount the enclosure to the selected location.
2. Using appropriate hardware (#8 x ¾ pan head wood screw or equivalent) fix the top mounting hardware to the wall approximately six feet above the floor, 10.5 inches apart and level.
3. Slip the TELSEC ESB2(x) top mounting holes, (as shown in Figure 6a & 6b) onto the top mounting hardware and slide down until it stops.
4. Install the bottom mounting hardware in the bottom mounting holes.

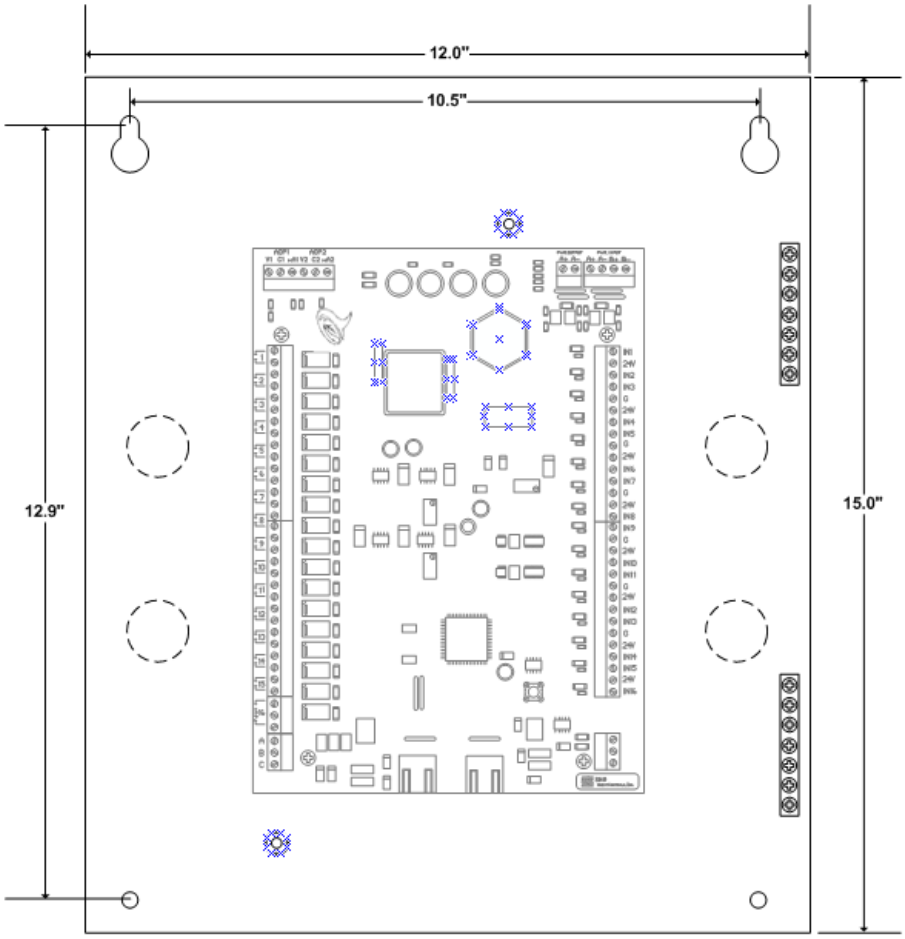


Figure 6a ESB2 - Mounting Holes and Grounding Studs

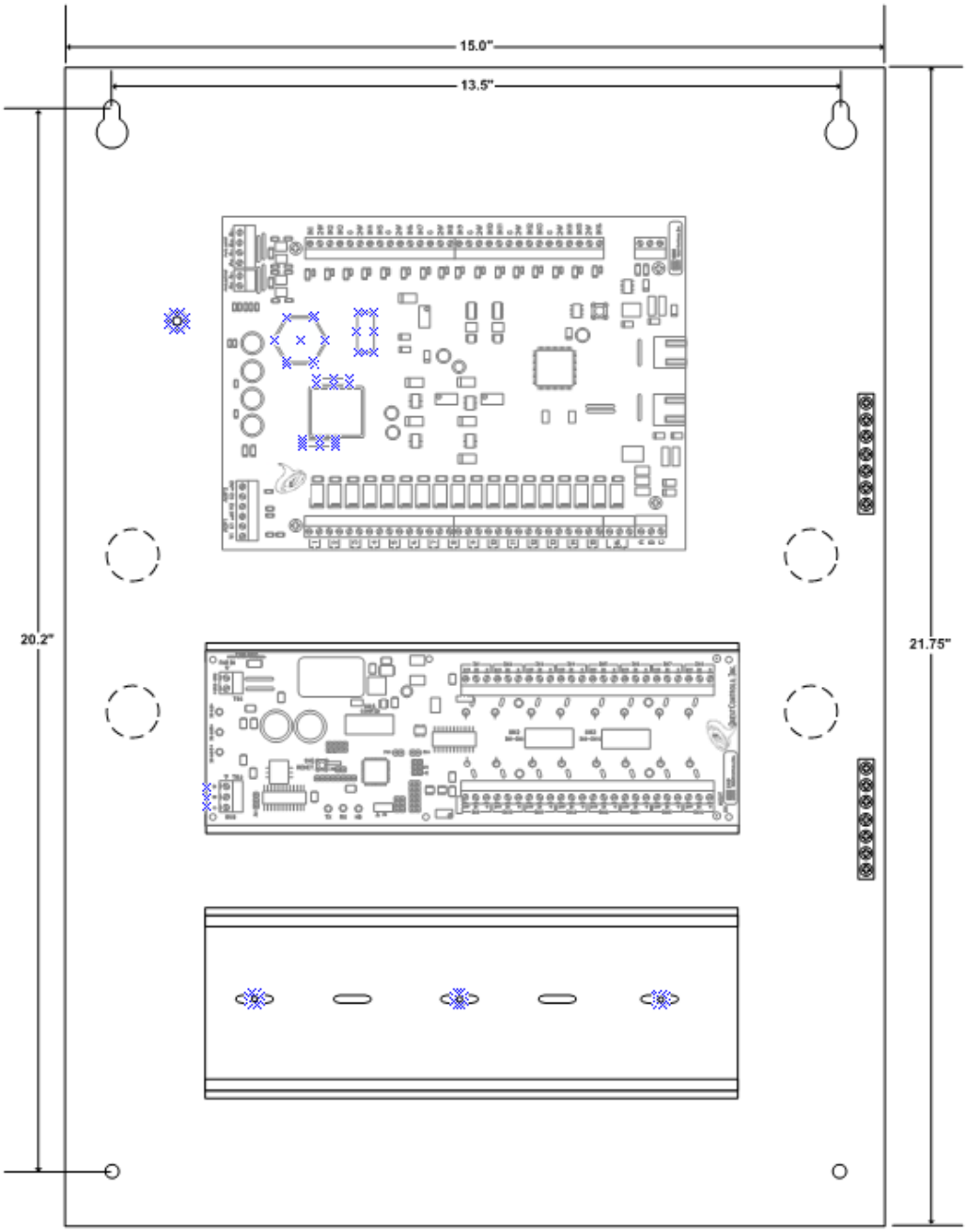


Figure 6b ESB2x - Mounting Holes and Grounding Studs

3.3 Powering the ESB2

The TELSEC ESB2(x) requires 24VAC or 20 - 60 VDC@0.5 amps to power the unit. It is also equipped with a long-life lithium battery as a power back up to retain the memory during power failure events.

An appropriate power supply, meeting the ESB2's power requirements, must be installed and connected to the ESB2. Locate the power supply near the TELSEC ESB2 and install the power supply in accordance with the power supply's instructions; ensure the power supplies buttons, switches and connections are protected from inadvertent traffic that could accidentally cause the power supply to power off.

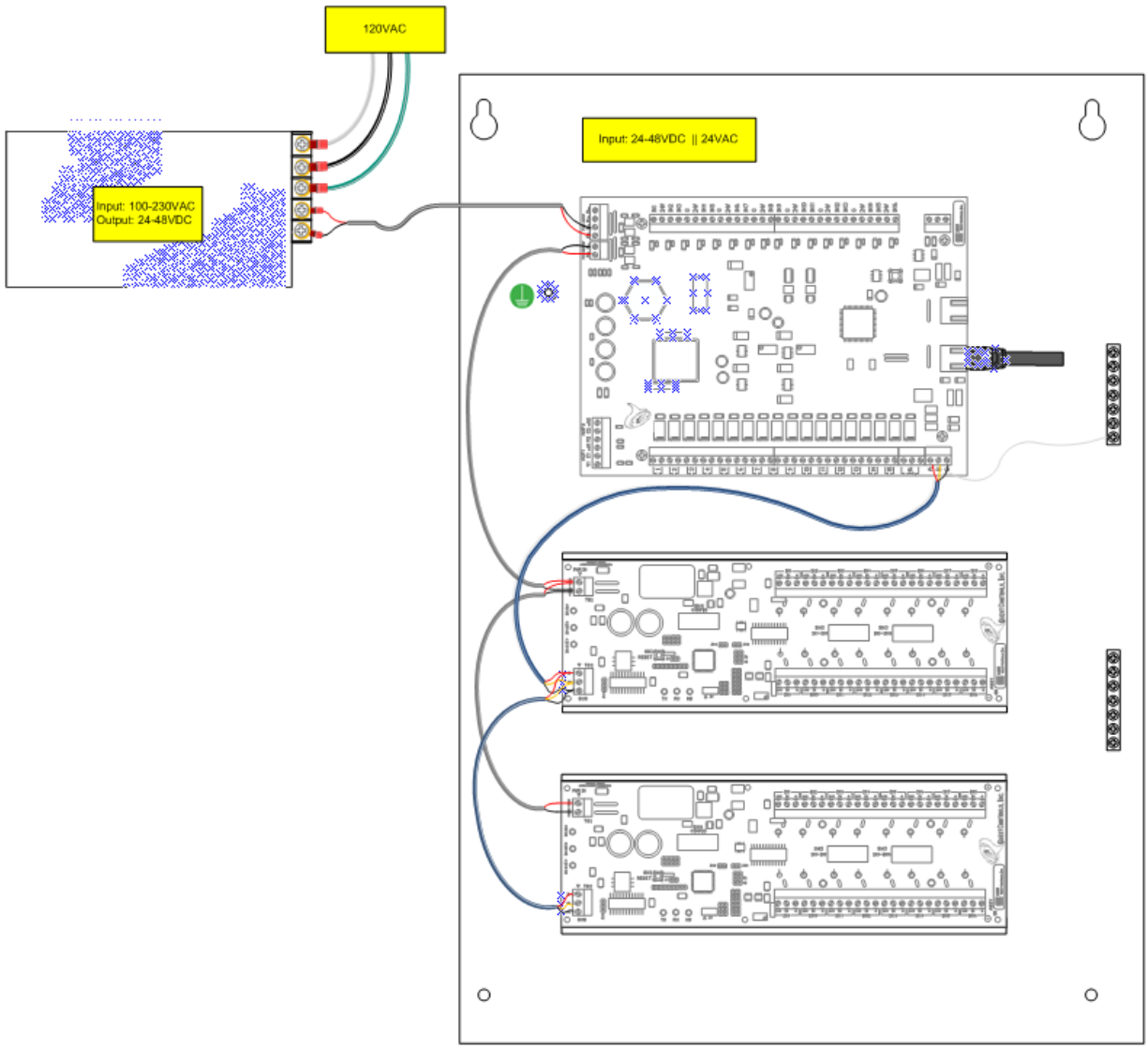


Figure 7 Powering the ESB2(x)

NOTE	Contact a Quest Controls sales representative for assistance with identifying an appropriate power supply and part number when ordering a power supply for the ESB2.

3.3.1 Connecting the Ground

1. Select the most convenient ground stud on the enclosure; all grounding studs and ground bus bars are common.
2. Using terminal lug, run 12 AWG (or larger) ground wire from one of the TELSEC ESB2 ground studs, identified in Figure 7, to the building ground and connect using appropriate hardware.

3.4 Connecting Inputs, Outputs and Peripherals

The TELSEC ESB2 is configured with 16 universal inputs that accept digital or analog signals, 16 digital outputs and 2 optional analog outputs for flexible control options. Using the RS-485 Input and Output expansion bus connections, the system is expandable to 272 universal inputs, 144 digital outputs and 66 analog outputs. The Inputs and Outputs are designed for easy installation of a wide array of peripheral equipment to customize your system.

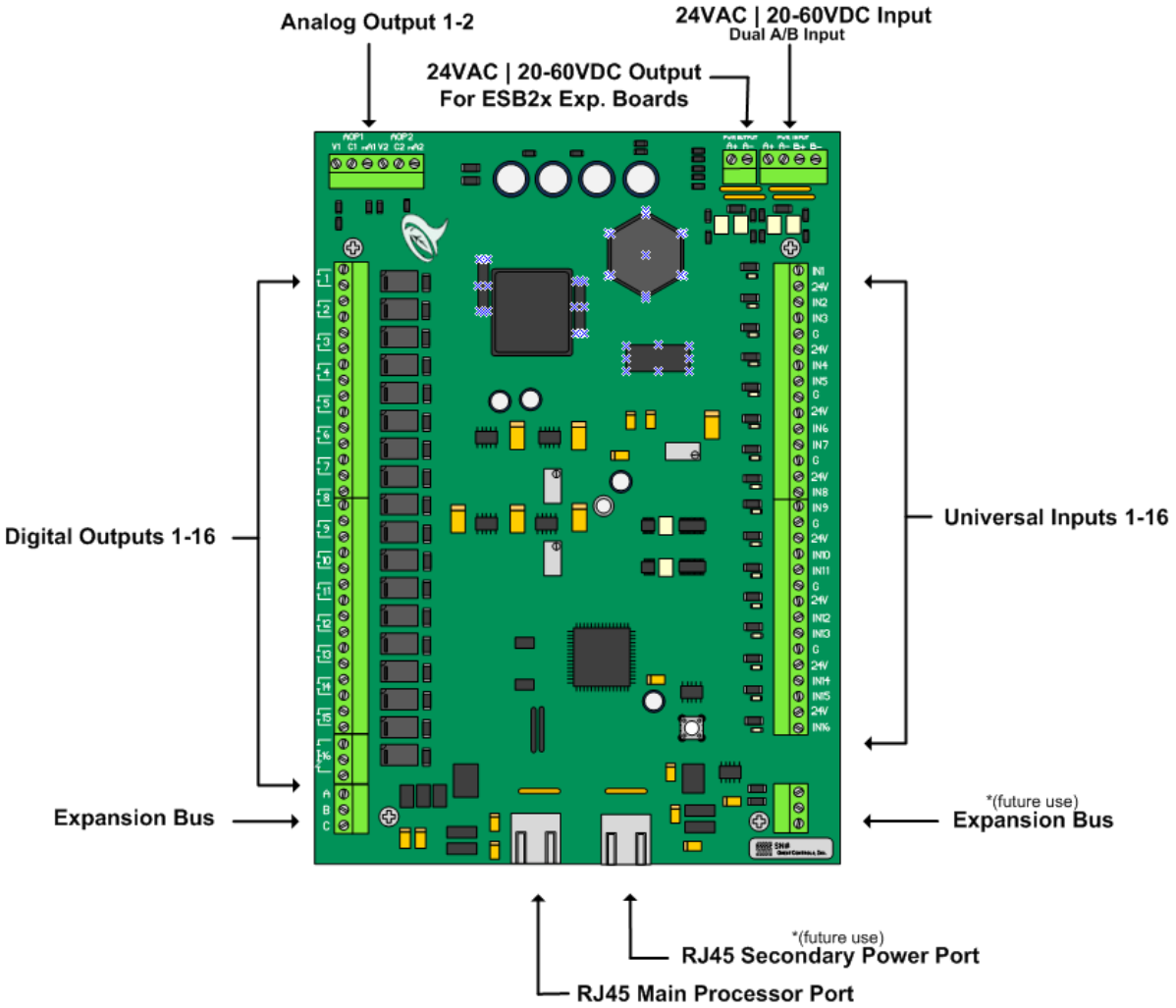


Figure 8 Location of Input, Outputs, and Peripheral Connectors

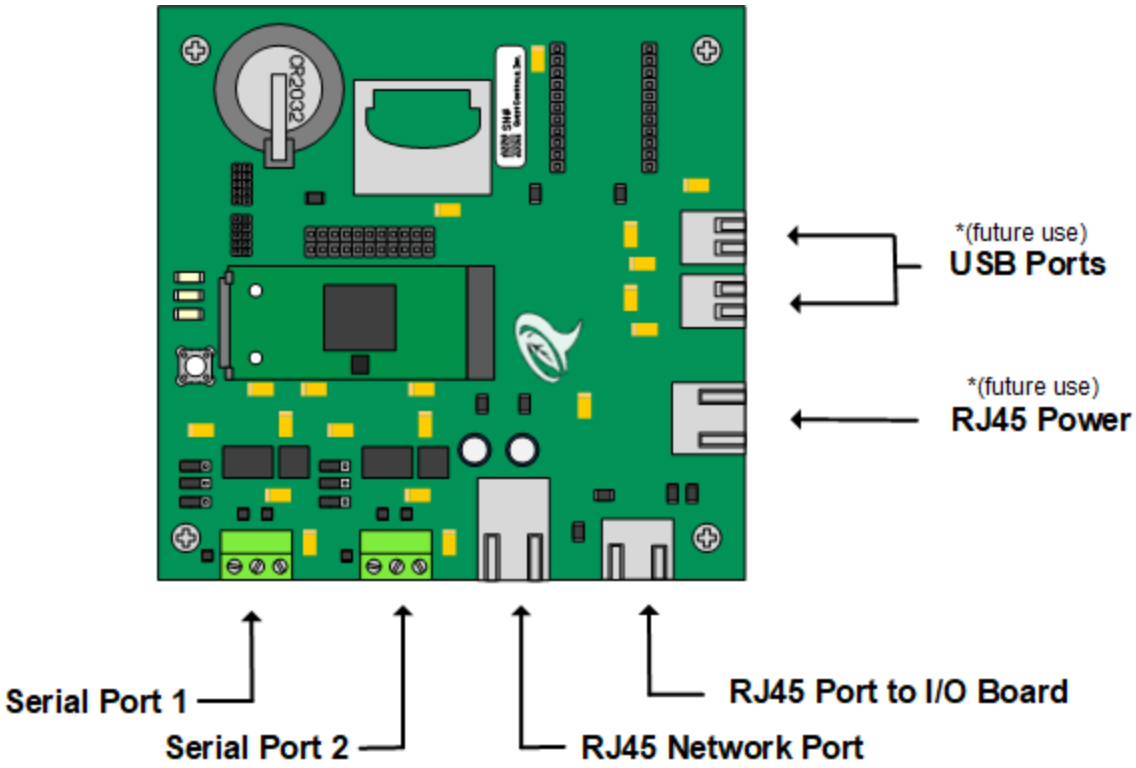


Figure 9 Location of Network and Peripheral Connectors

3.4.1 Connecting Inputs

The TELSEC ESB2 logic board includes 16 universal inputs located on the right side of the board, as shown in Figure 8 Location of Input, Outputs, and Peripheral Connectors. The logic board has 24VDC source voltage terminals available for powering sensors and/or sensing voltage for contact closures.

3.4.1.1 Analog Inputs

The inputs will accept 10K Type III thermistors, 0-6 VDC or 0-20 mA signals. The system has predefined scales for temperature Fahrenheit and Celsius along with the ability to define scales for each input to read in the proper engineering units.

1. Mount the device to be monitored in the desired physical location.
2. Identify a specific input (# 1 – 272) for the device to be monitored.
3. Using, as a minimum, 22 AWG wire, connect the device to "IN#" terminal of the assigned input, where "#" is the assigned input number. For thermistor sensors, connect the other lead to a G terminal. For inputs requiring a source voltage, connect the other lead to a 24V source voltage terminal. Consult the wiring instructions that come with the device for further direction.

For a 4-20 mA Device

Place a 249 Ohm resistor between the input and ground reference to convert the 4 – 20 mA current reading to a voltage. For complete connection instructions, refer to the instruction sheet for the specific sensor.

For a 10K Type-3 Thermistor Temperature Sensor

Connect the sensor between input and ground reference.
For complete connection instructions, refer to the instruction sheet for the specific sensor.

3.4.1.2 Digital Inputs

The universal inputs will accept a dry contact closure or a wet contact of 0-60VDC. Connect dry contacts between IN and G (ground) and define the input as DRYNC or DRYNO depending if the normal state is Open or Closed. For wet contacts connect the sensing voltage to the IN terminal for the input. There must be a ground connection between the ESB2 and the device being monitored. Any of the G terminals on the ESB2 will satisfy this requirement.

Upgrading Existing ESB systems

Existing ESB systems had digital inputs wired between the 12V source voltage and IN# terminals. When upgrading to the ESB2 you do not need to rewire. Leave the wiring between the source voltage (now 24V on the ESB2) and IN# and define the inputs as WETNC or WETNO depending on the normal state of the input being Open or Closed.

3.4.2 Connecting Outputs

3.4.2.1 Connecting Digital Outputs

Sixteen (16) normally open contacts are located on the left side of the TELSEC ESB2 logic board, shown in Figure 8 Location of Input, Outputs, and Peripheral Connectors. Fifteen (15) of the contacts (#1 thru #15) are available for controlling low voltage pilot duty relays using Class 2 wiring. Output #16 has both normally open and normally closed contacts. All relays are rated for 1 amp at 30V AC or DC, refer to Table 2 for additional rating specs.

1. Connect an 18AWG wire to the device being controlled.
2. Terminate the wire at the relay terminal assigned to control the device.

3.4.2.2 Connecting Analog Devices

There are two analog outputs on the upper left corner of the TELSEC ESB2 board, shown in Figure 8 Location of Input, Outputs, and Peripheral Connectors, labeled AOP1 and AOP2. Each analog output has terminals for providing 0-10 VDC output or 4-20 mA output. These outputs are typically used for controlling dampers, actuators and variable speed drives.

1. Determine whether the analog output is to be used as a voltage or current output.
2. Connect 18AWG wire from the common and either the VDC output or the mA output depending on the device being controlled.

For complete connection instructions, refer to the instruction sheet for the specific analog device.

3.4.3 Connecting Peripherals

A wide array of peripherals is available for the TELSEC ESB2. To install the peripheral(s) ordered with your unit, refer to the installation instruction sheet for that peripheral.

3.4.4 Connecting Expansion Boards

The TELSEC ESB2's input and output capacity can be increased with the addition of three different types of expansion boards:

Universal Input Board (UIB) - The Universal Input board (UIB) allows up to 16 individual inputs to be added, these include contact closure and analog inputs as well as an array of sensors and transducers. A maximum of sixteen boards can be added for a total of 272 universal inputs.

Digital Output Board (DOB) - The Digital Output Board allows for additional control points to be added. The digital board includes eight outputs. A maximum of sixteen digital boards can be connected to the TELSEC ESB2 for a total of 144 digital outputs.

Analog Output Board (AOB) - The Analog Output Board (AOB) allows for four (4) additional control points. A maximum of sixteen AOBs can be added to the system for a total of 66 analog outputs.

3.5 TELSEC ESB2 Modbus

The TELSEC ESB2 has two RS485 ports that can be used for communicating with Modbus RTU devices for monitoring, alarming and control. Devices are networked to the ESB2 using one of the RS485 communications ports in a daisy-chain fashion. The ESB2 is programmed to gather Modbus data from the external equipment and trend the data. The Modbus point data can be compared to user-defined thresholds in order to generate alarms using the standard SNMP, SMTP (email) and TL1 mechanisms of the ESB2. Additionally Modbus points can be used in control equations.

3.5.1.1 Modbus Interface Connector, Pins and Jumpers

1. There are two serial ports on the processor board located on the door of the ESB2. The ports are labeled COM1 and COM2
2. The connector pins from left to right are D0/-, D1/+ and C for common.
3. The TELSEC ESB2 Modbus interface line termination jumper is labeled as J8 for COM1 and J11 for COM2

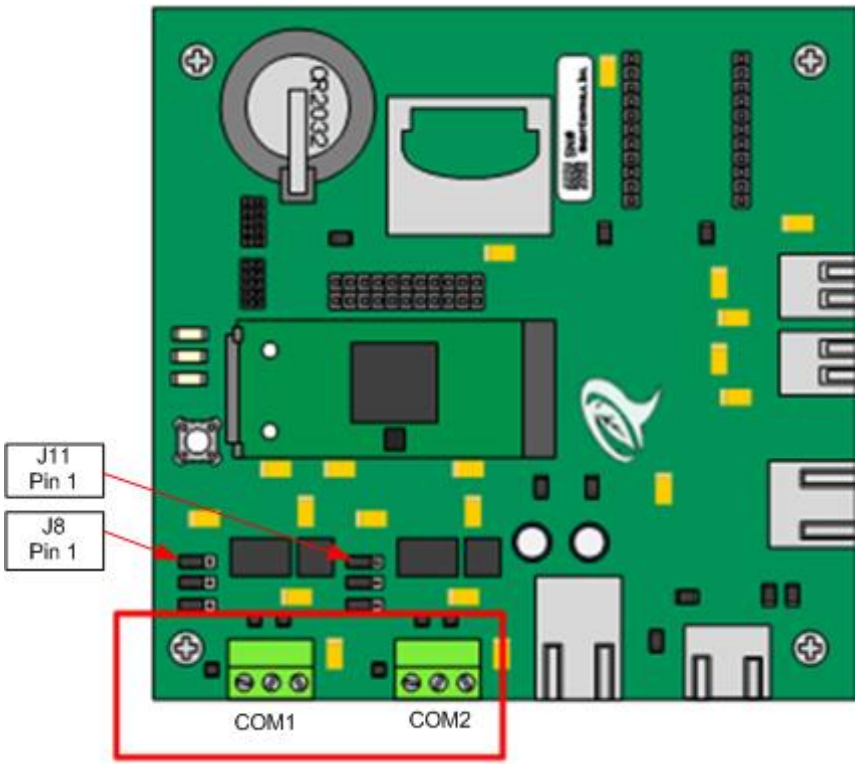


Figure 10 Modbus Interface Connections

3.5.1.2 Cabling

1. Connect no more than 32 Modbus devices to each serial port.
2. The maximum overall Modbus cable network length is 3,000 feet.
3. For normal applications, use Belden Cable #88102 or equivalent. This cable has two twisted pairs and a separate foil shield. For applications requiring a more mechanically robust cable, use Belden Cable #9842 or equivalent. This cable has two twisted pairs, a separate shield and a separate braided shield.
4. Use one of the twisted pairs for the D0 (-) and D1 (+) connections and use the second twisted pair for the C/Common connections. The cable shields must never be used for the C/Common connections.
5. All of the D0 (-) pins from every Modbus device on the network must be connected together. The same is true with the D1 (+) pins, and the C/Common pins.
6. Connect the Modbus devices together in a daisy-chain configuration only. No star, ring or tree cabling network configurations, or additional cable stubs are allowed. Refer to Figure 11 - Typical Modbus Networks
7. Connect all the cable shields together so that they form one continuous connection from one end of the Modbus cable network to the other. If a braided shield is also included in the cable, then connect all of them together as well. If there is a mix of braided and non-braided shields in the cabling network, then connect the braided shields together and then to the foil shields of the nearest non-braided cable. At no time are any shields to be left floating.
8. At the TELSEC ESB2, connect the cable shield to the protective earth ground. The cable shields must not be connected to earth ground anywhere else in the cabling network except at this point.

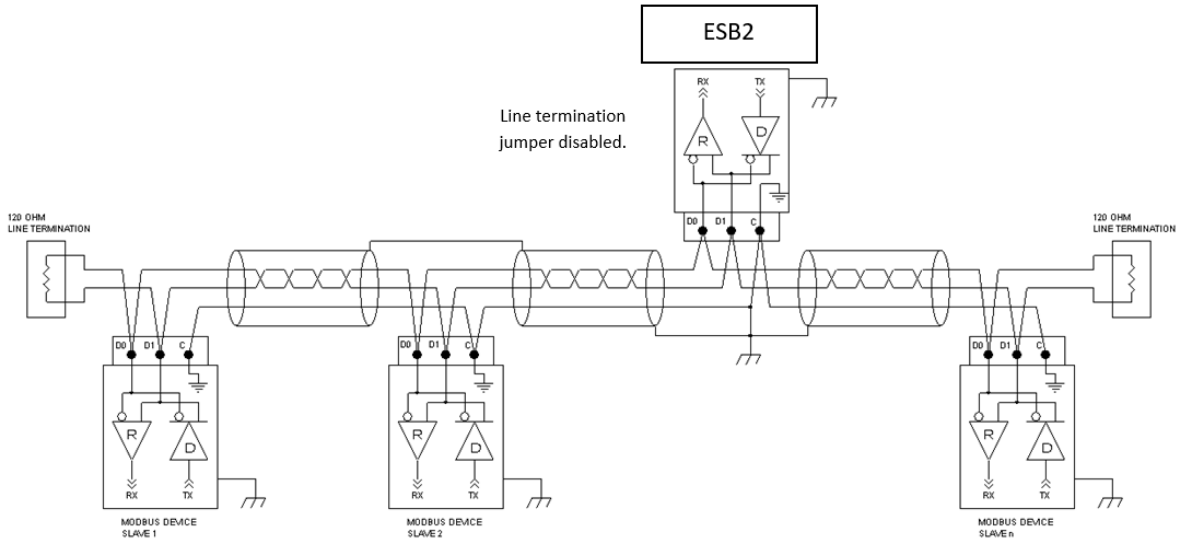
3.5.1.3 Line Termination

1. At each end of the Modbus cable network daisy-chain, connect a 1/2W 120-ohm resistor across the D0 (-) and D1 (+) conductors.
2. These resistors can be installed either directly into the connectors at the far-end Modbus devices, or on the end of a length of cable that is connected to the connectors at the far-end Modbus devices. This decision may depend on the customer's preference or on the available area around the Modbus connector at the far-end Modbus device.
3. If the TELSEC ESB2 is at one end of the Modbus cable network, do not install a termination resistor on this end of the cable network. Instead, move jumper J8 to the 1-2 position for COM1 (or J11 for COM2) which enables the built-in line termination resistor. If the TELSEC ESB2 is in the middle of the Modbus cable network, then move J8 for COM1 (J11 for COM2) to the 2-3 position. This will disable the built-in line termination resistor. See Figure 10 above and Figure 11 below. No more than two (2) line terminations are allowed per Modbus cable network.

3.5.1.4 Line Polarization

When there is no data activity on an RS-485 balanced pair, the lines are not driven and, thus susceptible to external noise or interference. To ensure the receiver line stays in a constant state, when no data signal is present, some devices need to bias the network. Polarization of the pair must be implemented **at one location for the entire Serial Bus**. Generally, this point is at the polling device. Other devices must not implement any polarization. Refer to the manual of the device you are monitoring to determine if line biasing is required. Move jumpers J9 and J10 for COM1 (or J12 &J13 for COM2) on the ESB2 Modbus board to position 1 & 2 (left) to enable line polarization. Refer to Figure 10 above.

Drawing A - shows the TELSEC® ESB2 in the center of the chain.



Drawing B - shows the ESB2 on the end of the chain.

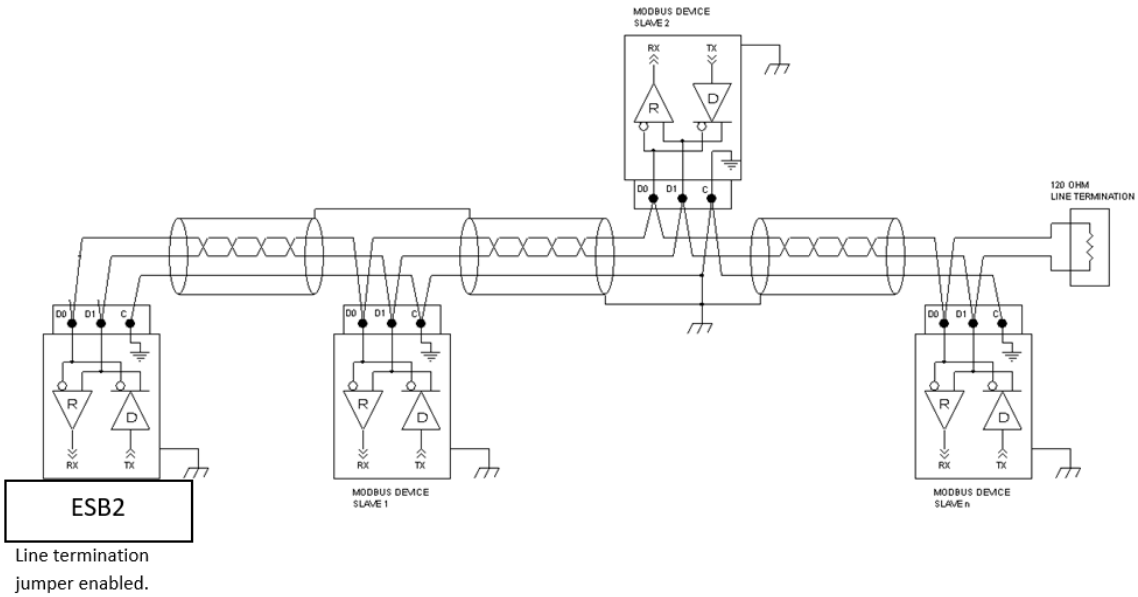


Figure 11 - Typical Modbus Networks

4 Touchscreen Operation

The ESB2 display is a color touchscreen designed to make reviewing status and alarming quick and easy without having to log into the systems web server for local interrogation. In addition to current status of all monitoring and control points, the display will provide the user with the ability to graph trend data, make simple setpoint changes and configure remote communications. The touchscreen can have passcodes enabled for the four main sections which are Status, Alarms, Programs and Settings. Each section has a separate passcode that can be enabled or disabled.

The display has a screensaver mode where the backlighting will turn off after no activity. Touch the screen to see the main start page. From this page you can select one of the options from the bottom menu or click on the alarm bell to see active alarms. The alarm bell will be red when active alarms are present and will be greyed out when there are no active alarm conditions.

Figure 12 - Touchscreen Start Page

4.1 Status Menu

Click on the status menu to see the available options. The display will have buttons for Analog Inputs, Digital Inputs, Digital Outputs, Analog Outputs, Modbus (is being used), Equation Variables and Equation Status.

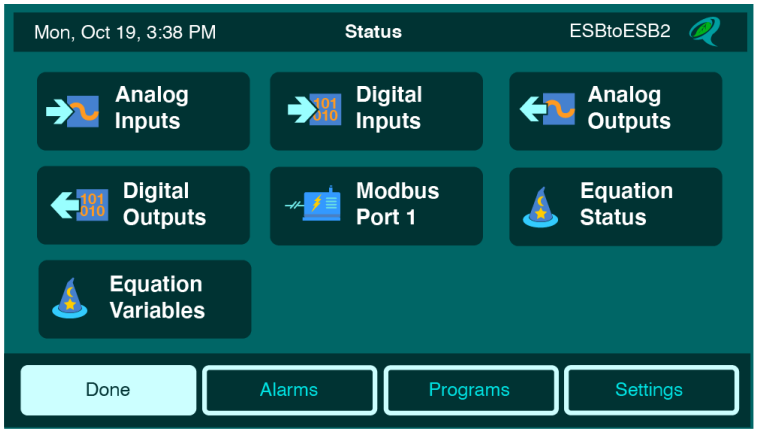


Figure 13 - Touchscreen Status Options

Select the desired option to see the current status of points. For example, clicking on Analog Inputs will display a table with the current status of all inputs. Use the Next and Back buttons to navigate through the available pages of data. Press the Done button to return to Status Screen Options.

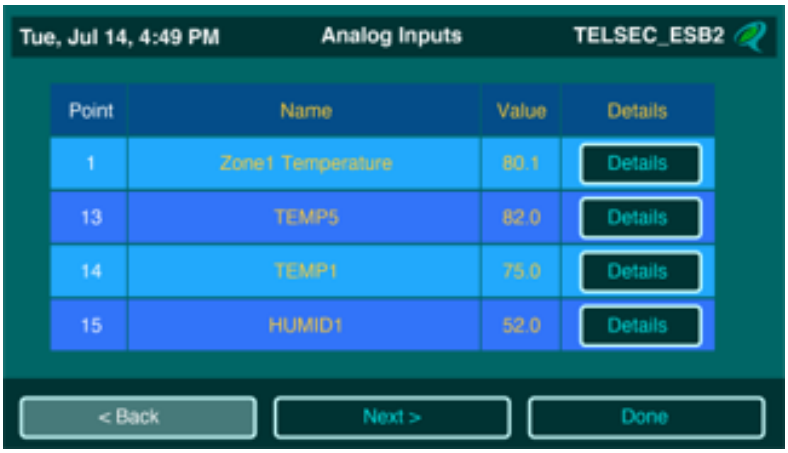


Figure 14 - Analog Input Status

4.1.1 Detailed Status

Some of status pages will have options for additional point detail. For example, next to each Analog Input will be a "Details" button. Pressing that button will provide options for viewing additional information such as Trend Data or Alarm Data.

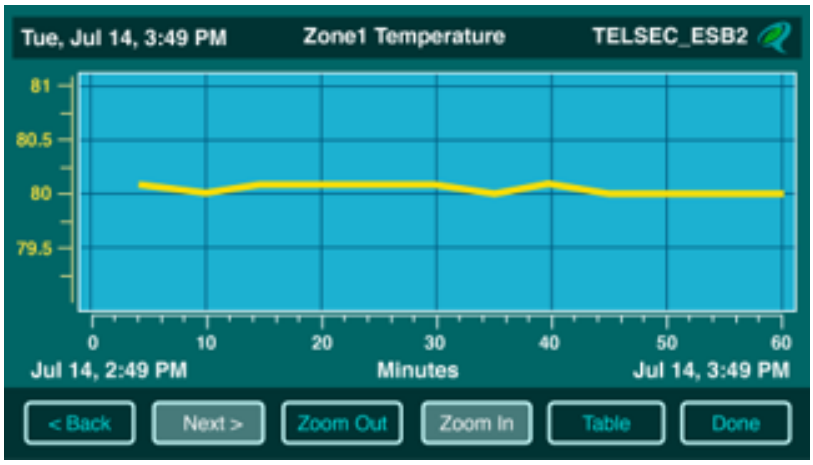


Figure 15 - Input Log Graph

Figure 15 shows the input data for the last hour. Clicking on the zoom In and Zoom Out buttons will change the time interval on the horizontal axis. Ise the Next> and < Back buttons to navigate through the pages of data.

4.2 Alarm Status

Touching the Alarms button from the main menu will bring up the Alarms menu. Here you can select the Active Alarms or Alarm History which will show all alarms in the system. Touching a specific alarm will show a list of all occurrences of that alarm. The menu along the bottom will have an "All" option to return the full alarm log.

You can also select Input or Modbus alarms which will only display alarms for those types. This is helpful in troubleshooting when you are dealing with a specific device. THE Alarms Status will show the current status of all defined alarms, whether the alarm is active or not and the date/time when it transitioned to its current status.

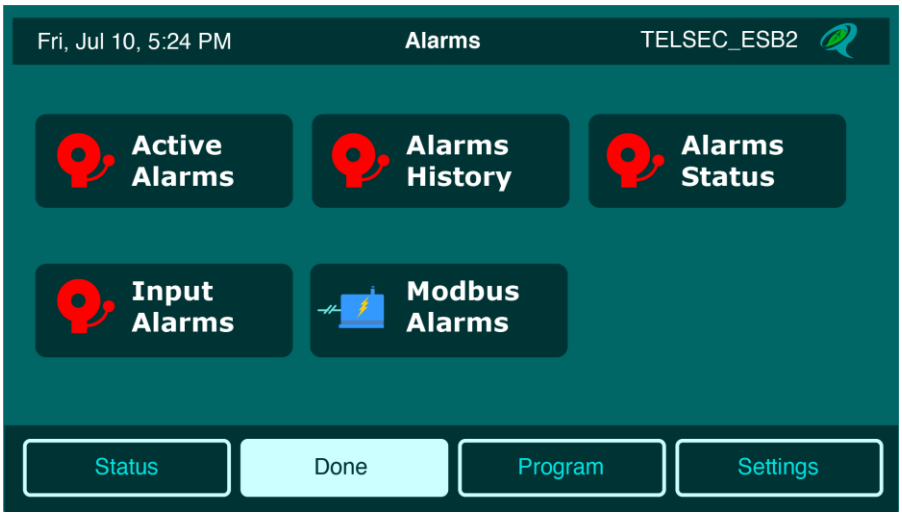
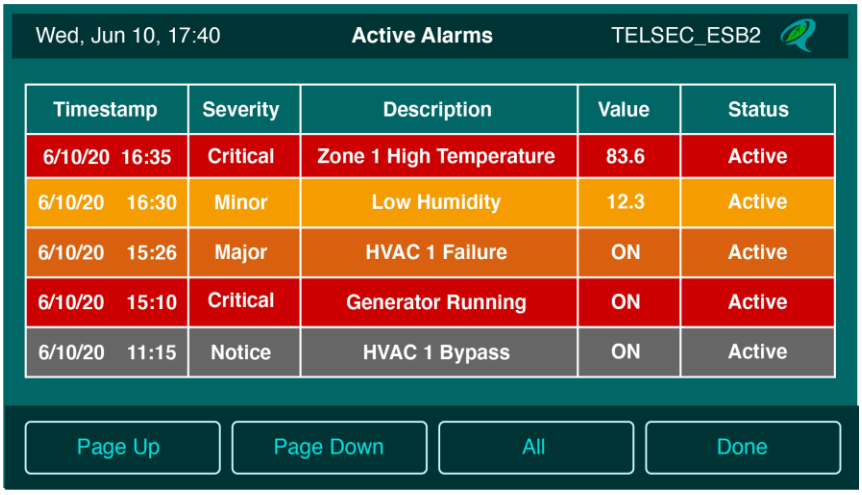


Figure 16 - Touchscreen Alarm Status Options

Alarm status pages for active and historical alarms will show the date/time the event occurred along with the severity of the alarm, description, sensor value and current status of Active or cleared.



Timestamp	Severity	Description	Value	Status
6/10/20 16:35	Critical	Zone 1 High Temperature	83.6	Active
6/10/20 16:30	Minor	Low Humidity	12.3	Active
6/10/20 15:26	Major	HVAC 1 Failure	ON	Active
6/10/20 15:10	Critical	Generator Running	ON	Active
6/10/20 11:15	Notice	HVAC 1 Bypass	ON	Active

Figure 17 - Touchscreen Active Alarms

4.3 Program Menu

The program screen will allow the user to change setpoints and perform bypasses of Outputs and Inputs. It is recommended that this section has the passcode protection turned on after the initial setup to prevent unauthorized use.

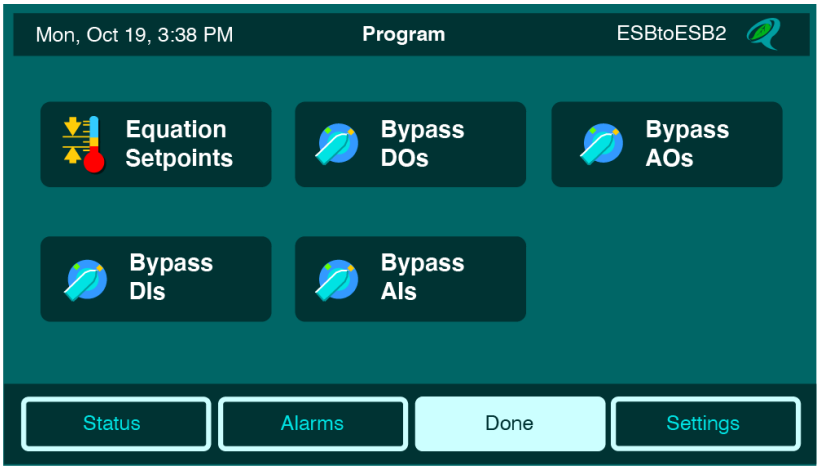


Figure 18 - Touchscreen Program Menu

Changing a setpoint is how a local user can affect how the ESB2 is controlling equipment. For example, changing the cooling setpoint from 80 to 78 degrees will lower the operating temperature in the building.

Bypassing on the other hand will override the current operation and force an output or input to a particular value. For example, use the Bypass Digital Output function to bypass an output ON or OFF for a period of time or indefinitely (forever). When a bypass is in effect, the system cannot change the value. Timed bypasses will timeout and go back to automatic, but Forever bypass must be set back to automatic manually.

Inputs work the same way where a bypass will override the current reading and will display the value entered when doing the bypass. This function is very helpful when testing control or alarm logic to ensure the system performs properly when the value of an input is greater than or less than a particular value. I.e., bypassing the zone temperature to a high value to ensure the HVAC turns on and the alarm strategy goes active with the correct severity etc.

4.4 Settings Menu

The settings menu allows the user to do the initial communications settings, set the system clock and set up passcodes for the menu sections. In addition, you can review system diagnostic information and perform operating system updates.

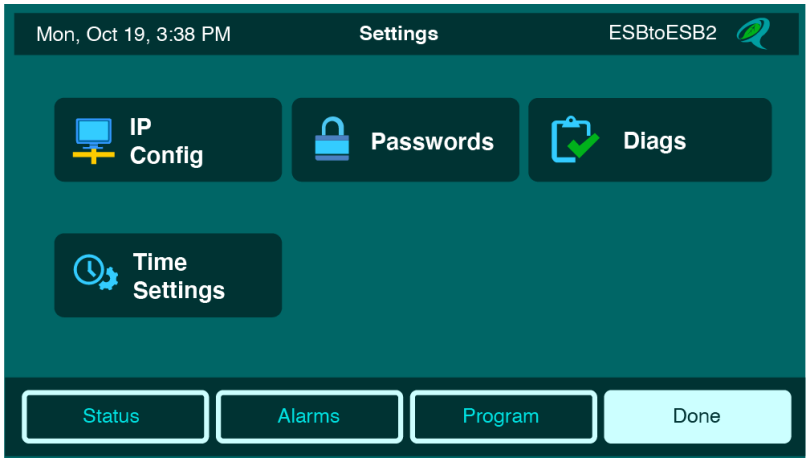


Figure 19 - Touchscreen Settings Menu

4.4.1 IP Config

The IP Config button will show the current Ethernet settings in the ESB2. Touch the configure button on the bottom menu to walk through the steps to configure the IP settings. Select Next to move to the next page after making changes or to accept the settings presently on the page. At the end of the process, the ESB2 will confirm you want to make the settings and then proceed after you acknowledge.

4.4.2 Passwords

This section will allow the user to enter a four-digit code for each menu section. Starting with the Status Menu, the ESB2 will allow the user to enter the desired code or blank out the current code by using the delete key. Touch the done key to move to the next menu option or press the Next key to move to the next entry without making changes.

4.4.3 Time Settings

This menu option is used to either manually set the system clock or to enter an NTP server for automatic updates from a time server. Select this option and follow the prompts to set the system clock.

4.4.4 Diags

This section has the following options:

- About – Shows the current firmware version and allows the user to check for updates. Note the update server must be programmed into the system first from either the web server or the IP config from the Touchscreen.
- Syslink Status – shows detailed communications information for the communications between the Processor board and the Main IO board.
- Module Status – Shows the communication status of all Quest expansion boards. See section 6.4.4

- Ping Gateway – Select this option to test communications between the ESB2 and the programmed gateway IP address.
- Ping Trap Servers – Select this option to test communications between the ESB2 and the programmed trap server IP addresses. The system will attempt to do an ICMP ping to each programmed address.
- Ping DNS – Select this option to test communications between the ESB2 and the programmed DNS server IP address to verify communications path.

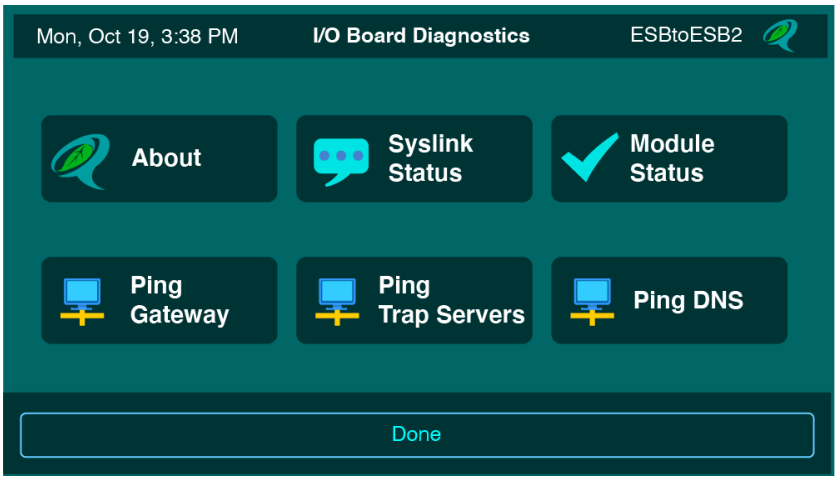


Figure 20 - Touchscreen Diagnostic Options

5 Communicating with the ESB2

Connect a standard Ethernet cable between the TELSEC® ESB2 and your local Ethernet switch or router. You can also direct connect to the system with your PC using a crossover Ethernet cable and setting a static IP address on your PC. The ESB2 has a default IP address of 192.168.1.31. Use this address to log into the unit and then change the IP address to a new permanent setting. Alternatively, the IP address can be changed using the front panel touchscreen. Log into the system select the menu option then the settings menu and finally the IP Config option. Follow the prompts to change the IP address.

5.1 Connecting to the System

The built-in web server uses a *username* and *password* combination to authenticate a user and allow access. No information will be shown without proper authorization. The default setting uses HTTP protocol with a username of **Admin** and password of **admin.** The username and password are case sensitive so make sure your caps lock key is turned off. The default password along with the five available User level passwords can be changed once you log in and go to the Settings/HTTP User Administration web page. Keep track of your password because there is no way to access the system without a valid password. The Admin user password can be reset through the Shell Command interface (see section 10) provided the shell level password is known. The alternative would be to cold start the system to return to default settings, which will remove all application programming and return the IP address as well as the password back to the original default settings. See section 14.8 for instruction.

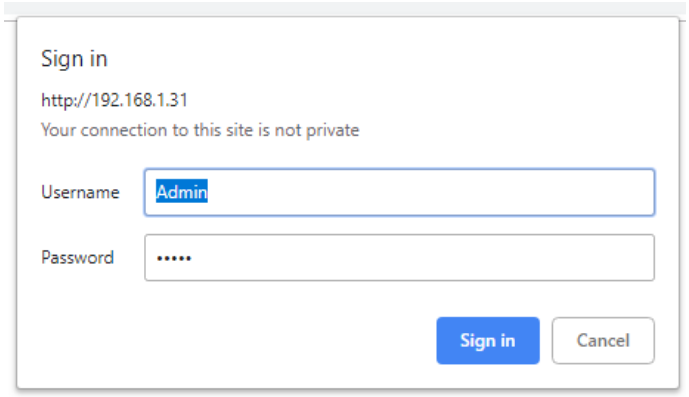


Figure 21 - Log in Screen

5.2 Web Page Navigation

All ESB2 web pages will have a top header section that shows the Site Name, the system's date and time, Alarm bell icon and a menu ribbon with the available choices for navigation. The choices presented on the menu ribbon will be dependent on the access level of the password used. Clicking on the alarm icon will navigate directly to the active alarms page.

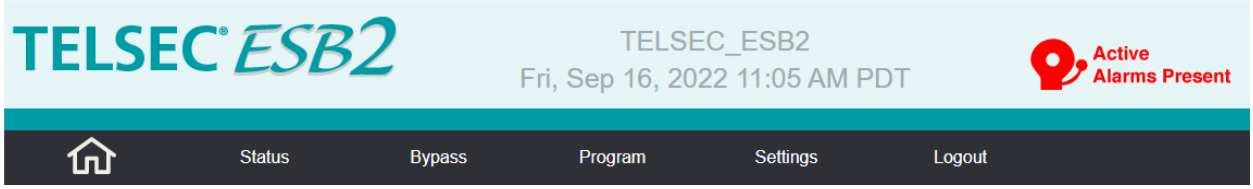


Figure 22 - Web Page Header Section

5.2.1 The Home Page

After you log in with your username and password, the ESB2 will display the Home page. This page can also be displayed if you click on the home icon on the menu ribbon. The home page will show the site information field and any Active Alarm condition.

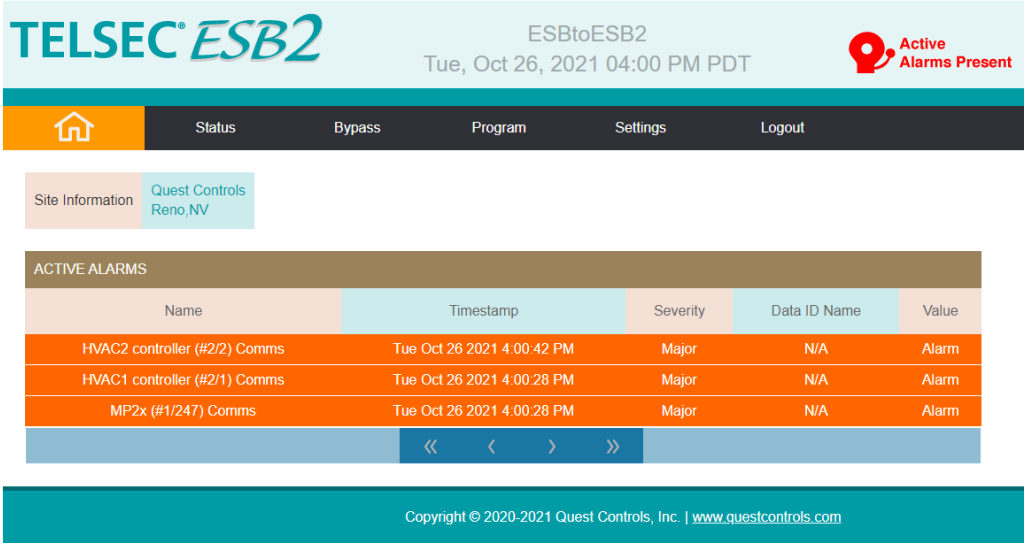


Figure 23 - The TELSEC® ESB2 Home Page

6 System Status

Hovering your mouse pointer on the Status option will provide the choices of Input Status, Modbus status (if defined), Output Status, Control Point Status, Qmbus Module Status, Alarm Status, Active Alarms, Historical Alarms and Log Data. This section of the manual will review each one of these options and the available functions.

6.1 Input Status

The input status page will show the current value of the universal inputs that are programmed in the system. The status page allows you to see all inputs in one group or separate groups for digital inputs and analog inputs. Click on the Tab header to expand or contract each section for viewing.

6.1.1 Digital Inputs

Digital inputs will show the input number followed by the defined name of the point and the current value. The page will show the ACCUMULATED ON which represents the entire time the input has been active (ON), this timer can be cleared. The INTERVAL ON timer shows how long this input has been active this time, when the input turns OFF this value will be held. When the input re-activates, this timer will clear to zero and start counting. The INTERVAL OFF timer reacts similarly to the ON timer but only counts when the input is off or inactive. The EVENT counter counts one for each complete cycle. I.e., the input going on then off counts as one cycle or event. The EVENT TIMER begins counting from the time the input is programmed or from the time the input was last cleared.

The system will keep track of the number of alarms that are active for the input and display that as a hyper link. Clicking on the link will bring up a window with a table of active alarms for the point.

The last column on the digital input page will show the defined "normal state" of the input. The purpose of this column is to indicate how the point is defined to make trouble shooting easier without having to navigate to the input definition page. The following will be displayed:

DRYNO = A normally open input wired between input and ground.
DRYNC = A normally closed point wired between input and ground
WETNO = A normally open input wired between the 24V terminal (or external voltage max 60 VDC) and the input terminal.
WETNC = A normally closed input wired between the 24V terminal (or external voltage max 60 VDC) and the input terminal.

ALL INPUTS

DIGITAL INPUTS

Point #	Name	Value	Related Alarms	Accumulated On	Interval On	Interval Off	Event Count	Event Timer	Normal State
5	HVAC1 Lockout	OFF	0	000:00:00:00	000:00:00:00	000:00:11:48	0	000:00:11:48	DRYNO
7	HVAC2 Lockout	OFF	0	000:00:00:00	000:00:00:00	000:00:11:48	0	000:00:11:48	DRYNO
8	Smoke Detector	OFF	0	000:00:00:00	000:00:00:00	000:00:11:48	0	000:00:11:48	DRYNO

ANALOG INPUTS

Figure 24 - Digital Input Status

6.1.2 Analog Inputs

Analog input status will show the input number, the programmed name for the point and the current value of the input in the defined engineering units. The related alarms column will indicate the number of active alarms currently present that reference the input. A number greater than 0 will be a hyperlink that will bring up a window showing a table of active alarms for the point.

ANALOG INPUTS

Point #	Name	Value	Related Alarms
1	Zone1 Temperature	60.1	1
13	TEMP5	82.0	1
14	TEMP1	75.0	1
15	HUMID1	52.1	1
16	TEMP2	86.0	1
17	TEMP3	76.0	1
18	TEMP4	77.0	1
26	HVAC1TMP	59.0	0
27	HVAC2TMP	64.0	0
29	TEMP6	85.0	1
55	GEN1FUEL	82.0	0
56	GEN2FUEL	42.0	1
71	LIGHTLV1	0.0	0

Figure 25 - Analog Input Status

6.2 Modbus Status

The ESB2 supports two serial ports that both can be configured as Modbus networks. The ESB2 acts as the primary system which polls Modbus devices. The Status menu will show Modbus (port #) where # equals the physical port being used for Modbus. If both ports are used for Modbus, then there will be two entries on the menu. Note Modbus will not show as a status option unless the ports are defined for Modbus.

Selecting the Modbus port status will display a status screen of available Modbus points. The points will be grouped based upon the device they are associated with, and the name of the device will be displayed in the header for each window. Clicking on the expand window icon will display the points for each device. The points will be listed with their defined name, the current value based on the point definition and a column showing any related alarms. Related alarms are Alarm Points that reference the Modbus point and are in an active alarm condition. Clicking on the related alarms number will display a popup window with a listing of the active alarms.

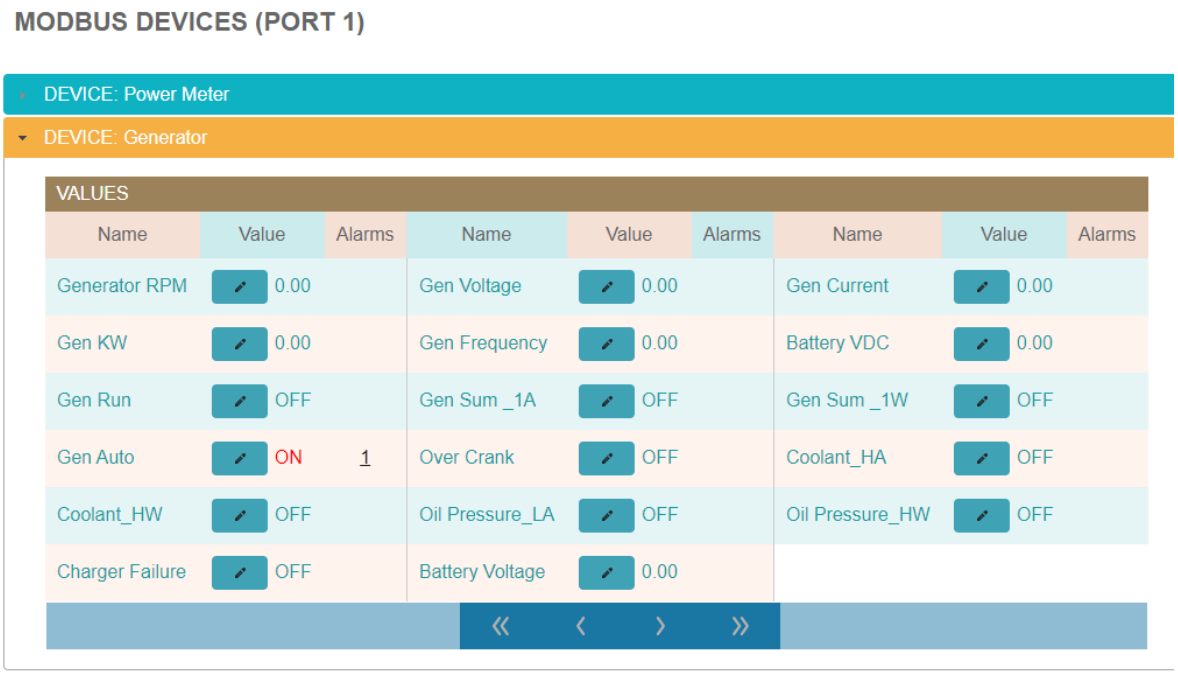


Figure 26 - Modus Input Status

Modbus points that are writable, will be displayed with a write icon provided the user has program level authorization. Clicking on this icon will bring up a window to allow the user to change the value. Enter the value in the converted engineering units that are being displayed. The ESB2 will use the point definition to determine the correct value to be sent to the device. Note: the actual ability to change a value is dependent on the device being monitored. Even though the ESB2 attempts to write the data, the device may reject the change.

6.3 Output Status

The output status page shows the current status of the defined digital outputs, digital output timers and analog outputs.

6.3.1 Digital Output Status

Each output will be shown with the defined name, Action which is either automatic (AUTO) or BYPASSED if a user has bypassed the output (Settings/Bypass). The Energized field shows the physical status of the output of either being energized or de-energized. The state field shows whether the point is Active or Inactive based upon the user defined program. The last column shows “restarting” if the processor board can’t talk with the output board or will have two dashes indicating normal operation.

DIGITAL OUTPUT STATES

Point #	Name	Action	Energized	State	Status
1	HVAC1_Fan	AUTO	DE-ENERGIZED	INACTIVE	--
2	HVAC1_Y1	AUTO	DE-ENERGIZED	INACTIVE	--
3	HVAC1_Cmpr	AUTO	DE-ENERGIZED	INACTIVE	--
4	HVAC1_Heat	AUTO	DE-ENERGIZED	INACTIVE	--
5	HVAC2_Fan	AUTO	ENERGIZED	ACTIVE	--
6	HVAC2_Y1	AUTO	DE-ENERGIZED	INACTIVE	--
7	HVAC2_Cmpr	AUTO	DE-ENERGIZED	INACTIVE	--
8	HVAC2_Heat	AUTO	DE-ENERGIZED	INACTIVE	--

Figure 27 - Digital Output Status

6.3.2 Digital Output Timers

Just like digital inputs, the digital outputs have timers associated with them that are automatically created when the point is defined. The page will show the ACCUMULATED ON which represents the entire time the output has been active (ON), this timer can be cleared. The INTERVAL ON timer shows how long this output has been active this time, when the output turns OFF this value will be held. When the output re-activates, this timer will clear to zero and start counting. The INTERVAL OFF timer reacts similarly to the ON timer but only counts when the output is off or inactive. The EVENT counter counts one for each complete cycle. I.e. an output going on then off counts as one cycle or event. The EVENT TIMER begins counting from the time the output is programmed or from the time it was last cleared.

DIGITAL OUTPUT STATES

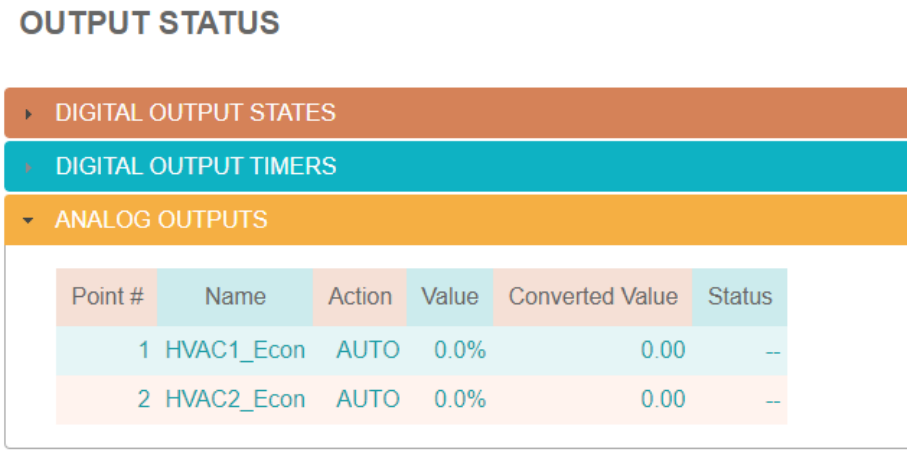
DIGITAL OUTPUT TIMERS

Point #	Name	State	Accumulated On	Interval On	Interval Off	Event Count	Event Timer
1	HVAC1_Fan	OFF	003:02:08:40	000:23:53:43	000:00:06:52	2	003:02:45:43
2	HVAC1_Y1	OFF	003:02:20:42	000:23:59:39	000:00:06:52	2	003:02:45:43
3	HVAC1_Cmpr	OFF	003:02:20:42	000:23:59:39	000:00:06:52	2	003:02:45:43
4	HVAC1_Heat	OFF	000:00:00:00	000:00:00:00	003:02:45:43	0	003:02:45:43
5	HVAC2_Fan	ON	003:02:43:28	000:00:06:20	000:00:00:32	2	003:02:45:43
6	HVAC2_Y1	OFF	003:02:22:56	001:00:00:50	000:00:06:52	2	003:02:45:43
7	HVAC2_Cmpr	OFF	003:02:22:56	001:00:00:50	000:00:06:52	2	003:02:45:43
8	HVAC2_Heat	OFF	000:00:00:00	000:00:00:00	003:02:45:43	0	003:02:45:43

Figure 28 - Digital Output Timers

6.3.3 Analog Outputs

Each output will be shown with the defined name, Action which is either automatic (AUTO) or BYPASSED if a user has bypassed the output (Settings/Bypass). The value field will show the percentage output from 0-100% where 100% equals the maximum output of 10V or 20mA. The Converted Value will show the status of the output based on the scale definition for the output. For example, if the output was controlling a variable speed motor, it can show the equivalent RPM value proportional to the output percentage. The last column shows "restarting" if the processor board cannot talk with the output board or will have two dashes indicating normal operation.



Point #	Name	Action	Value	Converted Value	Status
1	HVAC1_Econ	AUTO	0.0%	0.00	--
2	HVAC2_Econ	AUTO	0.0%	0.00	--

Figure 29 - Analog Output Status

6.4 Control Point Status

Control points are used in equations and consist of three groups which are Setpoints, Variables and Equations. This section will provide status information specific to each control point type. See section 11.1 for information on defining Equation control points.

6.4.1 Setpoints

Setpoints are used in control equations as comparisons to determine when to do an action. An example of a setpoint could be the value at which the air conditioner should be turned on. The setpoints section will display the defined ID used in equations for reference, the 32-character display name, the Initial Value (original defined value) of the setpoint and the current value (Value column) of the setpoint. The current value will differ from the initial value if there is a control equation written to automatically change the value. The last column is the Change option that will be displayed if the user has the program level access. Clicking on the change icon will allow the user to change the initial value of the setpoint.

CONTROL POINT STATUS

SETPOINTS

ID	Name	Initial value	Value	Change
zone_spt	Cooling Setpoint	78	78	
heat_rqrd	Heating Setpiont	55	55	
cool_dif	Cooling Differential	2	2	
deadband	Cooling Deadband	2	2	
heat_dif	Heating Differential	2	2	
humidity	Humidity Setpoint	65	65	
osa_lvl1	Econ Level 1	66	66	
osa_lvl2	Econ Level 2	69	69	
cool3dif	Cool Stage 3 diff	2	2	
fan_mode	Lead Fan Mode	1	1	
fan_delay	Fan Delay	120	120	
hightemp	High Temp Alarm	85	85	

VARIABLES

EQUATIONS

Figure 30 - Setpoint Status

6.4.2 Variables

Memory variable points are used in control equations to store values. An example of using a memory variable would be to store the controlling setpoint for a Lead and Lag HVAC system. A control equation would then write the values to be used based upon which HVAC unit is the lead and lag system. The Variable section will display the defined ID used in equations for reference, the 32-character display name, the Initial Value (original defined value) of the setpoint and the current value (Value column) of the variable. The current value will differ from the initial value if there is a control equation written to automatically change the value. Unlike setpoints, memory variables cannot be changed through the status page.

SETPOINTS

VARIABLES

ID	Name	Initial value	Value
ctrltemp	Control Temperature	0	-155
leadhvac	Lead HVAC Unit	2	2
hvac1_cl	HVAC 1 Cool	78	80
hvac2_cl	HVAC 2 Cool	78	78
hvac1_ht	HVAC 1 Heat	65	53
hvac2_ht	HVAC 2 Heat	65	55
mix1_spt	HVAC 1 Mixed Air	50	50
mix1_delta	HVAC 1 Mixed Delta	5	5
mix2_spt	HVAC 2 Mixed Air	50	50
mix2_delta	HVAC 2 Mixed Delta	5	5
change_prcnt	Economizer Percent Change	5	5

Figure 31 - Memory Variable Status

6.4.3 Equations

Equations are the programs that have been created to provide control logic in the system. Each equation will have an ID name used for reference in programming along with a 32-character display name, the initial value of the equation, the current value and the state of the equation. The value of an equation is an integer number that can be used to tell another equation to do something. For example, an equation used for cooling can look at the status of another equation used for de-humidification and turn on the cooling if the humidity is high in the facility. The State column indicates Enable or Disabled. All equations are typically enabled meaning they are available to run. A disabled equation will occur when the program (equations) written needs to prevent an equation from running.

EQUATIONS

ID	Name	Initial value	Value	State
calctemp	Calc Temps	0	0	Enabled
leadsw	Lead Switch	0	0	Enabled
leadstage	Lead Stage	0	1	Enabled
hvac1_fn	HVAC 1 Fan	0	0	Enabled
hvac1comp	HVAC 1 Cool	0	0	Enabled
osatemp	OSA air temp	0	1	Enabled
hvac1heat	HVAC 1 Heat	0	4	Enabled
hvac2_fn	HVAC 2 Fan	0	1	Enabled
hvac2comp	HVAC 2 Cool	0	0	Enabled
hvac2heat	HVAC 2 Heat	0	4	Enabled

Figure 32 - Equation Status

6.4.4 Qmbus Module Status

This page will show all defined expansion boards connected to the system along with the communication status. The base system contains sixteen inputs, sixteen digital outputs and two analog outputs, but can be expanded by adding Quest expansion modules. The system will support sixteen universal input modules with sixteen inputs each, Sixteen digital output modules with eight outputs each and sixteen analog output modules with four outputs each.

The page will show the type of board, note Syslink is the base IO module, the board address and status. The status will either be OK meaning it is communicating or FAILED indicated the module is not communicating. The last two columns provide statistics to the number of good and failed poll requests since the system was reset or power failed.

MODULE STATUS

Type	Address	Comm status	Good polls	Failed polls
Syslink	N/A	OK	6790	1
UIB	96 (0x60)	FAILED	2961	34

Figure 33 - Quest Bus Module Status

6.5 Alarm Status

The Alarm Status page will provide the current status of all the defined alarms. Each alarm point will be listed with the defined name, the name of the point being referenced in the alarm, a timestamp when the alarm point last transitioned to its current value and the current value of the point.

ALARMS STATUS

FILTER

Category: All Name: Data ID Name:

ALARMS

Name	Data ID Name	Timestamp	Value
Power Meter (#1/1) Comms	N/A	Thu Mar 21 2019 5:16:11 PM	OFF
Generator (#1/100) Comms	N/A	Thu Mar 21 2019 5:16:11 PM	OFF
Alarm Test	N/A	Mon Mar 04 2019 3:45:40 PM	OFF
Tower - Photocell Failure	Tower - Photocell Failure	N/A	OFF
Tower - Marker Light Alarm	Tower - Marker Light Fail Alarm	N/A	OFF
Tower - Beacon Fail - Night mode	Tower - Night Beacon Light Fail	Fri Mar 01 2019 10:45:37 AM	ON (alarm)
Tower - Beacon Fail Day Mode	Tower - Day Beacon Light Failure	N/A	OFF
Tower Light for alrm qualifier	Tower - Light Status	Fri Mar 22 2019 4:00:03 PM	ON

« ‹ › »

Figure 34 - Alarm Status Page

Use the forward, back, beginning and end icons to navigate through the alarms. The page supports filtering alarms based on category from the dropdown box such as Inputs or Modbus, or by word matching of names. Use the Name field for the alarm name and the Data ID Name field to search by an input name. For example, you can type Tower in the name field to get a list of all alarms that have the word "Tower" in the name.

6.6 Active Alarm Log

The Active Alarms page will display any alarm that is currently active in the system. The page allows for searching based on severity level and by date. The alarms displayed can also be downloaded to a CSV file by clicking on the Download To CSV button.

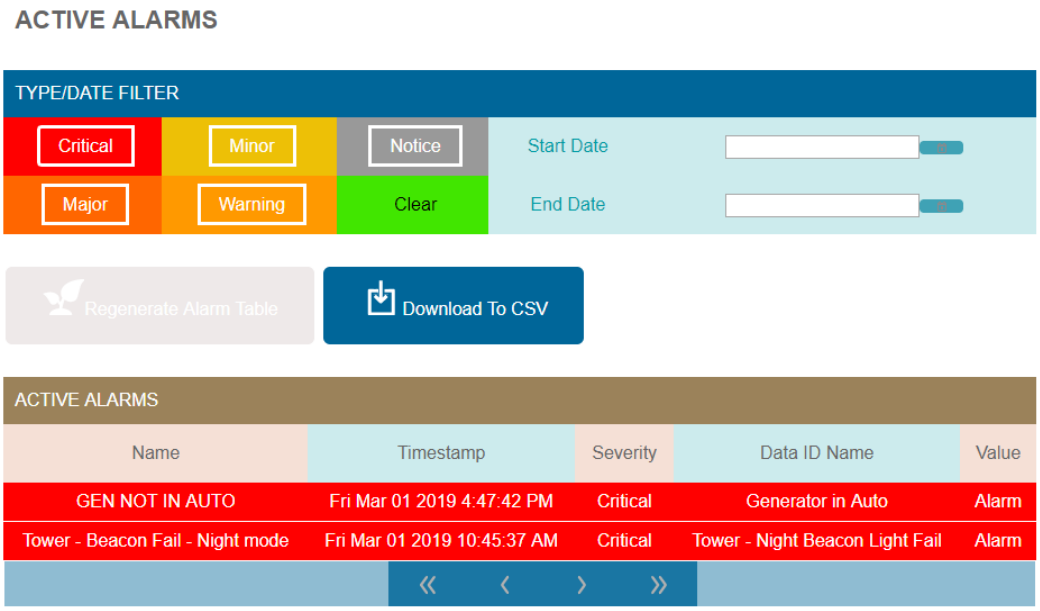
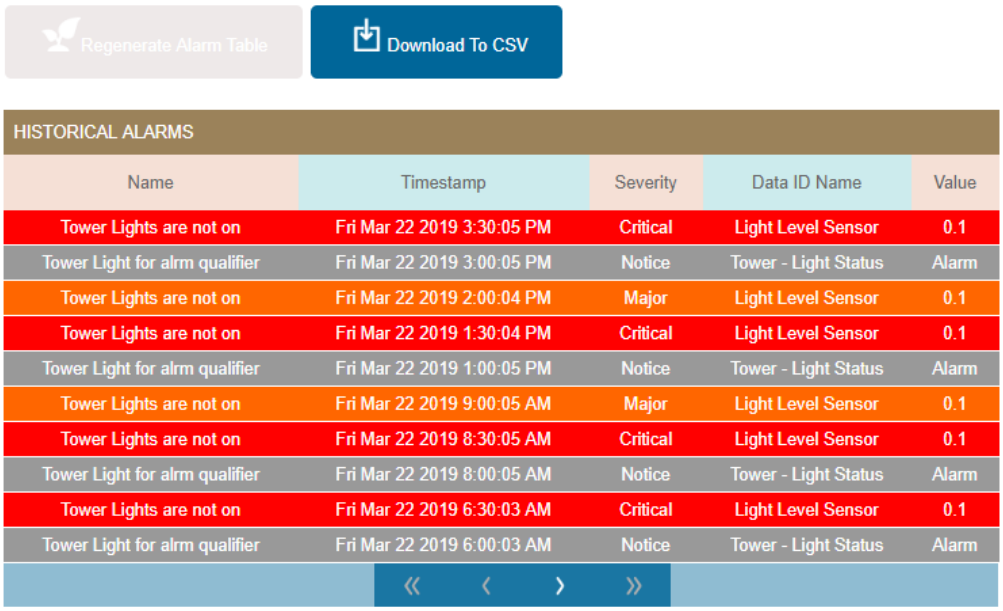


Figure 35 - Active Alarms Page

6.7 Historical Alarm Log

The Historical Alarm page will show the alarm history of the stored alarms. Use the back, next, beginning and ending button on the bottom of the table to navigate through the list of alarms. The Download To CSV button will create a CSV file and download it to your computer. See section 12 for information on the file format.



Name	Timestamp	Severity	Data ID Name	Value
Tower Lights are not on	Fri Mar 22 2019 3:30:05 PM	Critical	Light Level Sensor	0.1
Tower Light for alrm qualifier	Fri Mar 22 2019 3:00:05 PM	Notice	Tower - Light Status	Alarm
Tower Lights are not on	Fri Mar 22 2019 2:00:04 PM	Major	Light Level Sensor	0.1
Tower Lights are not on	Fri Mar 22 2019 1:30:04 PM	Critical	Light Level Sensor	0.1
Tower Light for alrm qualifier	Fri Mar 22 2019 1:00:05 PM	Notice	Tower - Light Status	Alarm
Tower Lights are not on	Fri Mar 22 2019 9:00:05 AM	Major	Light Level Sensor	0.1
Tower Lights are not on	Fri Mar 22 2019 8:30:05 AM	Critical	Light Level Sensor	0.1
Tower Light for alrm qualifier	Fri Mar 22 2019 8:00:05 AM	Notice	Tower - Light Status	Alarm
Tower Lights are not on	Fri Mar 22 2019 6:30:03 AM	Critical	Light Level Sensor	0.1
Tower Light for alrm qualifier	Fri Mar 22 2019 6:00:03 AM	Notice	Tower - Light Status	Alarm

Figure 36 - Historical Alarm Page

6.7.1 Filtering Alarm Data

Use the TYPE/DATE FILTER functions to filter the alarm severity and the Start/Stop Date fields to select the specific desired ranges. The page has additional filters for the alarm category (All, Inputs, Modbus by device) and the ability to word match the name of the alarm and/or the data point being referenced in the alarm. Press the Regenerate Alarm Table button once you create your search criteria. For example, a search of all Critical Tower Alarms:

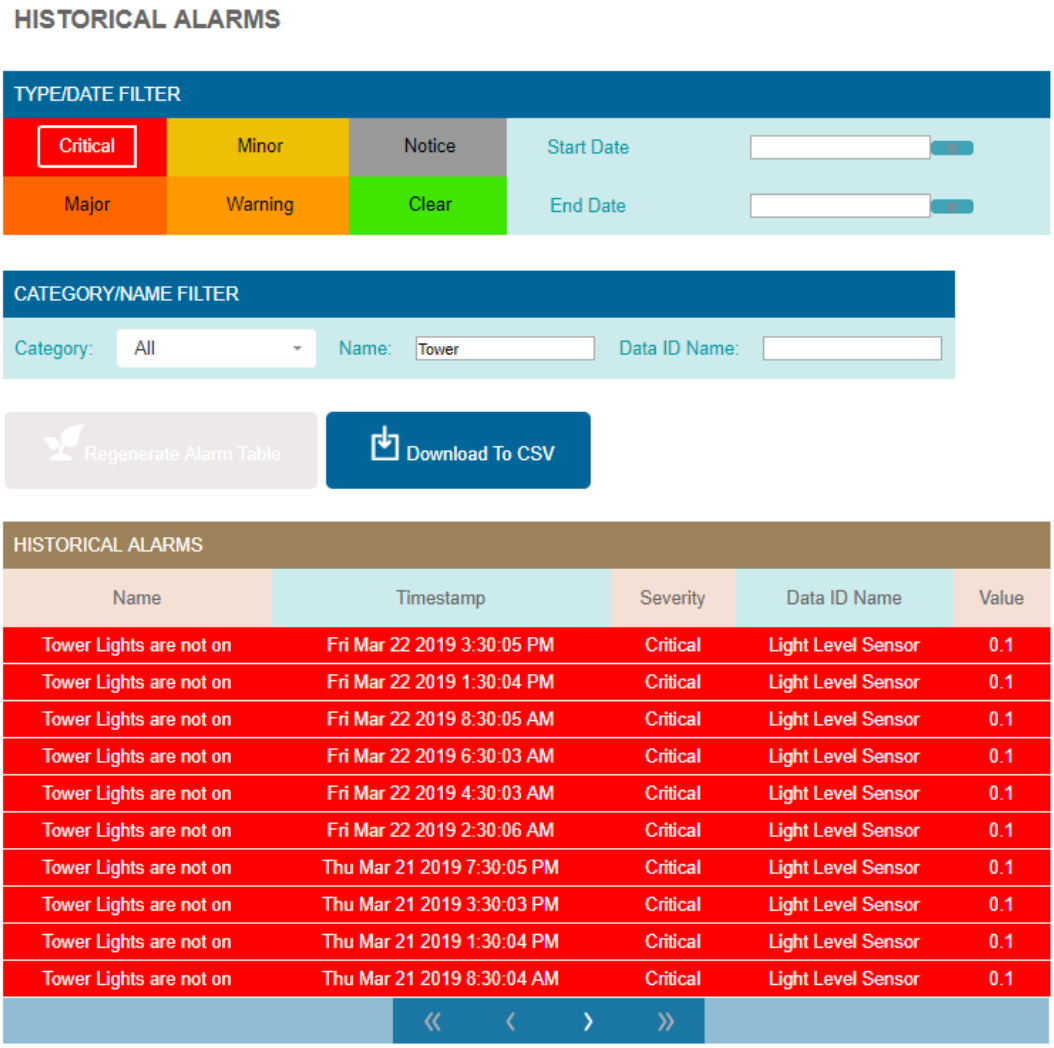


Figure 37 - Alarm History Search Results

Press the “Download to CSV” button if you want to save a copy of this search to your computer in a comma separated values (CSV) file format.

6.8 Log Data

The ESB2 will plot a graph of up to four points at a time. The page controls will allow the user to select from Analog Inputs, Digital Inputs, Digital Outputs and Modbus. Any combination of point types can be charted at one time. Select the point(s) to be graphed and then select the date range for the data. If no date range is specified, then the system will display the most recent data for the points when clicking on the Regenerate Graph button. When there are a lot of points present, you can type a name or partial name in the “Point name match string” field. The system will only show points that contain those characters. For example, you could type Zone in the point name match field and the system will only display analog inputs with value of Zone in the name.

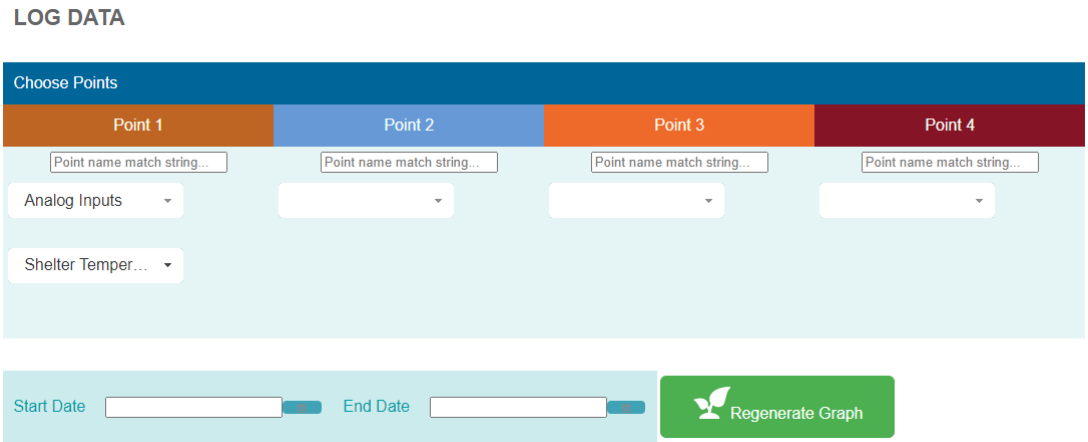


Figure 38 - Select Data Point to be Graphed.

The system will display the graph with a default view of the last 24 hours. Use the page buttons and zoom in and out icons to view the data in your web browser. The ESB2 will display a Download to CSV button which will download the data in a CSV file for viewing in another program. Note the download will be the most current 4095 entries per point when no time range was selected. If a large time range was selected, then the system will return the last 4095 entries per point for the range.

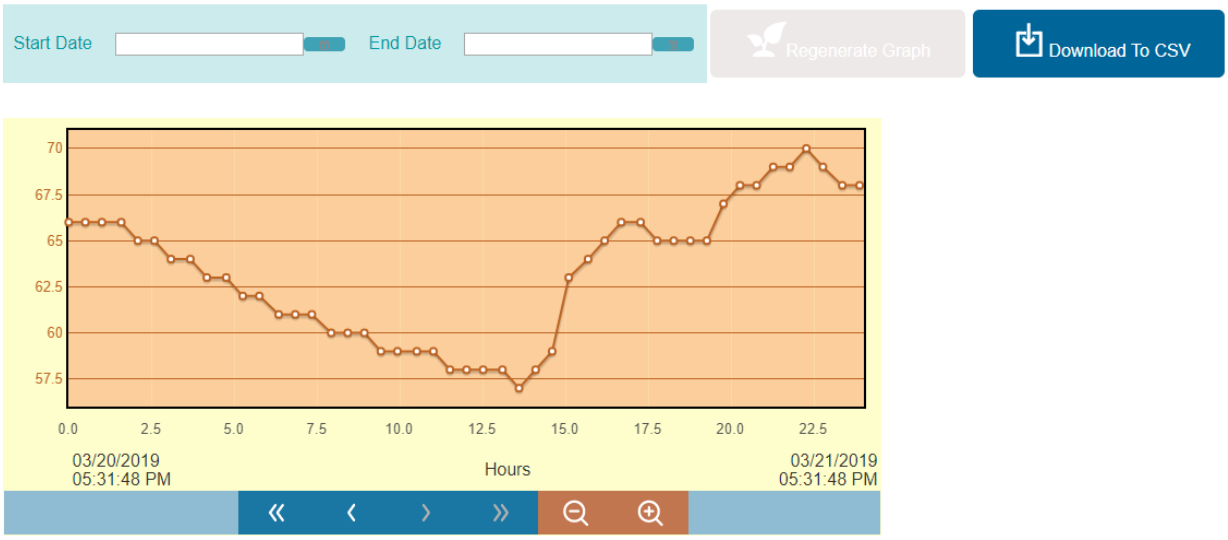


Figure 39 - Graph of Selected Data Points

7 Bypassing

The Bypass page is used to bypass the outputs on the ESB2 and to override the current status of directly wired inputs and Modbus points. Click on Bypass from the main menu bar and select Bypass Points

7.1 Bypass Inputs

Directly wired inputs and any defined Modbus point can have their current value overridden by clicking on the “Add new input bypass” button. This will bring up the following dialogue:

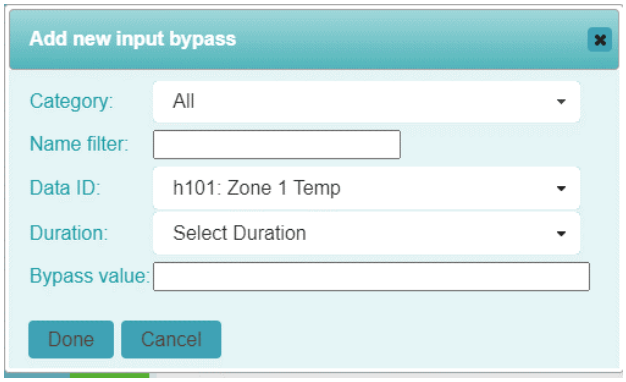


Figure 40 - Bypass Inputs

The category field is used to filter the points in the Data ID dropdown. The choice of ALL will show all defined inputs and Modbus points. You can select from All, Universal inputs or the Modbus device name. The Name Filter is used to reduce the number of available points in the drop down. Type the name or partial name of the point to bypass and the system will only show points that contain those characters. Select the point to be bypassed in the Data ID field and then select the duration. The bypass value field will have the option of ON or OFF for binary points or will allow the user to enter a numeric value for analog sensors. Press the Done button to complete the bypass. The point in bypass will then appear on the bypass page. Clicking on the Auto button for the input will return the point to automatic readings.

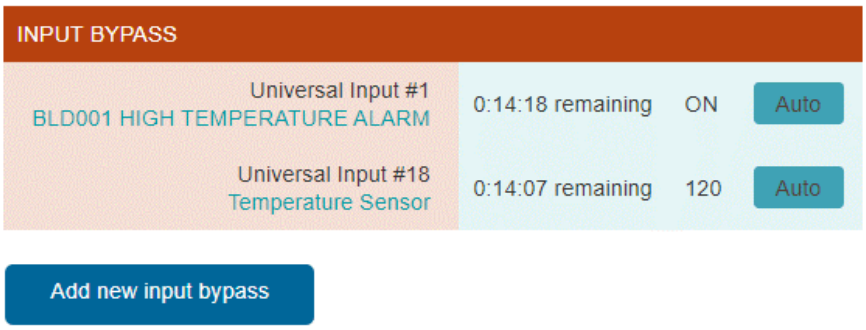


Figure 41 - Input Bypass Status

7.2 Bypassing Digital Outputs

The bypass output function is used to turn on (activate) or off (Inactive) the outputs. Select a time duration from the dropdown or the word FOREVER and then select the desired action of ON or OFF. Outputs bypassed for a time period will return to automatic mode once the time elapses. Outputs bypassed FOREVER must be manually returned to automatic by selecting Auto.

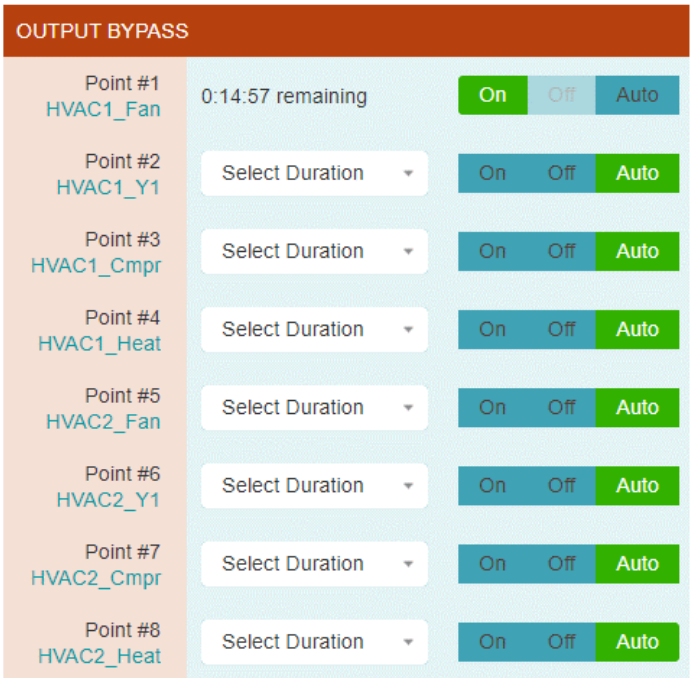


Figure 42 - Bypassing ESB2 Outputs

7.3 Bypass Analog Outputs

Bypass Analog Outputs is used to override the control equations and force the output to a desired percentage output for a period of time or for Forever. Time period bypasses will clear automatically or can be manually cleared back to automatic. Bypassing for Forever must be cleared manually.

Select the duration time from the dropdown and then enter the bypass value from 0 to 100%.

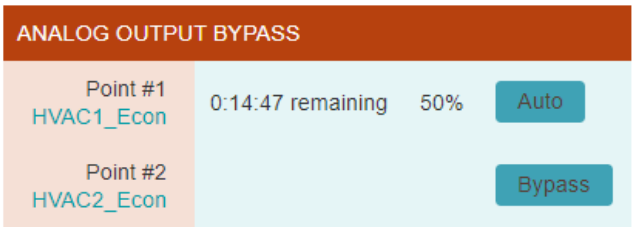


Figure 43 - Bypass Analog Outputs

8 Configuring the ESB2

Configuring the ESB2 consists of setting up the basic functions and features that will be used with your application. Items such as the network settings, system clock, serial ports, network protocols and alarm dispatch need to be configured so the system knows what and how to operate when you program the system for control, monitoring and alarming.

8.1 System Setup

Select Settings from the main menu and choose General. The ESB2 will display a list of options for configuration with the first one being Site Parameters. Click on each accordion header to modify the settings for each section and press the Save Settings button to write the changes to non-volatile flash memory. Save Settings will cause the system to save the changes and reboot to come up with the new values. You can navigate through each tab section on this page, make changes and then press the Save button to save all changes at once. **Changes will not be saved if you navigate from this page without pressing the "Save Settings" button.**

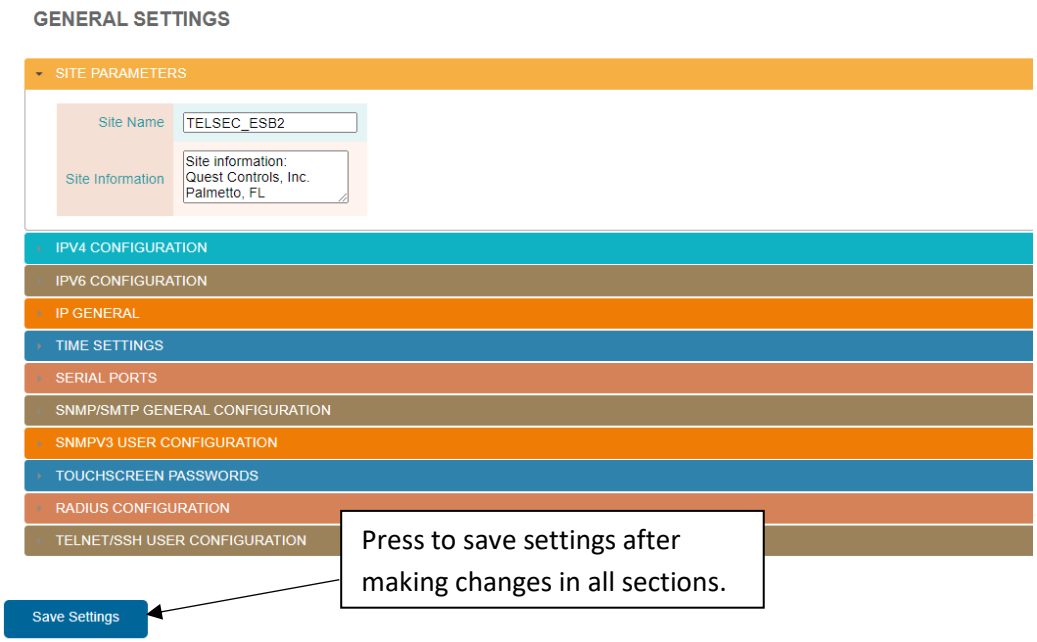


Figure 44 - Web Server Setup

8.1.1 Site Parameters

The Site name is displayed in the header of web pages and is used when sending Emails and/or SNMP traps. Enter the desired name to reference the site where the ESB2 is located. **Note spaces are not allowed and will be converted to underscores**. The Site Information section is to store unique information about the site such as the address to the facility or contact information.

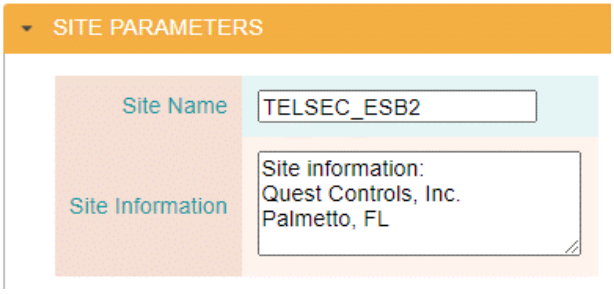


Figure 45 - Site Name and Information

8.1.2 IPV4 Configuration

The ESB2 supports dual stack for IPv4 and IPv6 or you can disable either IPv4 or IPv6. The default setting is for IPv4 to be enabled and IPv6 disabled. The system will not allow a user to disable both. Use this section to enable/disable IPV4 protocol and to configure an IP address for the unit. The system supports DHCP or Static. If Static is chosen, then enter the desired IP address, Subnet mask and Gateway address.

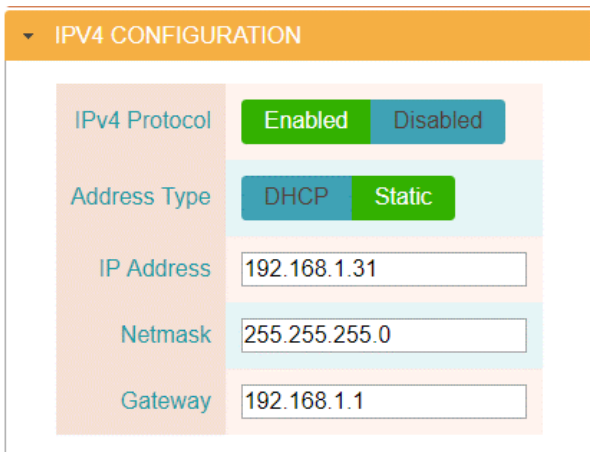


Figure 46 - IP address and settings

8.1.3 IPV6 Configuration

The ESB2 supports either dual stack for IPv4 and IPv6 or you can enable IPv6 and disable IPv4. Use this section to enable IPv6 and configure the network address using either Auto configuration or manual configuration. If Auto is chosen, then the system will display the address in the IP address field after pressing the Save Settings button and the system goes through its restart. For manual configuration, enter the IP address with the prefix and the gateway address without a prefix.

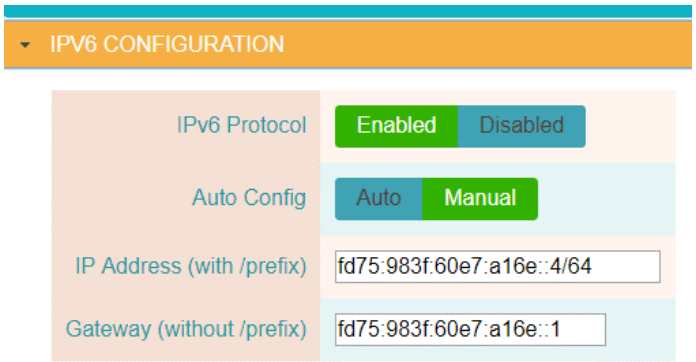


Figure 47 - IPv6 settings

8.1.4 IP General

This section is used to enter the DNS server IP addresses, change the default IP ports, enable/disable SSL and configure the location of where the update server is located. The Telnet and SSH port are used for Transaction Language 1 (TL1) alarming and access to the management shell see section 8.1.11 for configuring usernames and passwords. The TL1 port is a read only access that uses Telnet and does not require a log in sequence. Changing the port number to 0 for Telnet, TL1 Port or SSH will disable access to that function.

When selecting SSL, the system will allow you to copy/paste the contents of your own PEM data file if you have one. If no PEM data is entered, then the system will create a self-certification. Self-certification will allow HTTPS connection, but your browser will generate an error notice and ask you if you want to continue. The update server section is used to identify where to look for updates when a user presses the Check and Install Updates button found on the Settings/Advanced page. It is recommended that this field is not changed unless you know the IP address and path for update files.

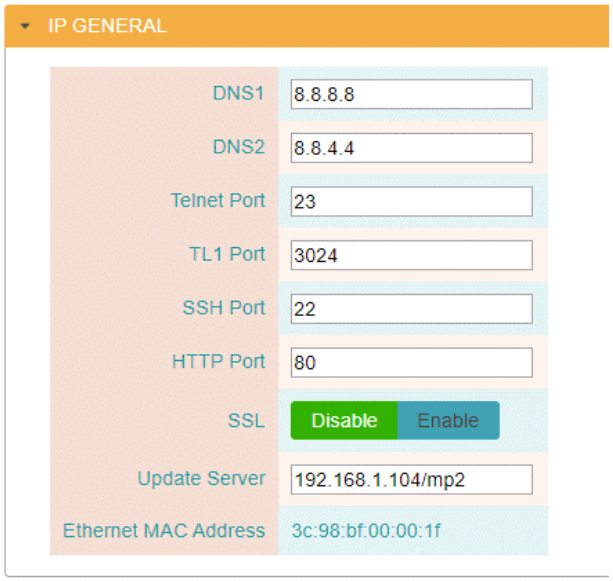


Figure 48 - Additional network settings

8.1.5 Time Settings

Use this section to either set the system clock manually or to turn on automatic time updates using an NTP Server. This page will also allow you to enter a time zone for the system and configure daylight savings time. In manual mode, the web page will allow you to set the time of the system manually by clicking on the time field. This will bring up a dialog box for setting the time which will include a button to set the time using time on your computer.

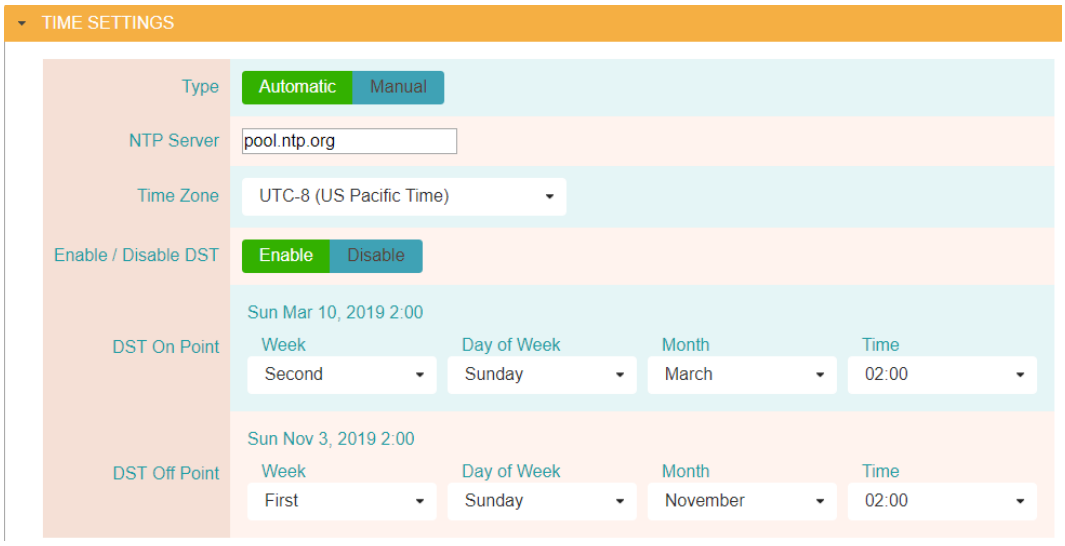


Figure 49 - System clock settings

8.1.6 Serial Ports

The ESB2 has two RS485 serial ports that can be used for Modbus RTU. The Modbus option makes the ESB2 a Modbus poller to poll equipment that support Modbus RTU protocol. Use the dropdowns to configure the serial ports for baud rate, data bits, stop bits and parity options. You can Select an alarm severity and delay in seconds for alarming if communications are lost to remote devices. The Comm clear delay will clear the alarm condition when there have not been communications errors present for the delay time (seconds). This section will also allow the user the ability to change the alarm severity, delay time and clear time for the communication errors on the ESB2 expansion board network.

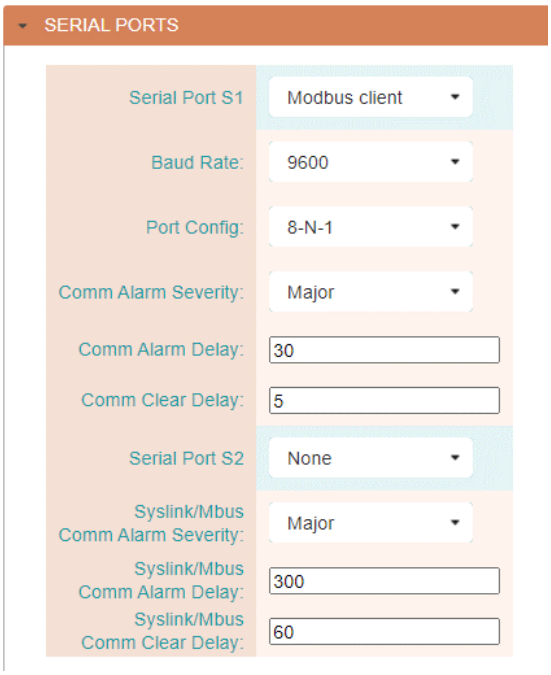


Figure 50 - Configure serial ports for Modbus

8.1.7 SNMP/SMTP General Configuration

Use this section to set up the common settings when using SNMP or SMTP (Email). When using SNMPv3, you can have SNMPv1 & v2c active by putting in a Read and Write community string. Leave these fields blank to disable v1 & v2c SET and GET operations. Contact Quest to obtain the MIB files for the ESB2. There is a specific ESB2 MIB for doing SET and GET commands along with Quest's standard Alarm MIB file for trap reception.

The SMTP setup is to allow the ESB2 to send text-based emails when an alarm occurs. Enter the SMTP server information, username and password if necessary, to allow the ESB2 to connect to the sever for sending email notifications. The email dispatch addresses, and alarm severity are configured in the Settings Alarm Dispatch page (section 8.2.2). The format of the email can be found in Appendix D – Email Alarm Format

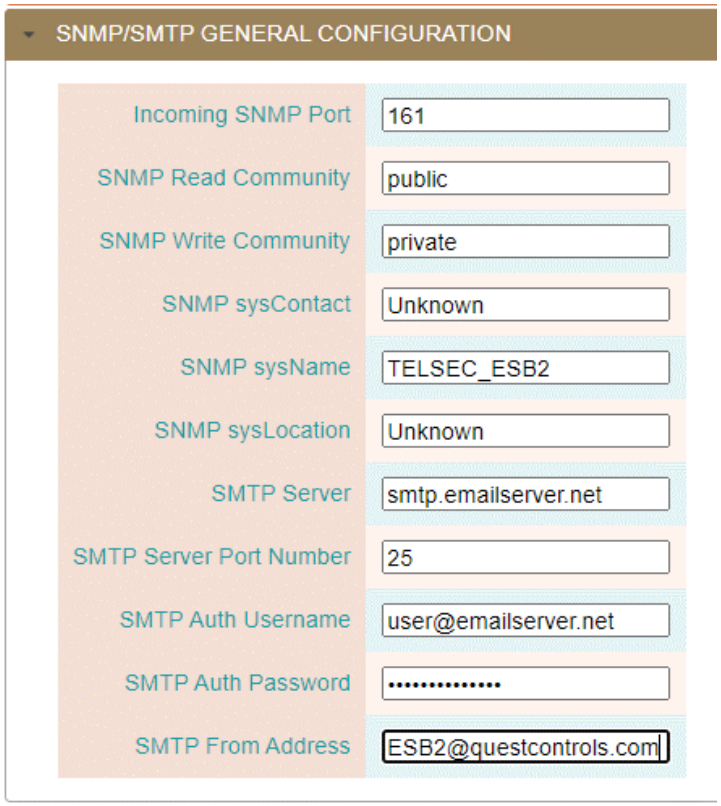


Figure 51 - SNMP and Email configuration

8.1.8 SNMPv3 User Configuration

Use this section to create up to four different users for SNMP Set/Get access to the system and sending traps if v3 traps are configured under the Settings/Dispatch page. The ESB2 supports authentication using MD5 or SHA and Privacy using DES or AES. Usernames and passwords are 8-32 characters in length and accept the characters a-z, A-Z and 0-9.

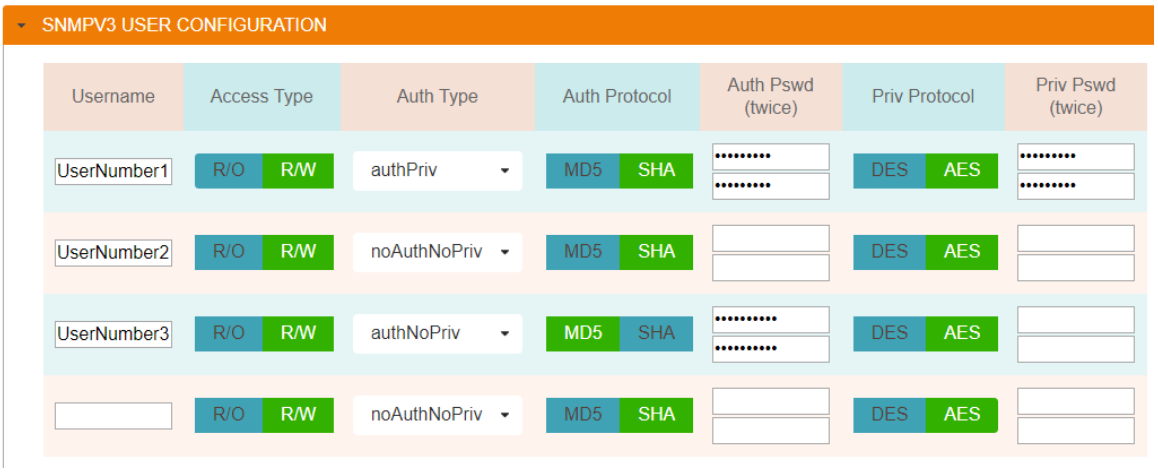


Figure 52 - SNMPv3 Username configuration

8.1.9 Touchscreen Passwords

This section will allow the user to set or remove passwords from the four menu sections on the front panel touchscreen. Enter the four-digit combination for each section or leave them blank if no passcodes are required.

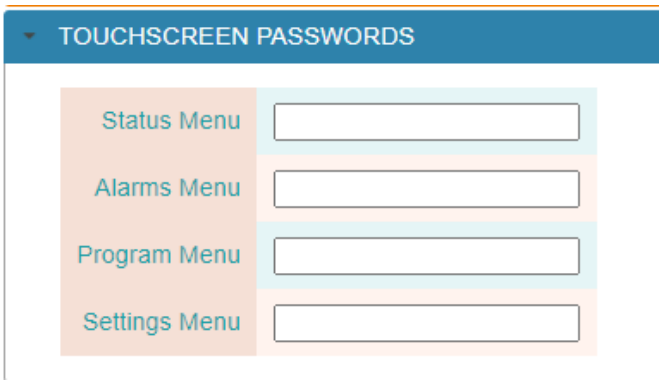


Figure 53 - Touchscreen Passcodes

8.1.10 RADIUS Configuration

If RADIUS is enabled, the ESB2 will contact the defined server when a user attempts to log into the system to determine if the username/password is valid. The ESB2 will grant access if the username/password is valid and will show the menu options based upon the user's access level defined in the RADIUS server. See Appendix B - RADIUS Server Configuration for the required setting when configuring the RADIUS server.

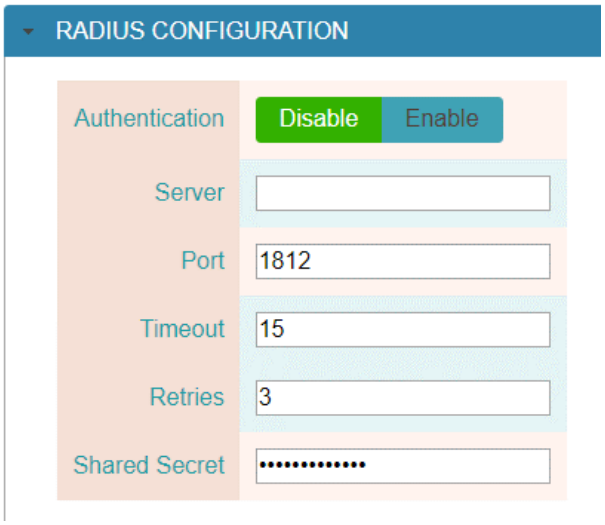


Figure 54 - RADUIS server interface setup

8.1.11 Telnet/SSH User Configuration

The ESB2 supports SSH and Telnet for access to the system to receive alarms via TL1 (Appendix E – TL1 Alarms and Commands) or to access the Shell console for managing the network settings of the ESB2 (section 10). The Telnet/SSH User Configuration section is used to set usernames, passwords, inactivity timeout delay and the access type for up to nine different users. The default username/password for

the “shell” level is Shelllevel/shelllevel and for TL1 access it is TL1level/tl1level. Change the default settings and or add additional users by entering the appropriate information and then select the access type. Press the save changes button to write the settings to flash. Usernames are 8-32 characters in length and can contain A-Z, a-z, 0-9 and special character _. Passwords are 8-32 characters in length and can contain A-Z, a-z, 0-9 and special characters _@#%*-

Figure 55 - Telnet/SSH User Configuration

8.2 Alarm Dispatch Settings

The Alarms Dispatch option under the Settings menu will display the current settings for both the SNMP dispatch and Email dispatching of alarms.

8.2.1 SNMP Dispatch

The ESB2 supports sending SNMP traps or Informs (supported in v2c or v3) to four trap servers. Enter the IP address of the trap server and select the SNMP version and either Trap or Inform. Some trap servers require a specific Community String to be used, but if not, use the value of Trap which is the default. For SNMPv3, you will need to use one of the defined Usernames that was created during the configuration of SNMPv3 (section 8.1.8). Press the “Save Settings” button to save your changes. **Note: if you are also using Email dispatching, then make those entries first prior to pressing the “Save Settings” button (section 8.2.2).**

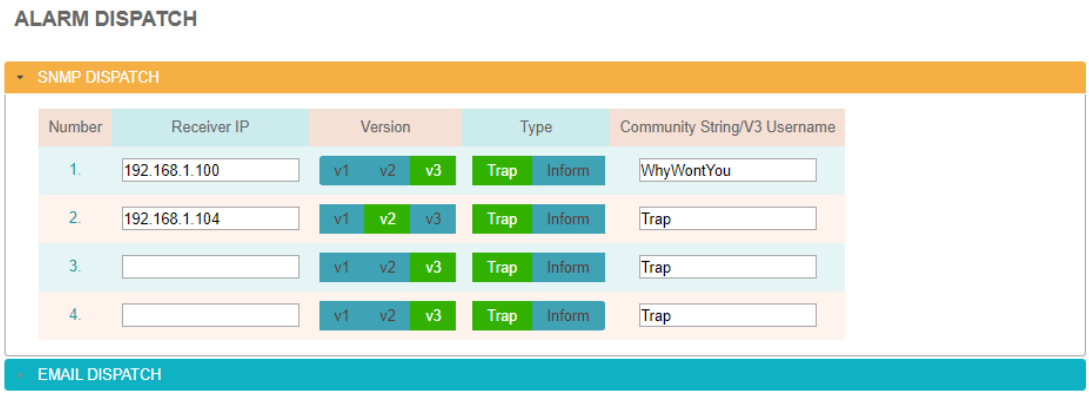


Figure 56 - SNMP Trap/Inform configuration

8.2.2 Email Dispatch

Click on the Email Dispatch expand icon to display the settings for creating email distribution. The Email server settings must be configured for email alarming to occur (See section 8.1.7). The ESB2 supports nine email addresses for sending alarms. Enter the users email address or distribution list address to be used. The subject field allows for a custom message to be included with the email. The system will create a subject line with the severity of the alarm followed by the text entered in the subject field. The last column allows for filtering of what type of alarms to be sent. The system will automatically send a clear email when the alarm condition clears. For example, if only Critical alarms are sent via email then the Clear for those critical alarms will be sent.

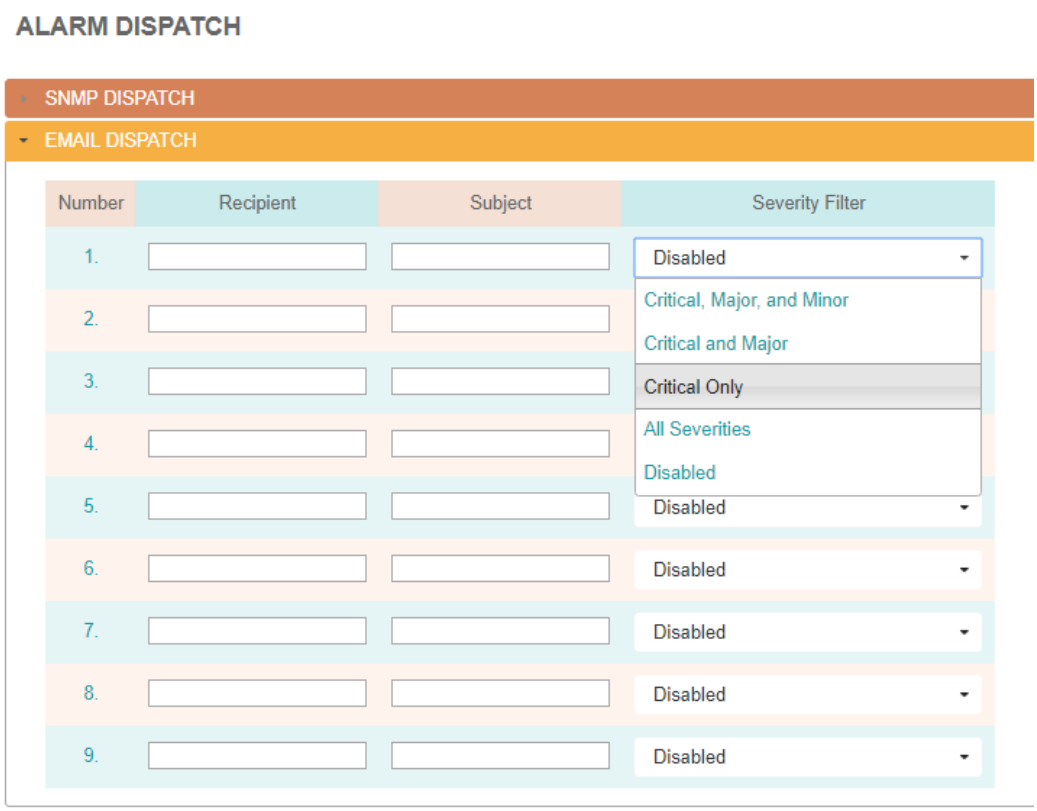


Figure 57 - Email alarm dispatch

Press the Save Settings button at the bottom of the page when completed. This will save any changes made to the email dispatch and changes made to the SNMP dispatch.

8.2.3 Heartbeat Function

The ESB2 can be enabled to send a heartbeat message based on a user defined interval. This is typically used to coordinate with a monitoring system platform to ensure the system is operational. Heartbeat messages are only sent via SNMP or TL1 over a Telnet/SSH connection. The heartbeat message is a user definable 32-character message. Enter the desired message and select the interval by entering in the number of seconds (1-3600) to wait between sending messages. A value of 0 will disable this function.

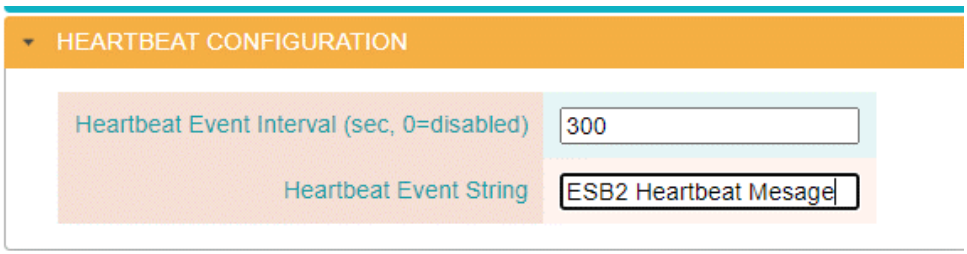


Figure 58 - Heartbeat configuration

8.2.4 Test Alarms

The Send Test Alarm button at the bottom of the page will cause a test trap to be sent to all the defined destinations along with a test email to the defined recipients. Any changes made to SNMP or Email dispatch must be saved first before the Test Alarm function is used. The Test alarm will have a critical severity with all the data fields showing TEST ALARM. The critical alarm will be followed by a clear alarm again with all the data fields showing TEST ALARM.

8.3 HTTP User Administration

This page sets up the six available usernames for the password and what areas the user can access. If the RADIUS server function is enabled, then this option does not affect user access. The default passwords are the same as the username, but in all lower case. No characters are displayed in the password field for security purposes. Passwords can be 5-32 characters in length and contain A-Z, a-z, 0-9 and special characters _@#%*-

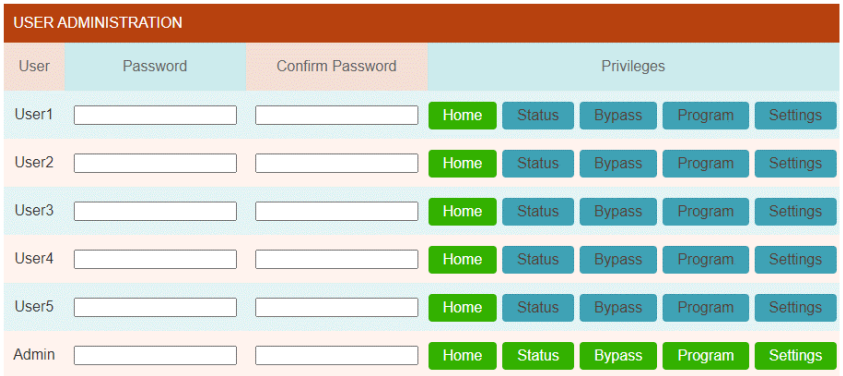


Figure 59 - Webserver password setup

8.4 Advanced Settings

8.4.1 System Version & Update

Choosing the Advanced settings will display the current version of the system's firmware. Selecting the Update button will cause the system to try and connect to an update server to look for upgrades. The IP address used for upgrades must be accessible from the network the ESB2 is on in order to perform the upgrade. The IP address for the upgrade server is configured in the IP General window. See section 8.1.4. Alternatively, the ESB2 can be upgraded locally using an SD card. Contact Quest Controls for upgrade information.

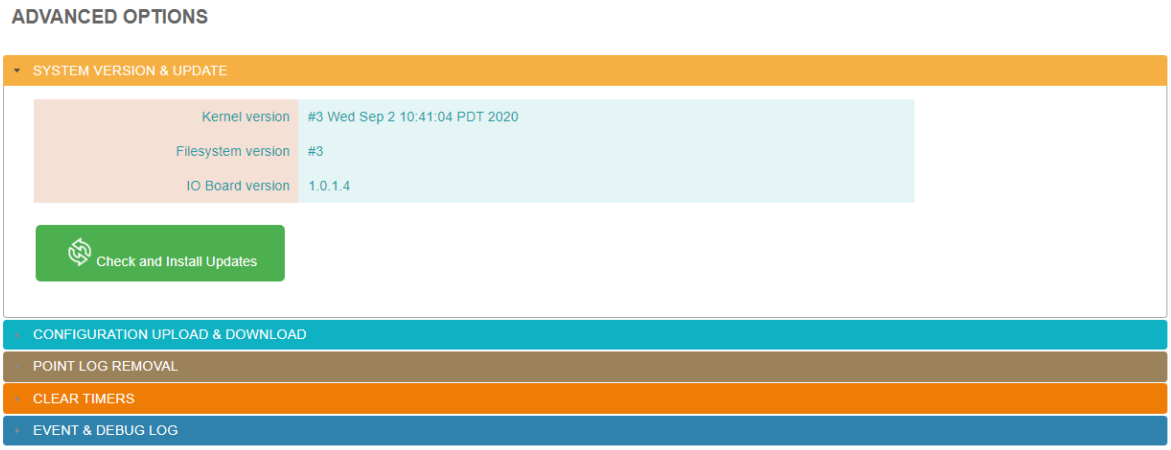


Figure 60 - Firmware version and update

8.4.2 Configuration Upload & Download

Select the expand arrow to display the option for downloading or uploading a configuration file to the system. The download option will download a text file to your web browser. The file will have all of the current settings configured in the system. This file can then be saved in case the site needs to be uploaded in the future. The Upload option will allow you to load a configuration file to the system. While the system will download all settings into a single file, it will accept uploading files with individual sections. This way you can upload only the sections requiring changes. Each section has a label in brackets [] to denote the section. The entire section must be loaded together to be successful.

First select the Choose File button to navigate and select a file from your computer. Then select the Upload button. The system will receive the file and then verify the information to ensure it conforms to the proper syntax. The system will then present a dialogue box displaying any errors with their line number or a notice of no errors. If there are no errors, then there will be a Save and Reboot button below the dialogue box. Pressing this button will cause the system to save the contents of the configuration file to flash and then restart with the new settings. This process will take approximately one minute.

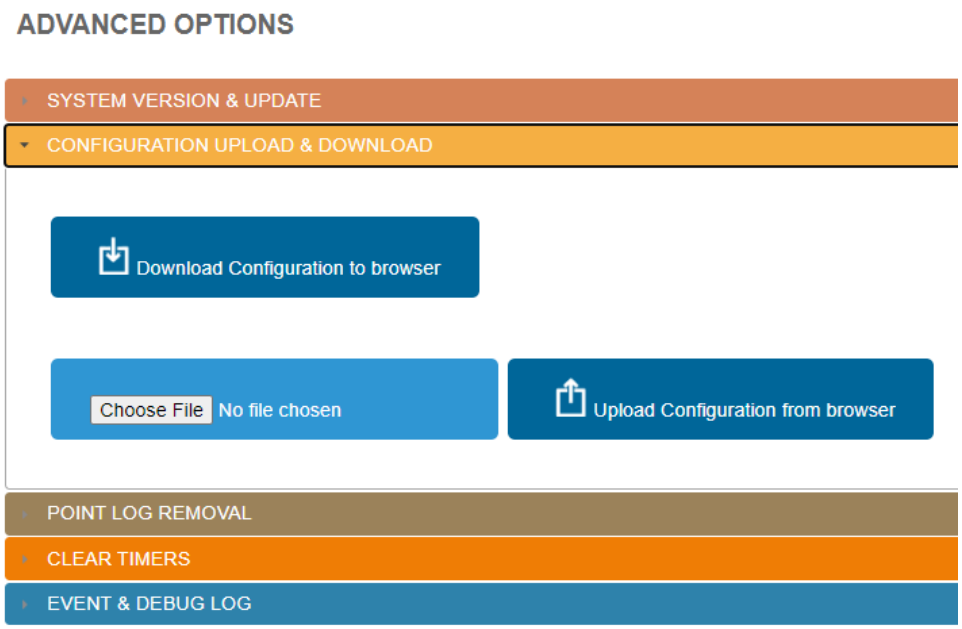


Figure 61 - Configuration file upload & download

8.4.3 Point Log Removal

This section will allow clearing of log data for a single point or erasing history log data for all the points. It is not necessary to periodically remove log data since the system will overwrite old data when the log is full. Typically, the manual clearing of logs might be done after the system is properly wired and configured to eliminate erroneous data that could have occurred during installation.

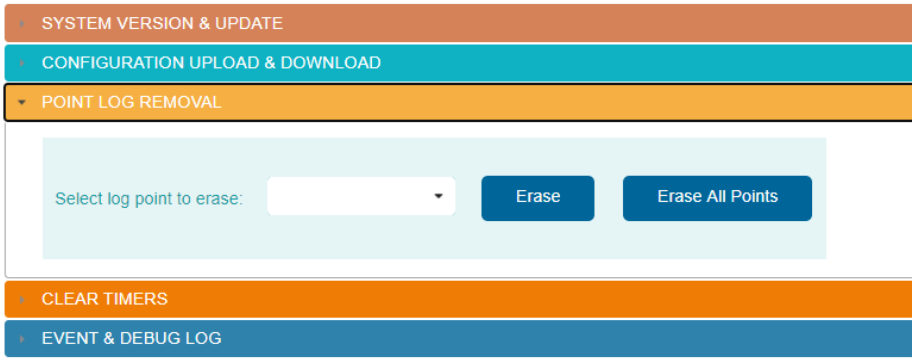


Figure 62 - Remove log data

8.4.4 Clear Input Timers

This option will clear all digital input timers or digital output timers at one time. Select which timers to clear from the drop down and then click the Clear button.

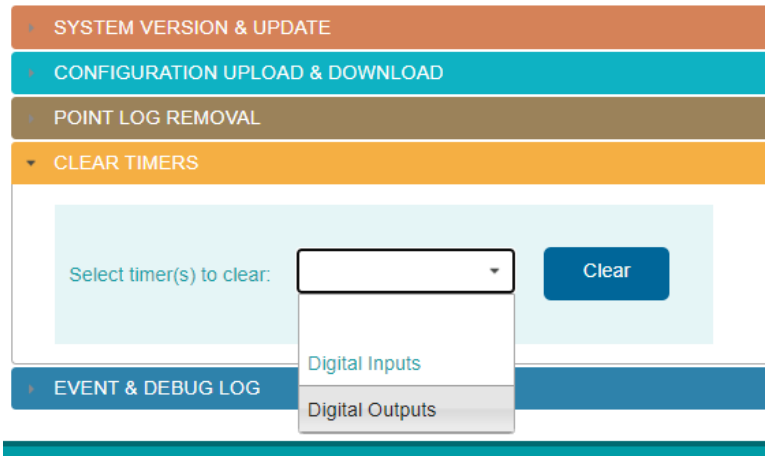


Figure 63 - Clear input timers

8.4.5 Event & Debug Log

The Event and Debug log will provide diagnostic information regarding the operation of the ESB2. Typical use of this log would be to resolve issues with sending traps or email alarms. Verifying that equations are running is another key use of the Event & Debug log. After loading equations (see 12.2) go to the Event & Debug log and verify that Equations are selected alarm with Debug from the drop down and select regenerate system logs. Correct any errors presented in the log.

Contact Quest for assistance if you are experiencing issues with your ESB2 system. The filter options allow you to select the category and then the detail level of events to be displayed. Select the choices you want and use the optional date range and press the Regenerate Debug Log button. The information will be displayed in a window with controls for forward, back, beginning and end of log navigation. A download button will also be displayed which when pressed will download a text file of the log to your browser.

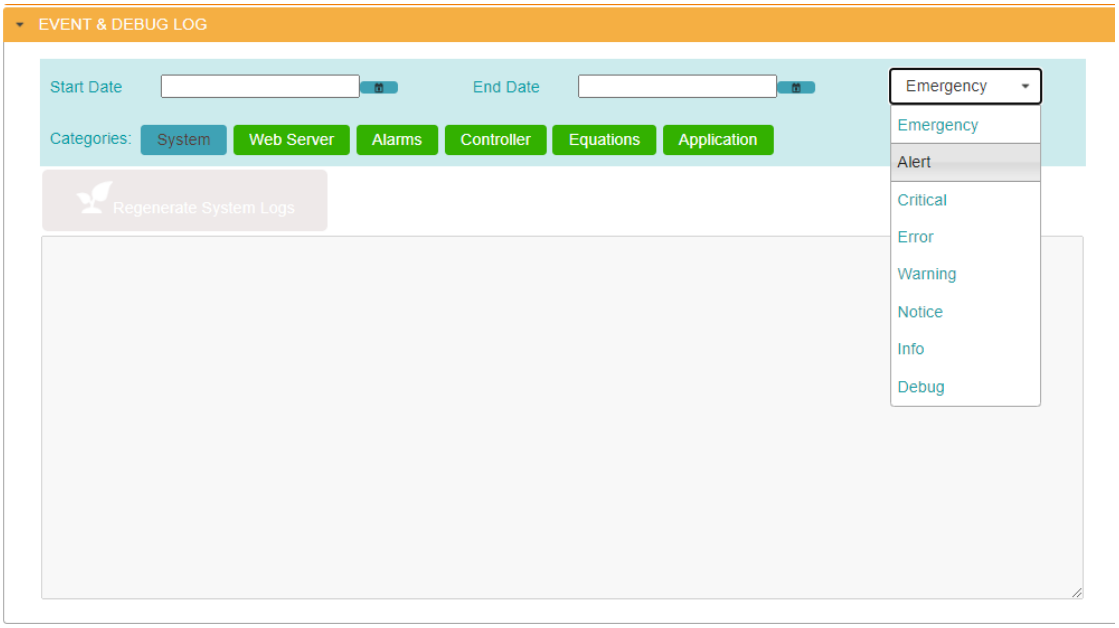


Figure 64 - Event and debug log

9 Programming the System

Hovering on the Program menu will display the available program options. For all program pages the system must save changes/additions to flash and do a reboot. When entering data, you will notice that after pressing the Done button there will be a 60 second count down timer in the header. You will have 60 seconds to enter the next point information, which will reset the timer to 60 seconds again. You can continue to add additional points in this manner or if you are completed, you can click on the Save and Reboot button to save immediately instead of waiting for the time out. Note for the Modbus programming option to appear in the menu, you must first enable Modbus on at least one of the serial ports. See section 8.1.6

9.1 Input Programming

The program input page will show all inputs that have been defined with their current settings. The point names are hyperlinks to bring up a window for changing the setting. The same window will appear when selecting the "Add new point" button. Use the forward, back, beginning and end icons to navigate through the pages of inputs. The Jump field can be used to enter a specific input number and jump directly to the page where that input is contained.

PROGRAM UNIVERSAL INPUTS

Point #	Name	Input Type	Offset	Custom Low	Custom High	Log Type	Log Interval (min)
1	Tower - Light Status	DRYNO	0	0	0	COS	0
2	Tower - Photocell Failure	DRYNC	0	0	0	COS	0
3	Tower - Marker Light Fail Alarm	DRYNO	0	0	0	COS	0
4	Tower - Night Beacon Light Fail	DRYNO	0	0	0	COS	0
5	Tower - Day Beacon Light Failure	DRYNO	0	0	0	COS	0
6	Light Level Sensor	Custom	0	0	300	Time	15
8	Zone Temp 1	TEMPF	0	0	0	Time	5
9	Fire Panel Alarm	DRYNO	0	0	0	COS	0

« ‹ Jump › »

Add new point

Save Settings

Figure 65 - Defining Universal Inputs

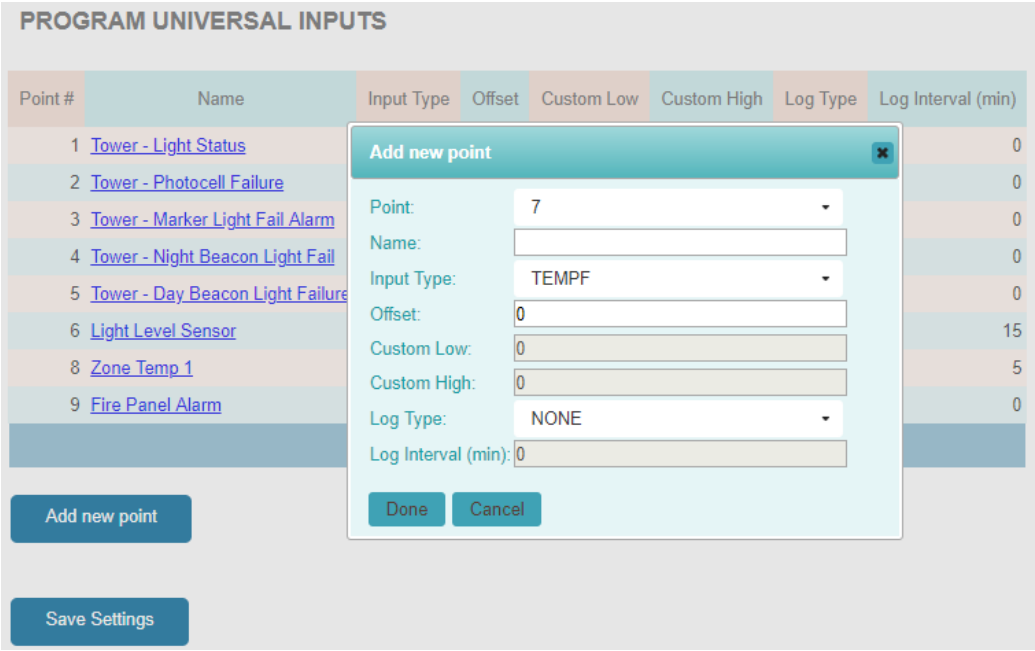


Figure 66 - Add/change input settings

9.1.1 Point

This is the physical location of the input on the ESB2 main board or expansion boards (if defined). Use the dropdown to select the desired input.

9.1.2 Point Name

The point name will allow up to thirty-two characters for a unique name to describe the point being monitored. The name is case sensitive and will be displayed exactly as you typed it.

9.1.3 Sensor Type

The sensor type dropdown will list the available choices for the input. They are as follows:

DRYNO	A dry contact closure that is normally open. This input will show an OFF value when open and an ON value when closed. *The input is wired between IN# and Ground
DRYNC	A dry contact closure that is normally closed. This input will show an ON value when open and an OFF value when closed. *The input is wired between IN# and Ground
WETNO	A wet contact closure that is normally open. A wet contact is an input that provides its own voltage. A value greater than 2.8 VDC is determined to be in the ON state and a voltage less than 0.8VDC is deemed to be in the OFF state. *The input is wired between IN# and 24V terminal or an external voltage. The maximum voltage allowed on the input is 65VDC.
WETNC	A wet contact closure that is normally closed. A wet contact is an input that provides its own voltage. A value greater than 2.8 VDC is determined to be in the OFF state and a voltage less than 0.8VDC is deemed to be in the ON state. *The input is wired between IN# and 24V terminal or an external voltage. The maximum voltage allowed on the input is 65VDC.
TEMPF	The ESB2 supports 10k Type III thermistors. Choosing TEMPF will display readings in Fahrenheit.
TEMPC	The ESB2 supports 10k Type III thermistors. Choosing TEMPC will display readings in Celsius.
CUSTOM	The custom scale allows you to enter a scale using the Custom Low and Custom High fields. Any sensor that provides a signal between 0-5 VDC or 0-20 mA can be used. Note the scale is a linear interpolation between the low and high values. Quest has a spreadsheet file to assist in determining scale factors. Please contact your representative to obtain a copy of the file.

9.1.4 Offset

The offset field is used for any analog sensor to correct the reading of the input. The ESB2 will read the input and add the offset value automatically so all alarms, logging and display screens show the corrected value.

9.1.5 Custom Low

This field is used when selecting the CUSTOM input type and will be the value displayed when the input is reading the minimum value or 0 VDC in. The ESB2 will use this value along with the Custom High value to create a scale and display the sensor in the correct engineering units.

9.1.6 Custom High

Used in conjunction with the Custom Low field to create a scale for the sensor attached to the input. The High value is the reading the sensor would have if the input is at maximum input value or 6 VDC. Quest has created a special Excel spreadsheet for calculating the minimum and maximum values based on your sensor output. Contact Quest to obtain a copy of this file for your use.

9.1.7 Log Type

Choose between TIME for a timed interval, COS for change of state or NONE to disable the logging for the input. The TIME option is best for analog sensors to log at a user specified interval. Using COS for digital inputs will optimize your logging to only put entries in the historical log when the point changes its ON/OFF state. If you select COS for an analog input, the system will log whenever the input value changes. Needless to say, this will fill up your log rather quickly if you decide to use COS on an analog point.

9.1.8 Log Interval

This is the logging interval in minutes when you select a log type of TIME. The field will allow values between 1 and 120 minutes for the interval.

9.2 Output Programming

The program outputs page will show the current settings for defined outputs. The output name is a hyperlink that when clicked on will bring up the change settings window for that output or you can click on the Add new point button.

PROGRAM OUTPUTS

Point #	Name	Default State	Follows	Acknowledge Input
1	HVAC1_Fan	DE-ENERGIZED	Equations	unused
2	HVAC1_Y1	DE-ENERGIZED	Equations	unused
3	HVAC1_Cmpr	DE-ENERGIZED	Equations	unused
4	HVAC1_Heat	DE-ENERGIZED	Equations	unused
5	HVAC2_Fan	DE-ENERGIZED	Equations	unused
6	HVAC2_Y1	DE-ENERGIZED	Equations	unused
7	HVAC2_Cmpr	DE-ENERGIZED	Equations	unused
8	HVAC2_Heat	DE-ENERGIZED	Equations	unused

Add new point

Save Settings

Figure 67 - Output settings page

9.2.1 Output Name

Enter up to thirty-two alphanumeric characters to create a unique name for the output. The name is case sensitive and will be displayed exactly as you typed it.

9.2.2 Default State

The default state is the condition of the relay when it is not being commanded to the active state. This option allows for the relay coil to be either de-energized or energized as the default state of the relay. Select whether the output should be De-energized when not activated (normal relay logic) or Energized when not in the activated state (reverse relay logic).

9.2.3 Output Follows Option

Use the dropdown to select what function you want the relay to follow. This dropdown will show the options of Equations meaning it can be actuated through equation logic, Alarm severity options or defined time schedules. Time schedules must be defined first for them to appear in the dropdown. By following an alarm severity, the output will be in the activated (ON) condition when there is an active alarm with the chosen severity. The output will stay activated until there are no longer any active alarms with the chosen severity.

The program output window has the option to select a digital input as an acknowledge function. If the output is activated due to an Active Alarm, then the output will de-activate when the acknowledge input is in the ON condition. This is useful when using a local enunciator panel that has a momentary input to acknowledge the alarm condition and turn off the output. For example, you may program an output to control a warning light when there is a critical alarm active and program another relay controlling the horn. Pressing the acknowledge button could silence the horn until the next alarm occurs. See Figure 68 below.

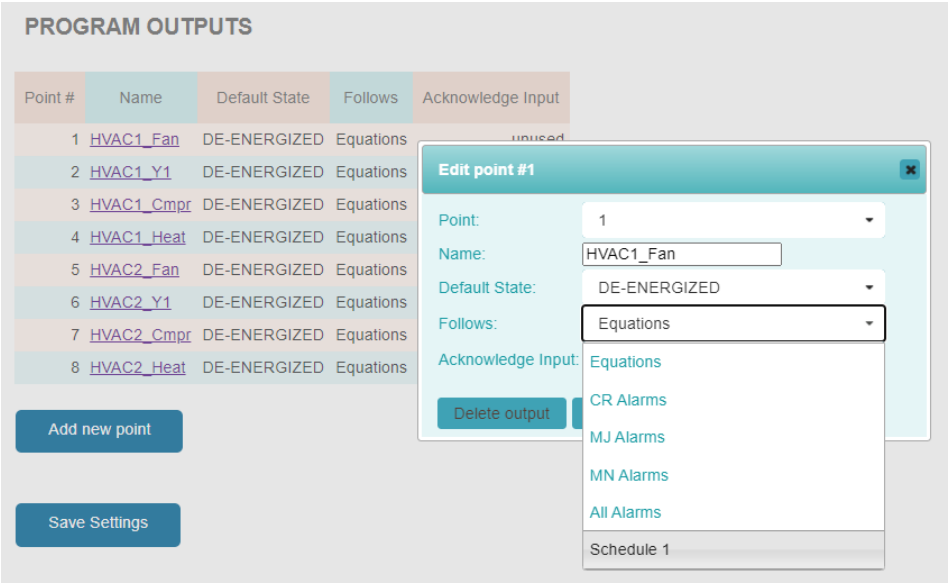


Figure 68 - Program output window

Press the Done button to go back to the program outputs page and then select another output to program or press the Save Settings button.

9.3 Analog Output Programming

The program analog outputs page will show the current settings for defined analog outputs. The output name is a hyperlink that when clicked on will bring up the change settings window for that output or you can click on the Add new point button.

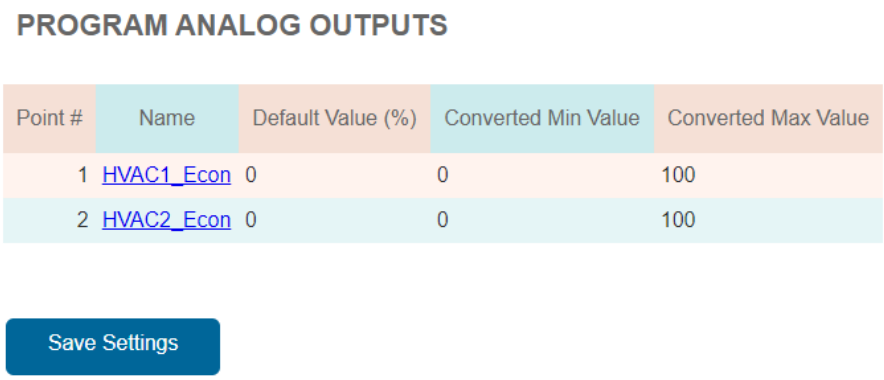


Figure 69 - Output settings page

9.3.1 Output Name

Enter up to thirty-two alphanumeric characters to create a unique name for the output. The name is case sensitive and will be displayed exactly as you typed it.

9.3.2 Default Value

The default value is the percentage output the point should be at before an equation changes the output to another value. Enter a value between 0-100 to set the value.

9.3.3 Converted Values

The ESB2 will display a value on the status page that is based on the percentage output. For example, if you are controlling a variable speed motor, the ESBS can display the RPMs based upon the percentage output that the AOP is set to. Enter the value to be displayed when the AOP is at the minimum (0%) position in the Converted Min Value field and enter the value to be displayed when the AOP is at its maximum output (100%) in the Converted Max Value field.

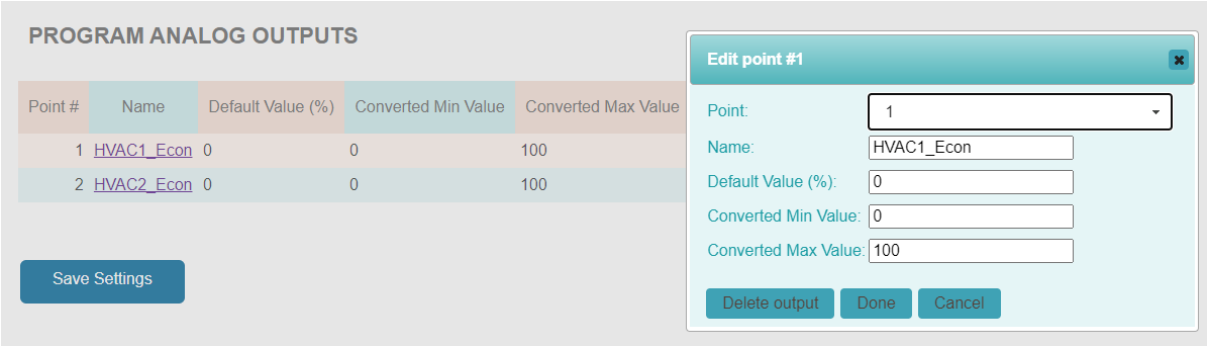


Figure 70 - Program output window

Press the Done button to go back to the program outputs page and then select another output to program or press the Save Settings button.

9.4 Modbus Programming

The Modus programming option will be present if either serial port 1 or 2 is defined for Modbus. Both ports can be configured simultaneously to provide a total of 1024 Modbus points (2 ports x 512 registers each). The ESB2 acts as a Polling Agent to request Modbus RTU data over the RS485 two wire serial ports.

9.4.1 Device Configuration

Select Modbus port 1 or port 2 from the program menu to see the Modbus program page. The first section of the page is for defining the name and the address of the device on the network. The name can be up to thirty-two-character alphanumeric characters in upper or lower case. The next section(s) of the page will have a header for each defined device with the name of the device. Clicking on the expand icon for the device will show all defined registers for the device along with the option to add points.

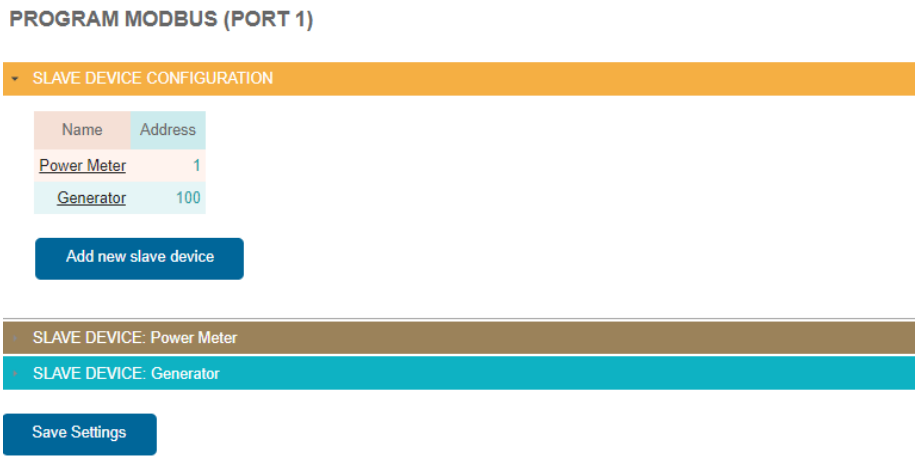


Figure 71 - Modus program page

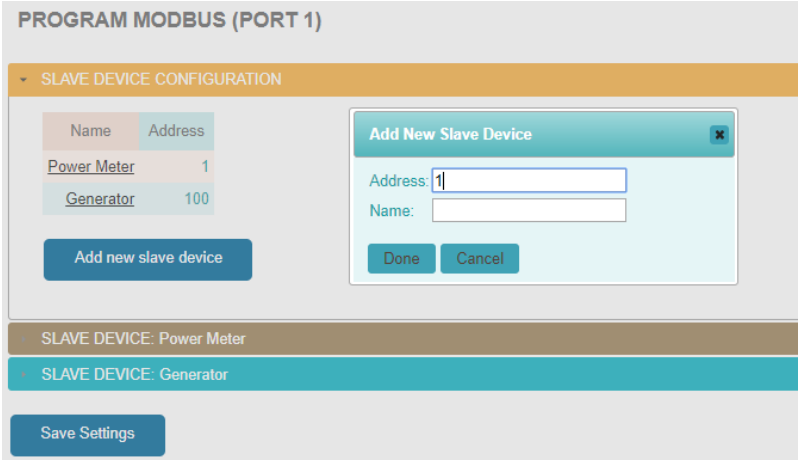


Figure 72 - Modbus define device page

9.4.2 Modbus Registers

Click on the expand window icon for the device that you want to add registers. The window will show the defined registers with navigation buttons for forward, back, beginning and end along with a link to add new points. The point ID is a hyperlink to bring up the define window for the previously defined register. The Add new point button will bring up the same window with blank form so you can add a new point.

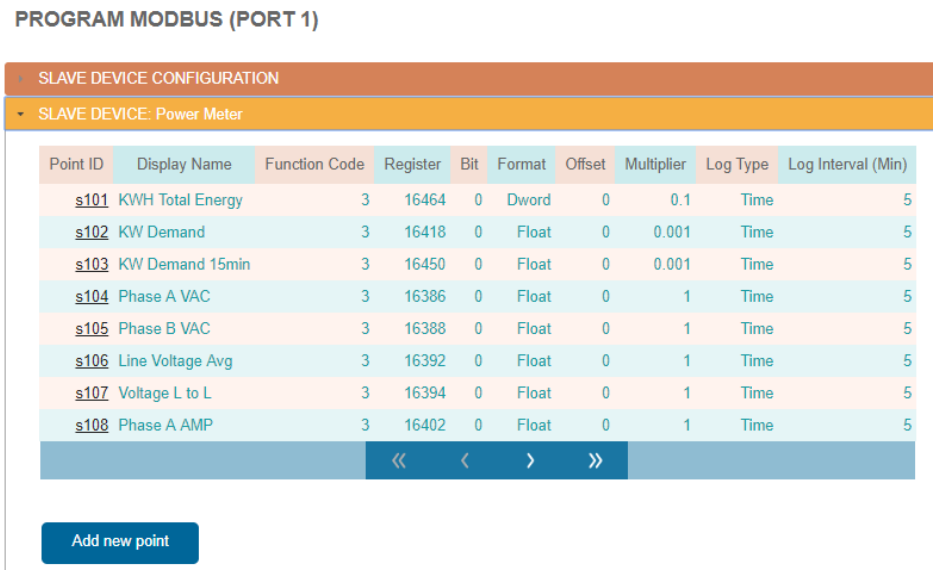


Figure 73 - Modbus registers by device

9.4.3 Defining Modbus Registers

Click on the Point ID of an existing Modbus point or select Add new point. The Modbus program point window will appear with the necessary fields for creating a point in the system.

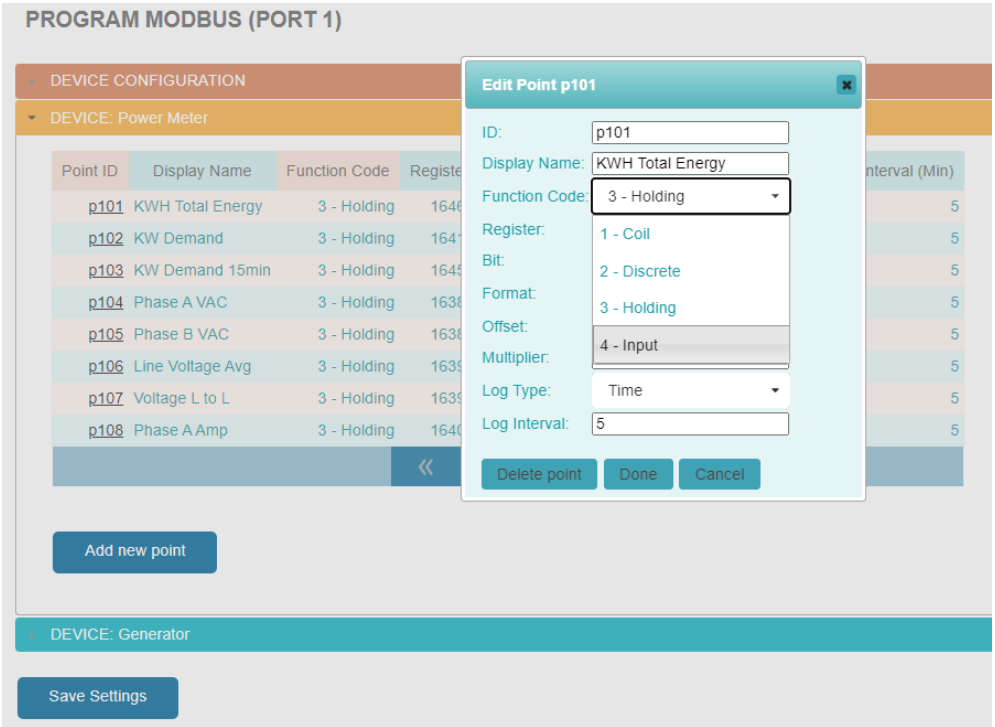


Figure 74 - Defining Modbus registers

9.4.3.1 Modbus ID

The ID field is a unique alphanumeric four-character code used to identify the point in the system. This is used for alarm programming, equations and SNMP to create a reference ID for the point. The four-character ID can be whatever the user would like to use to organize the points. For example, you may

want to have IDs per device so later you can sort data based on the ID, Such as G001 for generator point 1, etc. You may want to use the same numbering scheme when setting up multiple sites because the ID will also be the SNMP index number for doing sets and gets.

9.4.3.2 Display Name

The display name will allow up to thirty-two characters for a unique name to describe the point being monitored. The name is case sensitive and will be displayed exactly as you typed it.

9.4.3.3 Function Code

The function code option tells the ESB2 what type of Modbus register is being polled. 1 is for coil registers, 2 is for digital inputs, 3 is for holding registers and 4 is for analog input registers. Select the option for the type of register to be monitored.

9.4.3.4 Register Number

Enter the number of the register in the Register field. The ESB2 uses 0 based numbering scheme for the register numbers. If the register map uses the Modicon convention of 40000 series numbers for holding registers, then subtract 40001 from the number and use that for the register number. For coils subtract 1, Digital input subtract 10001 and for analog input registers subtract 30001.

9.4.3.5 Register Bit

Some devices will use a single register with 16 bits to represent multiple pieces of information. This field will allow you to select a specific bit of the 16-bit register. When using the bit function, the value of the register will be displayed as ON for a value of 1 and OFF for a value of 0. Enter the bit number where 1 means the lowest bit and 16 means the highest bit. Some register map will show 0-15 and in that case 0 is the first bit and 15 is the 16th bit.

9.4.3.6 Register Format

Use the Format drop down to select the register type. Digital inputs and Coil registers are single Byte format. Analog input registers are always Word registers. For Holding registers, they can be Word (16 bit), Dword (32bit), Float for floating setpoint or Sword for signed word format. Sign word is a 16-bit register that can have a positive or negative value.

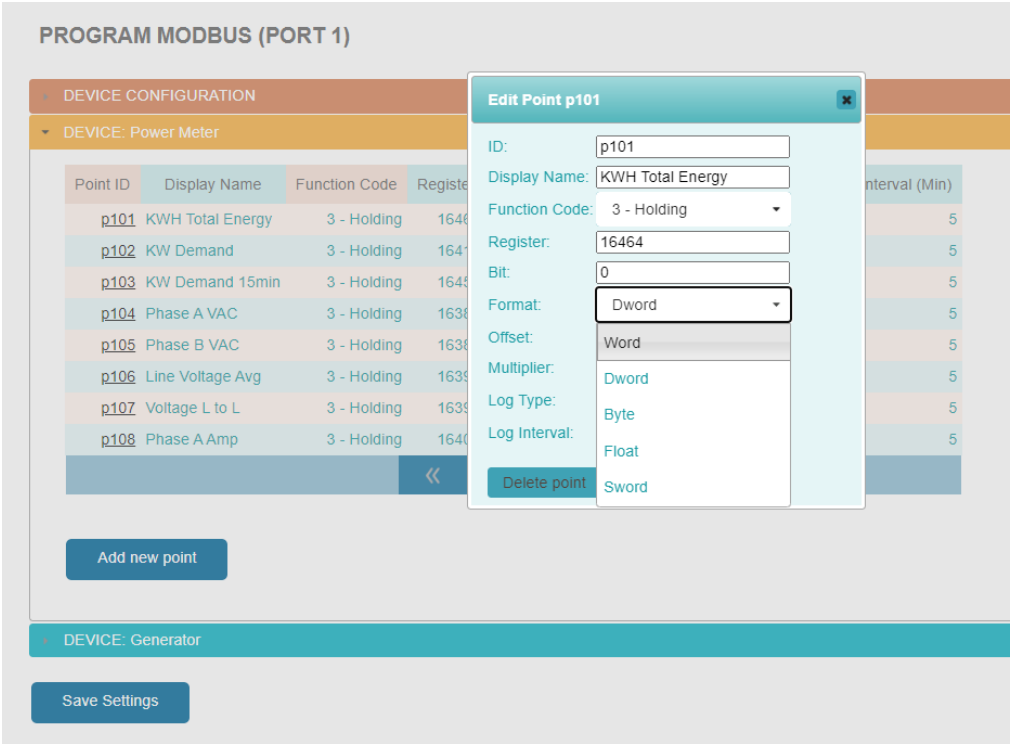


Figure 75 - Modbus Register Format

9.4.3.7 Register Offset and Multiplier

The offset field is used to correct or modify the value being returned by the Modbus device. The offset is applied after any multiplier being used for the register. Enter a positive or negative value to be applied to the reading. Some registers return an integer value that could be in different engineering units than what is desired to be displayed in the ESB2. The multiplier can be used to change the displayed value into the proper units. For example, to convert a register reading Celsius into Fahrenheit, the multiplier would be 1.8 and the offset would be 32. Or if a register is reading in hundreds and you want to change that to ones based you could have a multiplier of 0.01.

9.4.3.8 Log Type and Interval

Select the log type of NONE for no logging, COS for Change of Stage or Time for interval logging based on the number of minutes entered in the interval field. COS logging works best for registers that return a binary state such as 0 and 1 or on and off. Interval logging is best used for analog sensors to trend their value of a specific time interval. The maximum time interval is 1440 minutes (24hrs).

9.5 QMbus Modules

Adding additional input and output modules to the system is accomplished through this page. The page will list all defined Quest expansion modules with a button to add additional modules. Defined modules will show the address of the module in decimal and hex format. Clicking on this hyperlink will allow you to remove the module from the system. Clicking on the Add new module button will bring up a dialog with a dropdown to select which module address you would like to add to the system.

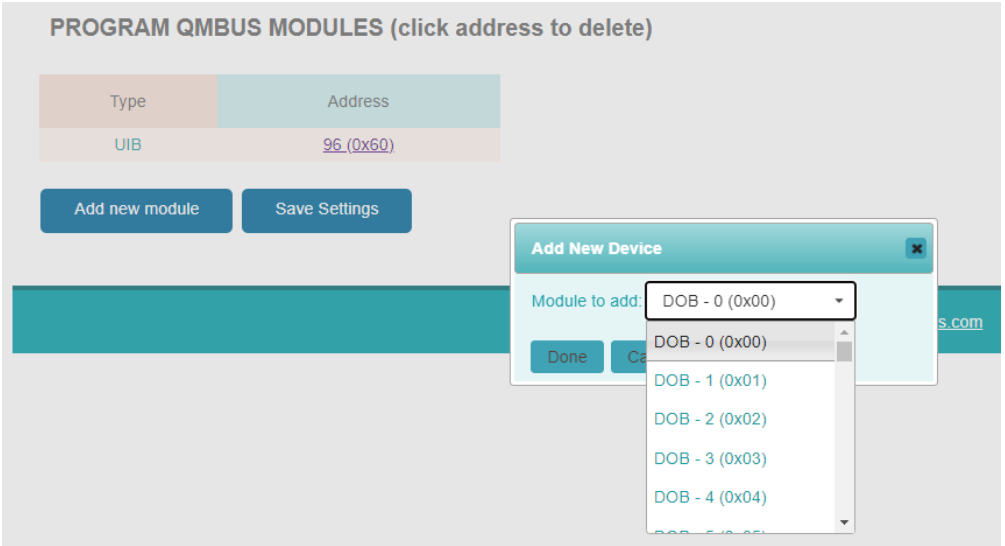


Figure 76 - Define Bus Modules

Table 4 - Expansion Modules and Point Numbers

Board	Bus Address	Hex #
DOB 1 (RLY 17-24)	0	0x00
DOB 2 (RLY 25-32)	1	0x01
DOB 3 (RLY 33-40)	2	0x02
DOB 4 (RLY 41-48)	3	0x03
DOB 5 (RLY 49 – 56)	4	0x04
DOB 6 (RLY 57 – 64)	5	0x05
DOB 7 (RLY 65 – 72)	6	0x06
DOB 8 (RLY 73 – 80)	7	0x07
DOB 9 (RLY 81 – 88)	8	0x08
DOB 10 (RLY 89 – 96)	9	0x09
DOB 11 (RLY 97 – 104)	10	0x0A
DOB 12 (RLY 105 – 112)	11	0x0B
DOB 13 (RLY 113 – 120)	12	0x0C
DOB 14 (RLY 121 – 128)	13	0x0D
DOB 15 (RLY 129 – 136)	14	0x0E
DOB 16 (RLY 137 – 144)	15	0x0F
UIB 1 (UIN 17-32)	96	0x60
UIB 2 (UIN 33-48)	97	0x61
UIB 3 (UIN 49-64)	98	0x62

Board	Bus Address	Hex #
UIB 4 (UIN 65-80)	99	0x63
UIB 5 (UIN 81-96)	100	0x64
UIB 6 (UIN 97-112)	101	0x65
UIB 7 (UIN 113-128)	102	0x66
UIB 8 (UIN 129-144)	103	0x67
UIB 9 (UIN 145-160)	104	0x68
UIB 10 (UIN 161-176)	105	0x69
UIB 11 (UIN 177-192)	106	0x6A
UIB 12 (UIN 193-208)	107	0x6B
UIB 13 (UIN 209-224)	108	0x6C
UIB 14 (UIN 225-240)	109	0x6D
UIB 15 (UIN 241-256)	110	0x6E
UIB 16 (UIN 257-272)	111	0x6F
AOB 1 (AOP 3-6)	112	0x70
AOB 2 (AOP 7-10)	113	0x71
AOB 3 (AOP 11-14)	114	0x72
AOB 4 (AOP 15-18)	115	0x73
AOB 5 (AOP 19-22)	116	0x74
AOB 6 (AOP 21-26)	117	0x75
AOB 7 (AOP 27-30)	118	0x76
AOB 8 (AOP 31-34)	119	0x77
AOB 9 (AOP 35-38)	120	0x78
AOB 10 (AOP 39-42)	121	0x79
AOB 11 (AOP 43-46)	122	0x7A
AOB 12 (AOP 47-50)	123	0x7B
AOB 13 (AOP 51-54)	124	0x7C
AOB 14 (AOP 55-58)	125	0x7D
AOB 15 (AOP 59-62)	126	0x7E
AOB 16 (AOP 61-66)	127	0x7F

9.6 Alarm Programming

Select Alarms from the Program menu to see the alarm program screen. This screen will display all programmed alarms in a table with the ability to navigate the table using the next, back, beginning and end buttons. In addition, the page has filter options to display all alarm points or just those associated with the direct inputs Modbus devices and Control points (EQU, VAR, SPT). The filter section will also allow you to search based on the alarm name or the name of the point used in the alarm. For example, to see all Tower alarms, you can type Tower in the name field and press enter. Now only alarms with the word Tower in the name will be displayed.

The alarm names are hyperlinks that when selected will bring up the program alarm window. The same window is used for new alarms when selecting the Add new alarm button.

Figure 77 - Program alarm page

9.6.1 Defining Alarm Strategies

Selecting an existing alarm name or clicking on the Add new alarm button will bring up the dialog for modifying or adding a new alarm.

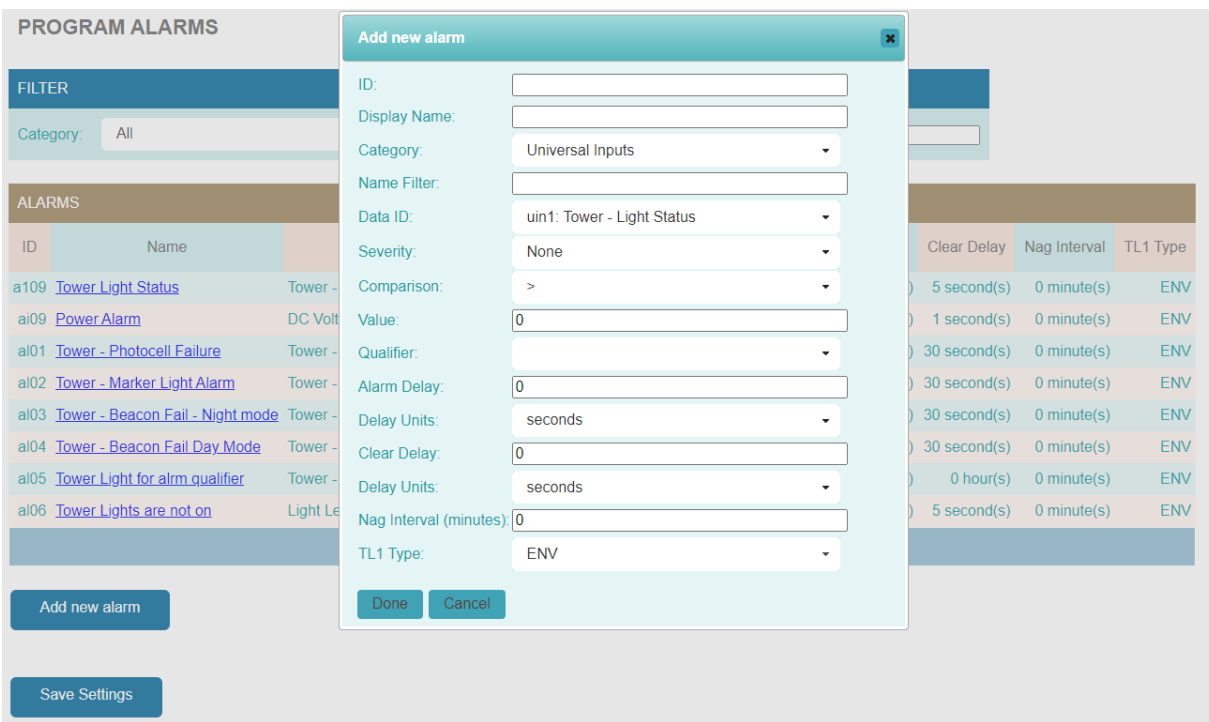


Figure 78 - Defining an alarm strategy

9.6.1.1 Alarm ID

The ID field is a unique alphanumeric four-character code used to identify the point in the system. This is used for alarm programming, and SNMP to create a reference ID for the point. The four-character ID can be whatever the user would like to use to organize the points but must start with an alpha character. For example, you may want to have IDs per alarm type so later you can sort data based on the ID, Such as GEN1 for generator point 1 alarm, etc. You may want to use the same numbering scheme when setting up multiple sites because the ID will also be the SNMP index number for doing sets and gets.

9.6.1.2 Display Name

The display name will allow a thirty-two-character unique name to describe the point being monitored. The name is case sensitive and will be displayed exactly as you typed it.

9.6.1.3 Category and Data ID

The category field is used in conjunction with the Data ID dropdown to filter the display list. This way you can limit the size of the dropdown list for specific points. The category list choices are ALL for every defined point in the system, universal inputs for a list of the defined inputs, Modbus/Device name for a list of defined Modbus points sorted by device and defined control points (EQU, VAR & SPT). Select the desired category and then select the Device ID to be used for the alarm strategy.

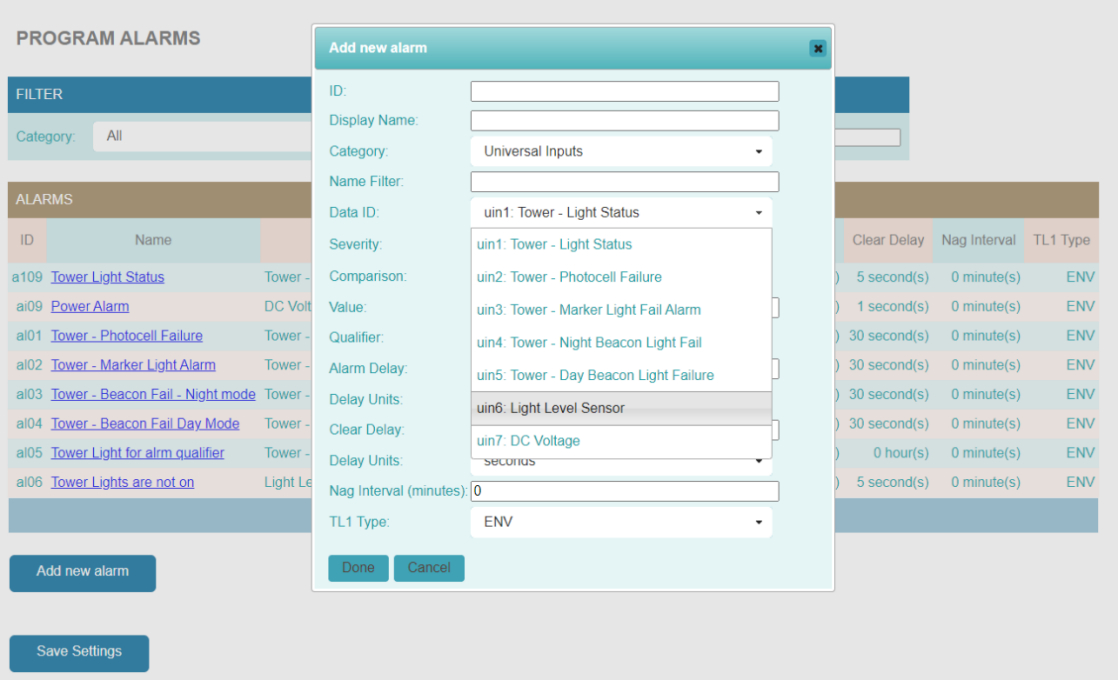


Figure 79 - Selecting a point to alarm

9.6.1.4 Alarm Severity

Select the desired alarm severity from the Severity dropdown. A severity of NONE will disable the alarm until the severity is changed to another level. Notice is a special case severity level where the alarm when active and cleared will appear in the alarm historical log, but alarms with Notice level are not sent as traps or emails.

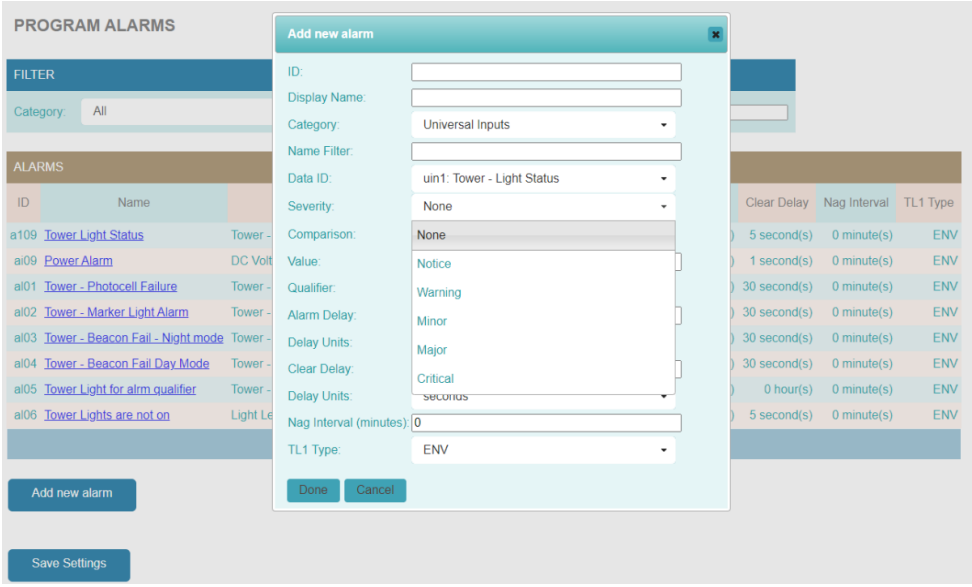


Figure 80 - Selecting the alarm severity

9.6.1.5 Comparison operator and value

The Comparison field is used to select the operator for creating the alarm strategy. The alarms are looking for "true" statements and will alarm when the condition described is in the true state. The value field is used with the operator to determine if the point being monitored is in the alarm condition. Digital inputs are compared to 0 for OFF and 1 for ON. Thus, to create an alarm when the digital input has an ON status, you would compare the input > (greater than) 0 or you could compare the input = (equal to) 1, or input ≠ (not equal to) 0. Analog sensors such as temperature and humidity, use the greater than (>) value or the less than (<) value for alarm condition determination. An example would be if the zone temperature sensor is greater than 80 degrees. Alarm values are entered as integer numbers.

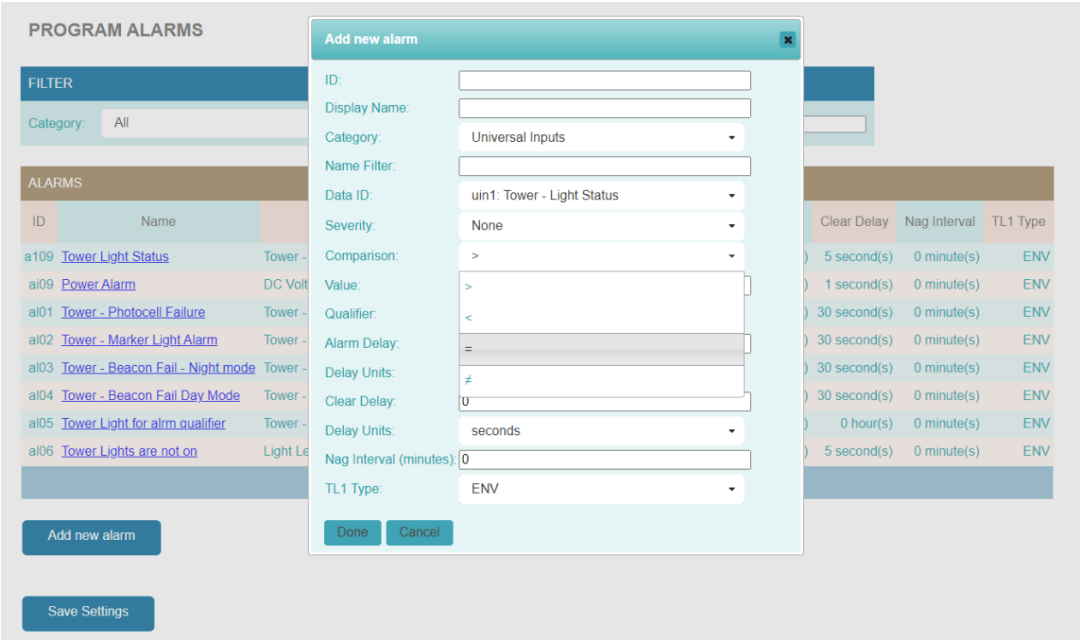


Figure 81 - Select comparison operator

9.6.1.6 Additional Qualifier

Alarm strategies allow for a qualifier to create additional conditions that must be true (on) before this alarm statement is true or active. The qualifier field is optional and will contain a list of previously defined alarm strategies and time schedules. Time schedules can be used to only create the alarm during certain times of the day or days of the week. By using other alarm strategies as qualifiers, you can create combinatorial logic to create and alarm based on multiple conditions. The ESB2 will allow you to string together multiple alarm strategies in this way to then generate the alarm condition. Keep in mind that ALL conditions must be true or active for the alarm strategy to be in alarm (active/on). If any conditions become false (off) then the alarm strategy is no longer active and has a status of off, which generates a clear alarm in the alarm log.

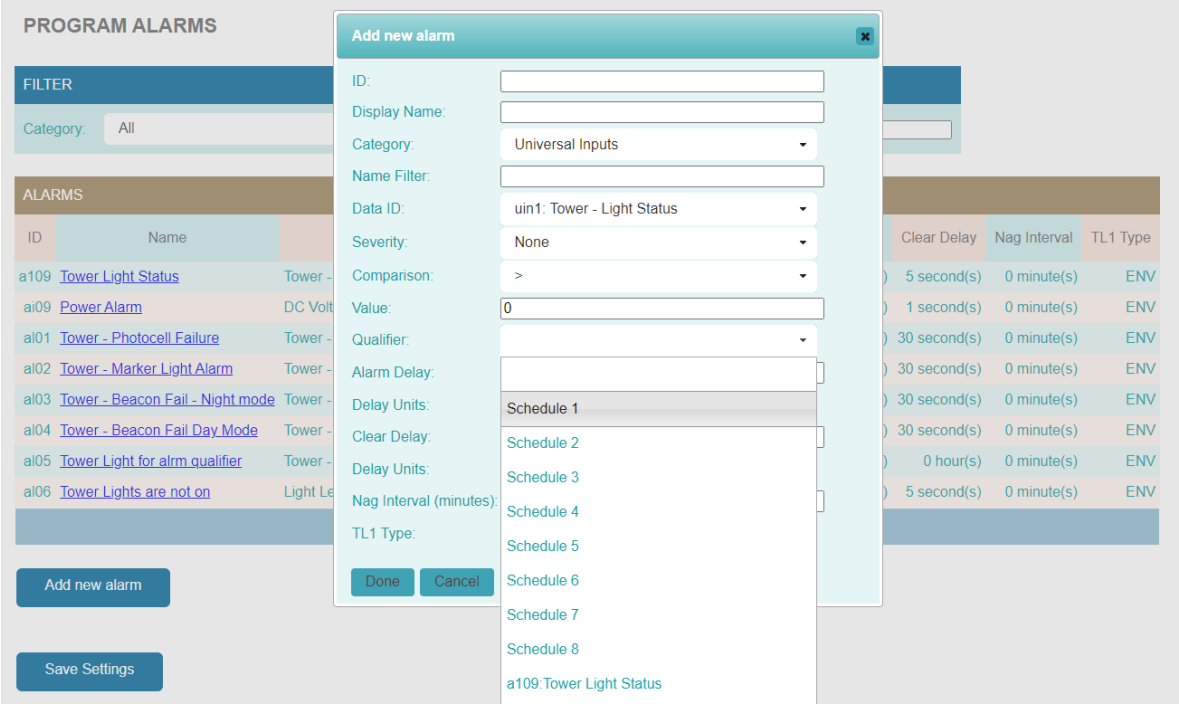


Figure 82 - Using additional qualifiers

9.6.1.7 Alarm Delay and Delay Units

Use the Alarm Delay field and the Delay Units drop down to create a delay time for the alarm condition. The alarm condition must be true for this time period for the alarm to be generated. An alarm will not be generated if the conditions become false before the delay time elapses. The maximum delay time is 25 hours.

9.6.1.8 Clear Delay and Delay Units

Use the Clear Delay field and the Delay Units drop down to create a delay time for the clear condition. A clear alarm will be generated once the alarm condition is no longer true for the defined delay time. The maximum delay time is 24 hours.

9.6.1.9 Nag Interval

The nag interval is used to resend the alarm if the alarm is still active. A value of 0 means the alarm will be sent once per occurrence. A value in minutes greater than 0 will create an interval timer where the alarm will be resent every time the time has been elapsed. The maximum nag interval is 1440 minutes.

9.6.1.10 TL1 Type

This field allows the user to select the TL1 Alarm type. The default is ENV for environmental alarm but can be changed to Equipment Service Affecting (EQPTSA) or Equipment Non-service affecting. This function is used with the supported TL1 alarm protocol.

9.7 Date Schedules

The ESB2 supports date schedules that can be a single date, two dates or a range between the two defined dates. Dates schedules can be used in Time Schedules to do different actions during the specified date range. Date schedules can also be referenced directly in equations. Examples of using date schedules would be to change time schedule operation on specific dates such as holidays or to define winter/summer modes for central cooling & heating plant operations.

Select "Dates" from the Program menu to see a list of defined dates. Each date will have a unique thirty-two-character name along with the dates the schedule is active. Click on the name of the schedule or click on the Add new date button to bring up the define dialog. Enter the unique name for the schedule and use the calendar tool to select the start date and ending dates. If there is only one date, then enter it in the start date field and select NONE for the date range. The other options for the date range it TO and. TO will make the schedule active (on) from the start date through the end date. AND will make the schedule active (on) only on the start and end dates and not the dates in between.

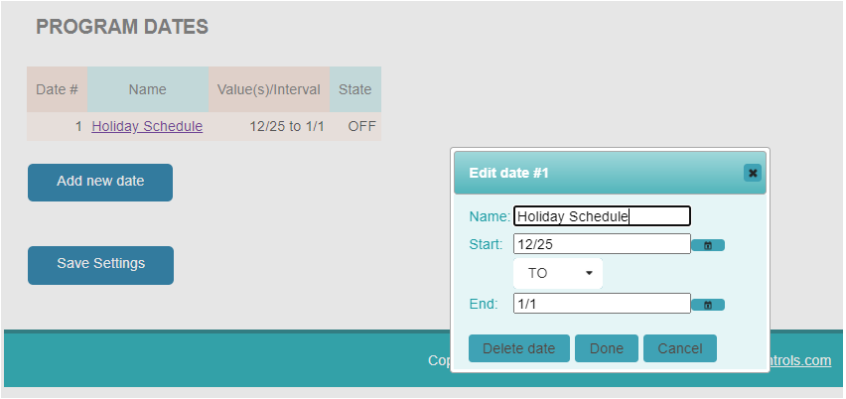


Figure 83 - Define Date Schedule

9.8 Time Schedules

Select Schedule from the Program menu to see the Program Schedule page. This page will show the defined schedules (up to 8 schedules) and their current status or state. The name of the schedule is a hyperlink that when selected will bring up a window showing the current settings and allow the user to modify the schedule. The "Add new schedule" button will bring up the same change schedule dialogue window. Press the "Save Settings" button once you have made changes to an existing schedule or created a new schedule.

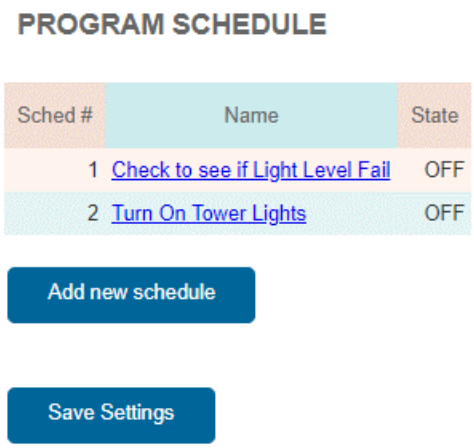


Figure 84 - Change/view time schedules

9.8.1 Change Time Schedule

The change schedule window has a place for the user to enter a unique thirty-two-character name to describe the schedule and up to sixteen priorities for turning the schedule on and off. Select the action for a priority line and then select the hours, minutes and AM/PM setting. Finally select which days of the week the priority line will be active or change the Day Type radio button to allow a Date Schedule and then select the desired schedule. Press the "Done" button once you are completed with filling out the priorities needed. Schedules typically have at least one ON and OFF priority to allow the schedule to change status based on time of day and the day of the week (or date schedule). Once the schedule is created, it will be available for use in programming outputs, equations and alarm strategies.

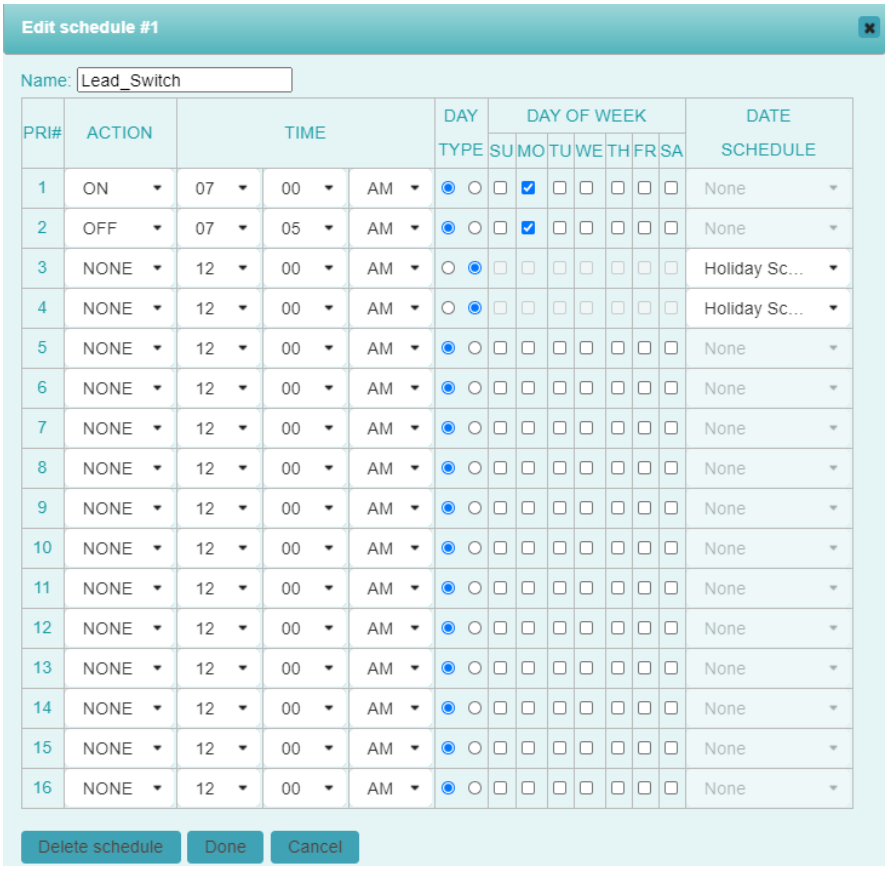


Figure 85 - Define schedule priorities

9.8.2 One-time Equations

The web server provides a page where the user can enter equations and see the results. This feature is ideal if you want to try the syntax of an equation to see how it will operate, make a quick change to a memory variable or to understand how the equation processor is interpreting data. Refer to section 11 for information regarding equations and program structure. When testing equations, omit the equation number and the back ticks for the beginning and ending of equations. They are not used with the one time run function.

The print command is a primary function in the One-time equation processor to display the status of the point(s) you desire. The print command will show the status of string, integer and floating-point types. Boolean variables (TOD, DAT, ALM) require additional commands to print the value correctly.

Format for integer and floating point values: print (<variable>) where <variable> is the status information you are looking for.

Format for string values: print (“<value>”)

Format for Boolean values: print (tostring(variable))

Result:

```
1: -155.09
1: This is a string value
1: false

[Runtime: 1 iteration(s)]
```

```
print(INP.uin1)
print("This is a string value")
print(tostring(TOD.tod1))
```

Run Equation

Figure 86 - One-time Equation Window

Multiple commands can be entered at the same time and the ESB2 will process each command in order and provide the results if the print command is used. Syntax error checking is employed on each command and the ESB2 will display an error message if there is an issue.

Result:

```
1: 2
1: 1

[Runtime: 1 iteration(s)]
```

```
print(VAR.varleadhvac)
VAR.varleadhvac = 1
print(VAR.varleadhvac)
```

Run Equation

Figure 87 - One-time Equations with Multiple Commands

10 Shell Commands

The ESB2 supports a management shell that is accessible through SSH or Telnet. From this shell, the user can issue various commands to administer the communications of the system as well as review status of various points. Use your favorite SSH or Telnet client to connect to the ESB2. The default username/password is Shelllevel/shelllevel. The username/password can be changed or additional username/password combinations can be created through the Telnet/SSH User Configuration on the web server. See section 8.1.11 of this manual. A command prompt will be shown after a successful login. Type the word "help" to see the available commands and format:

Available commands:
help
show alarms states
show alarms history
show inputs points history
show inputs points
show outputs digitals points history
show outputs analogs points
show outputs digitals points
show equations setpoints
show modbus points history
show modbus points
show modbus ports addrs points
show modbus ports addrs status
show modbus ports points
show modbus ports status
show system info
set ip
set mtu
set adminuser
set httppswd
set snmpmgr
set upgradeurl
set upgradecheck
set warmstart on
ping
exit
logout

All commands when entered will execute without further confirmation to the user. When using the "set" commands the system will output a message of "parameters set...unit will restart after you exit". You can do multiple "set" commands and then type "exit" or "logout" to exit and reboot the system. The system will then restart and come up with the new settings. The system will reboot in 5 minutes after the last set command if you don't issue an exit or logout command.

10.1 Help Command

Typing “help” will provide a list of all commands showing the format required to execute the command. Text contained between the less than and greater than <> symbols indicate required information for the command to process. Text between brackets [] are optional and are not required for the command to process.

10.2 show alarms states

This command will show the current state of all alarm points.

Format:
show alarms states [alarmId] [--json]

Where:
[alarmId] = optional for a specific alarm ID you want to review
[--json] = output the display in JSON format

10.3 show alarms history

This command will show the alarm history log.

Format:
show alarms history [alarmId] [--startts <MM/DD/YY_HH:MM:SS | epochTime> --endts <MM/DD/YY_HH:MM:SS | epochTime>] [--json] (Last 24 hours if startts/endts omitted)

Where:
[alarmId] = optional for a specific alarm ID you want to review
[--startts <MM/DD/YY_HH:MM:SS | epochTime> --endts <MM/DD/YY_HH:MM:SS | epochTime>] = optional start and ending time to review the log. Note the last 24hrs will be shown if the start/end times are omitted.
[--json] = output the display in JSON format

10.4 show inputs points history

This command will show an inputs history log.

Format:
show inputs points history <pointNumber> [--startts <MM/DD/YY_HH:MM:SS | epochTime> --endts <MM/DD/YY_HH:MM:SS | epochTime>] [--json] (Last 24 hours if startts/endts omitted)

Where:
<pointNumber> = the input number of the point log retrieval.
[--startts <MM/DD/YY_HH:MM:SS | epochTime> --endts <MM/DD/YY_HH:MM:SS | epochTime>] = optional start and ending time to review the log. Note the last 24hrs will be shown if the start/end times are omitted.
[--json] = output the display in JSON format

10.5 show inputs points

This command will show the current status of input points.

Format:

show inputs points [pointNumber] [--include-timers] [--json]

Where:

[pointNumber] = optional to select a single point vs. viewing all points
[--include-timers] = This option will include the timers for digital inputs
[--json] = output the display in JSON format

10.6 show outputs digitals points history

This command will show the historical log of a digital output.

Format:

show outputs digitals points history <pointNumber> [--startts <MM/DD/YY_HH:MM:SS | epochTime> --endts <MM/DD/YY_HH:MM:SS | epochTime>] [--json] (Last 24 hours if startts/endts omitted)

Where:

<pointNumber> = the output number of the point log retrieval.
[--startts <MM/DD/YY_HH:MM:SS | epochTime> --endts <MM/DD/YY_HH:MM:SS | epochTime>] = optional start and ending time to review the log. Note the last 24hrs will be shown if the start/end times are omitted.
[--json] = output the display in JSON format

10.7 show outputs analog points

This command will show the current status of the analog output points.

Format:

show outputs analogs points [pointNumber] [--json]

Where:

[pointNumber] = optional to select a single point vs. viewing all points
[--json] = output the display in JSON format

10.8 show outputs digital points

This command will show the current status of the digital output points.

Format:

show outputs digitals points [pointNumber] [--include-timers] [--json]

Where:

[pointNumber] = optional to select a single point vs. viewing all points

[--include-timers] = This option will include the timers for digital outputs

[--json] = output the display in JSON format

10.9 show equation setpoints

This command will show the current value of the defined setpoints.

Format:

show equations setpoints [dataId] [--json]

Where:

[dataId] = optional for a specific setpoint ID you want to review

[--json] = output the display in JSON format

10.10 show modbus points history

This command will show the history log data for a Modbus point.

Format:

show modbus points history <dataId> [--startts <MM/DD/YY_HH:MM:SS | epochTime> --endts <MM/DD/YY_HH:MM:SS | epochTime>] [--json] (Last 24 hours if startts/endts omitted)

Where:

<dataId> = the modbus four-character ID of desired point to get log history.

[--startts <MM/DD/YY_HH:MM:SS | epochTime> --endts <MM/DD/YY_HH:MM:SS | epochTime>] = optional start and ending time to review the log. Note the last 24hrs will be shown if the start/end times are omitted.

[--json] = output the display in JSON format

10.11 show modbus points

This command will show the current status of a specified modbus point.

Format:

show modbus points <dataId> [--json]

Where:

<dataId> = the modbus four-character ID of desired point to get status.

[--json] = output the display in JSON format

10.12 show modbus ports addrs points

This command will show the current status of all modbus points for a specific device address.

Format:

show modbus ports addrs points <portNumber> <address> [--json]

Where:

<portNumber> = The ESB2 serial port that the device is connected to. Value is either 1 or 2.

<address> = The Modbus address of the device.

[--json] = output the display in JSON format

10.13 show modbus ports addrs status

This command will show the current status of a specified modbus device to determine if it is communicating properly.

Format:

show modbus ports addrs status <portNumber> <address> [--json]

Where:

<portNumber> = The ESB2 serial port that the device is connected to. Value is either 1 or 2.

<address> = The Modbus address of the device.

[--json] = output the display in JSON format

10.14 show modbus ports points

This command will show the current status of all Modbus points on the specified serial port number. The output will be organized by the device address number.

Format:

show modbus ports points <portNumber> [--json]

Where:

<portNumber> = The ESB2 serial port that the device is connected to. Value is either 1 or 2.

[--json] = output the display in JSON format

10.15 show modbus ports status

This command will show the current status of all Modbus devices on the specified serial port number. Use this command to see troubleshoot communications issues with devices.

Format:
show modbus ports status <portNumber> [--json]

Where:
<portNumber> = The ESB2 serial port that the device is connected to. Value is either 1 or 2.
[--json] = output the display in JSON format

10.16 show system info

This command will show information about the system such as: Site ID, Site Information, firmware version, time zone, daylight savings change dates and the IO board firmware revision.

Format:
show system info [--json]

Where:
[--json] = output the display in JSON format

10.17 Set IP Command

Typing "set ip" only will show the current settings for the network configuration. To change the settings, use the following format:

Format: set ip <ipaddress> <subnetmask> <gateway> <DNS1 | none> <DNS2 | none> <http_port>

Where:
<ipaddress> = the IP address for the device in decimal dot format
<subnetmask> = The subnet mask entered in decimal dot format
<gateway> = the gateway IP address entered in decimal dot format.
<DNS1 | none> = DNS 1 server IP address in decimal dot format or the word none when DNS is not used
<DNS2 | none> = DNS 2 server IP address in decimal dot format or the word none when DNS is not used
<http_port> = The port number to use for HTTP access.

Example:
set ip 192.168.1.58 255.255.255.0 192.168.1.1 8.8.8.8 8.8.4.4 80

Where 192.168.1.58 is the IP address, 255.255.255.0 is the subnet, 192.168.1.1 is the gateway, 8.8.8.8 is DNS 1 server, 8.8.4.4 is DNS 2 server and the HTTP port is 80.

set ip 192.168.1.58 255.255.255.0 192.168.1.1 none none 80

10.18 Set MTU Command

Type “set mtu” to see the current settings. The set mtu command will change the size of the Maximum Transmission Unit value. The default value is 1500.

Format: set mtu <value>

Example: set mtu 1200

Sets the max mtu size to 1200.

10.19 Set httppswd Command

The set httppswd command will change the passwords for the built in web server. There are a total of six usernames (User1 – User5 & Admin) that can be changed. Note this changes the password only and does not change the access levels assigned to each username. To change the access level, go to Settings/HTTP User Administration. See section 8.3

Format: set httppswd <username><pswd> <pswd copy>

Where:

<username> = User1, User2, User3, User4, User5, Admin

<pswd> = the password value used when logging into the webserver using the username

<pswd copy> = reenter the password again for confirmation.

Example: set httppswd Admin Quest1234 Quest1234

Sets the password for username Admin (Admin level) to Quest1234

10.20 Set snmpmgr Command

This command will allow you to set the four SNMP trap receiver addresses. Typing “set snmpmger” without any parameters will show the existing settings for the four trap server addresses.

Format: set snmpmgr <#> <ipaddress|none> <v1|v2|v3> <Trap|Inform> <community string/v3 username>

Where:
<#> = The trap server number to be changed 1-4
<ipaddress|none> = IP address of the trap server or the word none to remove and existing definition
<v1|v2|v3> = the trap format to be used
<Trap|Inform> = Whether to send traps or informs to the server.
<community string/v3 username> = the SNMP community string to be used or one of the defined SNMP usernames to be used if sending v3 traps/informs (see section 8.1.8).

Example: set snmpmgr 1 192.168.1.104 v2 Inform Trap

Sets the first trap manager to IP address 192.168.1.104 and will send SNMP v2c informs with a community string of Trap.

10.21 Upgrade URL

The set upgradeurl will allow the user to change the IP address and path of where the Ethernet module can locate upgrades. Typing “set upgradeurl” and pressing enter will show the current location of where upgrades can be found when the system is told (through the web server link or the upgrade command) to check for upgrades. The format for changing the location is as follows:

Format: set upgradeurl <ipaddress/filelocation>

Example: set upgradeurl 192.168.1.105/ESB2

10.22 Checking for Upgrades

The set upgradecheck command can be used to have the system go check to see if there is an available upgrade and if so, the system will upgrade to the version stored on the upgraded server assuming the build number of the code is greater than what is currently running in the ESB2. Entering “set upgradecheck” will show the current firmware build number. Entering “set upgradecheck on” will cause the system to check for and perform the upgrade.

Example:

Command: set upgradecheck

Response: kernel version is #2 Fri Oct 4 10:28:18 PDT 2019
filesystem version is 2

Command: set upgradecheck on
Response: upgrade check turned on...unit will restart and check after you exit

10.23 Resting the System

The set warmstart on command will reset the ESB2 without losing any program data that has been saved to flash.

Example:

Command: set warmstart on
Response: unit will warmstart after you exit

10.24 Ping Command

The ping command will allow you to verify communications on the network. This command can be helpful when reusing an IP address for the ESB2. By doing the ping, the Ethernet switch the ESB2 is connected to will update its ARP table for the new MAC address of the unit.

Format: ping <ipaddress>

Example:

Command: ping 192.168.1.1

Response: PING 192.168.1.1 (192.168.1.1): 56 data bytes

```
64 bytes from 192.168.1.1: seq=0 ttl=64 time=10.726 ms
64 bytes from 192.168.1.1: seq=1 ttl=64 time=0.758 ms
64 bytes from 192.168.1.1: seq=2 ttl=64 time=0.775 ms
64 bytes from 192.168.1.1: seq=3 ttl=64 time=0.802 ms

--- 192.168.1.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.758/3.265/10.726 ms
```

10.25 Exit & Logout Command

The exit and logout command have the same function. Typing exit or logout will terminate your session and if changes were made, will cause the system to reboot and come back up with the new settings.

11 Equation Programming

Equations are defined and programmed by uploading a text file to the ESB2. The equation programming is based on Lua and conforms to the format for creating programs using the Lua program language. Lua is case sensitive therefore you must adhere to this constraint when creating programs. Where possible, the ESB2 will accept upper and lower characters for names when defining points but referencing the point in an equation must be in lowercase.

Creating equations is broken up into two sections in the configuration upload file. The first section is where you will define the equation name and initial value along with memory variables (VAR) and setpoints (SPT). This section of the program file will have the header section of [equations]. The second section for programming equations is where the actual equation is written. This section will have the heading of [equations_lua]. Both of these sections must be in your upload file in order to send equations to the ESB2. Note: This assumes you have already defined the other points in the system either through the web server or by uploading a text file with the various sections. You may choose to have a single file containing all the sections (like what the ESB2 returns when you download a file) or just the sections you are modifying. See Section 12 for additional information on creating upload files.

11.1 Defining Equation points

In your text file for uploading to the ESB2 you need to create a section called [equations]. All points in this section must be declared or defined prior to being used in an equation. The options for this section are EQU, VAR and SPT. There is not a present limit on the number of each point type that can be defined. The following is the format for each point type.

11.1.1 Defining Setpoints (SPT)

Setpoints allow a user with the appropriate credentials to change the value through the web server. The setpoints will be displayed under Status/Control Point Values/Setpoints. This tab will have a table listing all the setpoints defined listed in alphabetical order based on the id name. A change setpoint icon will be present if the user has program level credentials or higher. Setpoints are also changeable through SNMP with the correct credentials.

Format: SPT,id,name, initial value

Where:

Id = reference id 1-32 alphanumeric character unique name used to reference the point in an Equation. Since this is the name referenced in equations, it is advisable to make them as short as possible to reduce typing. For example, you could create a numerical list for id names. I.e. 1,2,3 etc. When writing equations, you can refer to the point by either the id name or the display name, but if the id name is used, you must prefix it with spt.

Name = display name. this is a 1-32 character unique name for display in the webserver and alarm transmission. You can use this name in equations. When using the display name in an equation, do not prefix with spt.

Initial value = the initial value of the point as an integer.

Example:

SPT,1,Cooling_Setpoint_for_HVAC_1, 78

SPT,2,Heating Setpoint for HVAC 1, 65

The points can be referenced in an equation as SPT.spt1 or SPT. Cooling_Setpoint_for_HVAC_1. Note if you reference a display name that has spaces, the ESB2 will assume underscores (_) in the equation.

11.1.2 Defining Memory Variables (VAR)

Variables are used for calculations and to store integer numbers. Variables are not changeable from the web server or SNMP. Use VARs when you want to store number values for equation processing, but you don't want them (or need them) to be changeable. The variables will be displayed under Status/Control Point Values/Variables. This tab will have a table listing of all the variables defined listed in alphabetical order based on the id name.

Format: VAR,id,name, initial value

Where:

Id = reference id 1-32 alphanumeric character unique name used to reference the point in an Equation. Since this is the name referenced in equations, it is advisable to make them as short as possible to reduce typing. For example, you could create a numerical list for id names. I.e. 1,2,3 etc. When writing equations, you can refer to the point by either the id name or the display name, but if the id name is used, you must prefix it with var.

Name = display name. This is a 1-32 character unique name for display in the webserver and alarm transmission. You can use this name in equations. When using the display name in an equation, do not prefix with var.

Initial value = the initial value of the point as an integer.

Example:
VAR,1,Lead_System, 1
VAR,2,Cooling Delay, 65

The points can be referenced in an equation as VAR.var1 or VAR. Lead_System. Note if you reference a display name that has spaces, the ESB2 will assume underscores (_) in the equation.

11.1.3 Defining Equations (EQU)

Equations are used to create the actual control logic for the ESB2. The status of Equations will be displayed under Status/Control Point Values/Equations. This tab will have a table listing of all the equations defined listed in alphabetical order based on the id name.

Format: EQU,id,name, initial value

Where:

Id = reference id 1-32 alphanumeric character unique name used to reference the point in an Equation. Since this is the name referenced in equations, it is advisable to make them as short as possible to reduce typing. For example, you could create a numerical list for id names. I.e. 1,2,3 etc. When writing equations, you can refer to the point by either the id name or the display name, but if the id name is used, you must prefix it with equ.

Name = display name. This is a 1-32 character unique name for display in the webserver and alarm transmission. You can use this name in equations. When using the display name in an equation, do not prefix with equ.

Initial value = the initial value of the point as an integer.

Example:
EQU,1,HVAC1 Cool Control, 0
EQU,2,HVAC1 Heat Control, 0

The points can be referenced in an equation as EQU.equ1 or EQU.HVAC1 Cool Control. Note if you reference a display name that has spaces, the ESB2 will assume underscores (_) in the equation.

11.2 Creating Equations

The [equations_lua] section of the upload file is used to create equation logic. Each equation starts with the reference id (id) as defined in the [equations] section followed by three backticks (```) then the equation logic followed by an additional three backticks to end the equation block. Each equation is a self-contained program created to do a particular function. The equation language is case sensitive. all functions are used in lower case. Specific keywords can be in uppercase. Refer to section 11.2.2 for a list of all keywords used in the system. There is no set limit to the number of available equations. A typical equation can be used to turn on or off a control point based on some conditions. For example, an equation can compare the value of an input to a setpoint and turn on or off an output based on that comparison. An equation as follows can be written to perform this function:

```
1```
if INP.uin1 > SPT.spt1 then OUTP.rly1 = ON
elseif
INP.uin1 < SPT.spt1 - 2 then OUTP.rly1 = OFF
end```
```

The following sections of the manual will provide additional information on the Lua program language and specific keywords, functions and constructs that have been created for use within the ESB.

11.2.1 Conditional Statements

One of the primary functions of equations is to create conditional statements that when true will cause an action. A condition can be a simple comparison such as an input to a setpoint, or it can contain multiple comparisons that must be true before the corresponding action can occur. Equations with condition statements take the format as follows:

```
if<condition> then <action>
elseif
<condition> then <action>
else
<action>
end
```

The function 'if' is used to tell the system that there is a comparison statement that needs to be evaluated. 'elseif' is used for subsequent conditions to be evaluated. They will be considered if the preceding condition(s) were false. 'else' is used to tell the system to do the following action statement. The action statement will be processed if all the preceding 'if' conditions were false. Finally, the 'end' function is to tell the system that it has reached the end of the condition statements for this section of the equation. The function 'end' must always be used when using conditional statements.

11.2.1.1 Conditional Statements

Conditional statements are created by using the 'if' keyword and can be made up of multiple statements by using the conjunction 'and' and 'or'. Statements with 'and' are evaluated by first looking at the condition to the left of 'and'. If that portion is true, then the condition on the right of 'and' is evaluated and must be true for the equation to move on to the action section. Statements with 'or' are evaluated left to right as well. The condition to the right of 'or' is evaluated only if the condition on the left is false.

Example:

If TOD.tod1 and INP.uin1 > SPT.spt1 then OUTP.rly 1 = ON
Time schedule 1 must be true and input 1 must be greater than setpoint 1 in order to turn on relay 1.

If TOD.tod1 or TOD.tod2 then OUTP.rly2 = ON
Relay 2 will turn on if time schedule 1 is true (on) or if time schedule 2 is true (on).

11.2.1.2 Control loops

The following are additional control loop functions that can be used in equations:

11.2.1.2.1 While function

Use the 'while' function to do a continuous action for a period of time or counts. For example,

```
VAR.counter = 0
while (VAR.counter < 20)
do
   VAR.counter = VAR.counter + 1
   wait (1)
end
```

This equation will increment VAR.counter every second until the counter reaches a value of 20 and then will stop incrementing.

11.2.1.2.2 Repeat function

The 'repeat' function will repeat an action until the value of the point no longer qualifies to cause the action to occur. For example:

```
VAR.counter = 0
repeat
   VAR.counter = VAR.counter + 1
until (VAR.counter > 20)
```

This equation will increment the value of VAR.counter without a delay until the value of the variable is greater than 20.

11.2.1.2.3 For function

The 'for' function will repeat an action starting at an initial value to the maximum value by incrementing the variable by a specific amount. The three values that are defined when using the 'for' function are the initial value, stop value, increment value.

Example:

```
VAR.counter = 0
for i = 0,10,1 do
   VAR.counter = VAR.counter + i
end
```

This equation will increment VAR.counter by 1 starting at a value of 0 and ending when VAR.counter has a value of 10. The point will be incremented every time through the loop by a value of 1.

11.2.2 Equation Keywords

The ESB2 supports certain keywords within the Lua program language that are unique to the ESB2 application. The keyword is in ALL Caps and will prefix the point ID being used. For example, SPT is the keyword for Setpoint and when used in an equation it will have the form of SPT.x where x equals either the point id name (with spt prefix) or the display name (see section 11.1.1). For example, a set point defined as: SPT,1,Cooling_Setpoint_for_HVAC_1, 78 can be referenced in equations as either SPT.spt1 or SPT.Cooling_Setpoint_for_HVAC_1

Below is a list of keywords, a description of their function and point data type:

Keyword	Description	Read/Write	Data Type
EQU	Equation numeric value plus enable/disable.	R/W	Integer
VAR	Memory Variable numeric value	R/W	Integer
SPT	Setpoints numeric value	R/W	Integer
INP	Universal input reading. Format INP.uinx where x = the input number.	R/O	Float
OUTP	Relay outputs (rlyx) and Analog output (aopx). Used to read status or to set a value.	R/W	Integer

MODP	Modbus points, use the 4-character reference ID used to read the modbus point value or to write to the modbus register.	R/W	Integer or Float
TOD	Time of day schedule status	R/O	Boolean
DAT	Alternate Date schedules status	R/O	Boolean
ALM	Alarm program status. Use 4-character id for alarm points or busABC where A = q for qmbus board, s for syslink (main io board), 1 for modbus serial port 1 and 2 for serial port 2. BC = hex address for qmbus and modbus addresses. For syslink use lc	R/O	Boolean
RAC	Relay Accumulated On time (in seconds) for rlyx	R/O	Integer
RLO	Relay Interval On time (in seconds) for rlyx	R/O	Integer
RLF	Relay Interval OFF time (in seconds) for rlyx	R/O	Integer
RTM	Relay Total time (in seconds) since timers were cleared for rlyx	R/O	Integer
REC	Relay Event Counter for rlyx	R/O	Integer
DAC	Digital Input Accumulated On time (in seconds) for uinx	R/O	Integer
DNO	Digital Input Interval On time (in seconds) for uinx	R/O	Integer
DNF	Digital Input Interval OFF time (in seconds) for uinx	R/O	Integer
DTM	Digital Input Total time (in seconds) since timers were cleared for uinx	R/O	Integer
DEC	Digital Input Event Counter for uinx	R/O	Integer
ON	Can be used in comparisons or assignments for the value of non-zero (true, 1, etc.)	N/A	N/A
OFF	Can be used in comparisons or assignments for the value of zero (false, 0, etc.)	N/A	N/A
LTIME	Local Time – used to return the appropriate pieces of the system clock	R/O	Integer
LTIME.HOURs	Shows the current hour 0-23 of the system clock	R/O	Integer
LTIME.MINUTES	Shows the current minutes 0-59 of the system clock	R/O	Integer
LTIME.SECONDS	Shows the current seconds 0-59 of the system clock	R/O	Integer
LTIME.DAY	Shows the day of the month 1-31	R/O	Integer
LTIME.MONTH	Shows the month of the year 1-12	R/O	Integer
LTIME.YEAR	Shows the current year of the system clock yyyy	R/O	Integer
LTIME.DATE	Shows the current date mm/dd/yyyy	R/O	Integer
S	Seconds conversion function to convert time to seconds. Format: S(“DD:HH:MM:SS”), S(“HH:MM:SS”), S(“MM:SS”). Example: S (“01:05”) would = 65 seconds	R/O	Integer

11.2.3 Point types and reference

The following table lists the available point types, references, and keywords for use in equations. When referencing a point in an equation, you will use a combination of the keyword and point reference. For example, Universal Input 1 would be referred in an equation as INP.uin1.

Point Type	Keyword	Point Ref
Universal inputs 1-272	INP	uinx (where x = 1-272)
Relay outputs 1-144	OUTP	rlyx (where x = 1-144)

Analog Outputs 1-68		aopx (where x = 1-68
Modbus points	MODP	4-character ID
Time of Day schedules	TOD	tod + schedule number
Date schedule	DAT	dat + schedule number
Alarm programs	ALM	4-character id
Set points	SPT	spt + reference id or display name
Memory Variables	VAR	var + reference id or display name
Equations	EQU	equ + reference id or display name

11.2.4 Equation Functions

The ESB2 has unique functions that have been created to work with the Lua program language. This section will provide information on those functions and how to use them in an equation.

11.2.4.1 Conditional Loop Timer

The function 'timerExp' is used to look for a specific condition to be true for a period of time. The function is made up of two parts. The first part is the condition or conditions that need to be true and the second part is the time in seconds that the condition must be true for in order for the function to be true and allow the control sequence.

Format: timerExp ({<condition>, <time>})
Where:
<condition> = the condition to be evaluated
<time> = the number of seconds the condition has to be true to proceed. This value could be a SPT or VAR instead of static number of seconds.

Example 1: if timerExp({INP.uin1 > SPT.spt1, 120}) then OUTP.rly1 = ON
Example 2: if timerExp({INP.uin1 > SPT.spt1, VAR.var1}) then OUTP.rly1 = ON

This line of an equation will compare input 1 to setpoint 1 and if the input is greater than the setpoint for 120 seconds then the equation will proceed to turn on relay 1.

11.2.4.2 Static Wait

The function 'wait' will pause the processing of the equation for a number of seconds. This can be a static number i.e. 60 for 60 seconds or it can be the value of a VAR or SPT.
Format: wait (<time>)
Where:
<time> = the number of seconds or a SPT or VAR containing the number of seconds.

Example 1: if INP.uin1 > SPT.spt1 then wait (60) OUTP.rly1 = ON
Example 2: if INP.uin1 > SPT.spt1 then wait (VAR.var1) OUTP.rly1 = ON
In this case when input 1 is greater than the setpoint, the equation will wait the time delay and then do the action.

11.2.4.3 Wait Until

The function 'waitUntil' will pause the equation from processing further until the conditions are met. This way you can have an equation do an action and then wait until a condition is true before doing further actions in that equation.

Format:
waitUntil(function() return < condition > end)
Where:
<condition> = the condition that must be true before continuing the processing of the equation.

Example1:
if timerExp({INP.uin1 > SPT.spt1, 120}) then OUTP.rly1 = ON waitUntil (function() return timerExp({OUTP.rly1 == ON, 180}) end) OUTP.rly2 = 1

This equation statement will check to see if input 1 is greater than setpoint 1 for 120 seconds and then turn on relay 1. It will then wait until relay 1 is on for 180 seconds and then turn on relay 2.

Example2:
if (TOD.tod2 and VAR.var4 == 1)
then VAR.var4 = 2 setEquVal (2) waitUntil(function()return TOD.tod2 == false end)

This equation statement will check to see if TOD schedule 2 is true (ON) and if variable 4 is equal to 1. It will then change variable 4 to a value of 2 , set the equation value equal to 2 and wait until TOD schedule 2 is off before resuming.

11.2.4.4 Equation Enable

The function 'enable' allows an equation to enable another equation. An enabled equation means that is running and not prevented from operating (see disable). Typically, this function would be used to enable the operation based on certain conditions.

Format: enable ("<refname>")
Where <refname> = the reference id of the equation to be enabled. You can use the prefix 'equ' or omit it and just use the id name.

Example:
Define an equation such as: EQU, e1, Control Equation, 0
In an equation to enable the function use: enable ("e1") or enable ("eque1"). Since equation names are string values, you must use quotation marks.

11.2.4.5 Equation Disable

The function 'disable' allows an equation to disable another equation. A disabled equation is prevented from operating until the equation is enabled again. Note: all equations are automatically enabled on reset or power failure.

Format: disable ("<refname>")
Where <refname> = the reference id of the equation to be disabled. You can use the prefix 'equ' or omit it and just use the id name.

Example:

Define an equation such as: EQU, e1, Control Equation, 0
In an equation to disable the function use: disable ("e1") or disable ("eque1"). Since equation names are string values, you must use quotation marks.

11.2.4.6 Set Equation Value

The function 'setEquVal' is used to set the value of the equation to a particular value. This would be the same as using EQU.equ<refname> = <value> but allows the equation to change its value without having to reference the equation name.

Format: setEquVal (<value>)
Where: <value> = the integer value to be assigned to the equation. Instead of a static value you could use the value of an input, VAR, EQU or SPT.

Example1: setEquVal (1)
Example2: if INP.uin1 > SPT.spt1 then OUTP.rly1 = ON setEquVal (INP.uin1)

11.2.4.7 Averaging Function

The function 'math.avg' was created in the ESB2 to allow averaging of multiple points and then return a floating point number. For example, you can average multiple sensors and use the average of those sensors for your control sequence. Note if you have a SPT or VAR equal the value of 'math.avg' then the value will be converted to an integer by truncating everything to the right of the decimal point. There must be a minimum of two arguments separated by commas to use this function. It is recommended to not use more than ten arguments though there is no arbitrary limit.

Format: math.avg (argument1, argument2, …argument10)
Example1: math.avg (INP.uin1, INP.uin2)
Example 2: define a variable to equal the average: VAR, 1,Sensor Average, 0
VAR.var1 = math.avg(INP.uin1, INP.uin2)
The average is converted to an integer since VAR points are integers.

11.2.4.8 Mathematical Functions

The ESB2 supports basic Lua mathematical functions. The pertinent math functions are listed in this section. See the Lua version 5.3 reference manual for further details or a full list of all math functions.

math.abs (x)
Returns the absolute value of x. (integer/float)

math.exp (x)
Returns the value e^x (where e is the base of natural logarithms).

math.fmod (x, y)
Returns the remainder of the division of x by y that rounds the quotient towards zero. (integer/float)

math.max (x, ···)
Returns the argument with the maximum value, according to the Lua operator <. (integer/float)

math.maxinteger
An integer with the maximum value for an integer.

math.min (x, ···)
Returns the argument with the minimum value, according to the Lua operator <. (integer/float)

math.mininteger
An integer with the minimum value for an integer.

math.modf (x)
Returns the integral part of x and the fractional part of x. Its second result is always a float.

11.2.4.9 Local Variables

A local variable can be defined and used within an equation. Local variables are present only to the equation they are created in and can be used as a convenience when writing equations. The value of local variables is not shown on the web server like VAR and SPTs. An example of using a local variable would be if you are comparing the outcome of some mathematical expression and you want to reference a local variable throughout the equation vs. having to write the mathematical expression multiple times.

Format: (in an equation) local <variablename> = <argument>
Where:
<variablename> = the name you want to assign to the variable
<argument> = the argument the local variable is to be equal to.

Example:

```
Equ3```
local avgtemp = math.avg(INP.uin1, INP.uin2)
If avgtemp > SPT.cooling then OUTP.rly1 = ON
elseif
avgtemp < SPT.cooling then OUTP.rly1 = OFF
end```
```

11.2.4.10 Comparators

The ESB2 uses comparators to create control sequences in equations. The comparators are used to compare one condition to another to determine truth statements. Points being compared can be integers or floating point numbers.

==	Equal to
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
~=	Not equal to

Note == (equal to) and <> (not equal to) should only be used with integer numbers since the comparison has to be exact in order to be true.

Example: if INP.uin1 > SPT.cooling then …

11.2.4.11 Boolean Operators

Certain points are Boolean operators and are treated differently when doing comparison since Boolean variable points can only be True or False. Time of day (TOD), date schedules (DAT) and alarm (ALM) status points are examples of Boolean variable points. Refer to the table in section 11.2.2 for point details.

Use the following format only when referencing Boolean points. Any other format such as "== ON" will not operate as intended:

If <point> – true when the Boolean variable is true (1, ON).
If not <point> – true if the Boolean variable is false (0, OFF).
If <point> == true – true when the Boolean variable is true.
If <point> == false – true when the Boolean variable is false.

Example:
ltson```
If TOD.tod1 then OUTP.rly16 = ON
elseif
not TOD.tod1 then OUTP.rly.16 = OFF
end```

almout```
if ALM.a001 == true then OUTP.rly10 = ON
elseif
ALM.a001 == false then OUTP.rly.10 = OFF
end```

The above equations can be further shortened because the variables being evaluated are Boolean and can only be true or false. IF you look for true then when it is not true, you can have the equation do the action without further comparison.

ltson```
If TOD.tod1 then OUTP.rly16 = ON
else
OUTP.rly.16 = OFF
end```

almout```
if ALM.a001 == true then OUTP.rly10 = ON
else
OUTP.rly.10 = OFF
end```

11.2.4.12 Networked Points

Points that are considered network points are those points that are in addition to the points on the main IO board. Such as points on Quest expansion boards or Modbus points. Under normal conditions, the value of the point being displayed is the value that will be used in equations. When an expansion board or Modbus device is not communicating, the status value shown will be error. In equations though the

value of the point will be evaluated as 0. Normally this is not an issue since if the board is in error, then no control can be affected from equations. If you are passing a value from one expansion point to another, say from one Modbus device to another, then you could add a check to ensure the Modbus device where the point is being read from is not in error. For example:

```
If ALM.bus101 then return 1
else
MOD.tmp2 = MOD.tmp1
end
```

This equation checks to see if the Modbus device on serial port 1 address 01 is in alarm. If it is in alarm, the equation will return 1 and not proceed. When the device is not in error, the equation will allow the Modbus point with reference id 'tmp2' equal the Modbus point with reference id tmp1.

The same function can be used to check the status of ESB2 expansion modules. See section 11.2.2 for the keyword and format of bus alarms.

11.2.5 Commenting Equations

Equations support adding comment lines to document information about what the equation is for or any detail that might be necessary for trouble shooting later on. The system supports single line comments or multiple lines of comments. The multi-line function can also be used to temporarily comment out a section of an equation for troubleshooting. The options are:

Two dashes (--) will allow comment text until the end of the line.

Example:

```
1```
-- This equation is to control the fan on relay 2
if SPT.spt10 = 1 then OUTP.rly1 = ON
elseif
OUTP.rly2 == ON or OUTP.rly3 == ON then OUTP.rly1 = ON
else
wait (30) OUTP.rly1 = OFF
end
```
```

Two dashes plus two sets of brackets (--[[ ]]) will allow multiple lines of comments for documentation or to prevent a section of the equation from operating.

Example 1:

```
1```
--[[This equation is to control the fan on relay 2
It uses setpoint 10 to determine which HVAC unit should be the lead system]]
 if SPT.spt10 = 1 then OUTP.rly1 = ON
elseif
```
```

```
OUTP.rly2 == ON or OUTP.rly3 == ON then OUTP.rly1 = ON
else
wait (30) OUTP.rly1 = OFF
end
```

Example 2: Prevents the elseif statement from operating if commented out.

```
-- This equation is to control the fan on relay 2
if SPT.spt10 = 1 then OUTP.rly1 = ON
--[[elseif
OUTP.rly2 == ON or OUTP.rly3 == ON then OUTP.rly1 = ON]]
else
wait (30) OUTP.rly1 = OFF
end
```

11.3 Equation Examples

This section will provide examples of various control equations along with the memory variables that need to be defined. Defining of other points such as inputs, outputs, Modbus, time schedules and date schedules are covered separately, and it is assumed the user has an understanding of how to define those points and has previously defined the points.

11.3.1 Calculate Control Temperature

The equation will check to see if input 1 or input 2 is in error. If one of the sensors is in error, then the other sensor will be used as the controlling temperature value. If both sensors have good readings, then the average of the two sensors will be used for the controlling temperature.

Data to be included in the upload file:

```
[equations]
EQU, 4, Calculate Controlling Temperature, 0
VAR, 2, Controlling Temperature, 0

[equations_lua]
4```
if timerExp({INP.uin1 > 200 or INP.uin1< -100 ,60}) then VAR.var2= INP.uin2
elseif
timerExp({INP.uin2 > 200 or INP.uin2 < -100, 60})  then VAR.var2= INP.uin1
else
VAR.var2= math.avg ( INP.uin1, INP.uin2 )
end
```

11.3.2 Fan Control

The following equation will turn the supply fan on if the HVAC unit is the lead system. If it is not the lead system, then the fan will turn on when there is a call for cooling or heating. The fan will be turned off after 30 seconds if the system is not the lead and there is not a call for heating or cooling.

Data to be included in the upload file:
[equations]
EQU, 1, Fan Control, 0
SPT, 10, Lead System, 1

[equations_lua]
1```
if SPT.spt10 == 1 then OUTP.rly1 = ON
elseif
OUTP.rly2 == ON or OUTP.rly3 == ON then OUTP.rly1 = ON
else
wait (30) OUTP.rly1 = OFF
end
```

### 11.3.3 Cooling Control

This equation will compare input 1 to a setpoint and if the input is greater than the setpoint for two minutes (120 seconds) then the equation will turn on output 1. Output 2 will stay on until the input is below the setpoint minus an off delta for 120 seconds and then turn off the output.

Data to be included in the upload file:
[equations]
EQU, 2, Cooling Control, 0
SPT, 1, Cooling Setpoint, 78
SPT, 2, Cooling Off Delta, 4

[equations_lua]
2```
if timerExp ({INP.uin1 > SPT.spt1, 120}) then OUTP.rly2 = ON
elseif
timerExp ({INP.uin1 < SPT.spt1 – SPT.spt2, 120}) then OUTP.rly2 = OFF
end
```

11.3.4 Heating Control

The heating control equation below will look at the calculated control temperature (see 11.3.1 above) to determine when to turn heating on. When the control temperature is below the heating setpoint for the delay time, the heat will be turned on.

Data to be included in the upload file:
[equations]
EQU, 5, HVAC 1 Heat, 0
SPT, 3, Heating Setpoint, 65
VAR, 2, Controlling Temperature, 0
VAR,3, HVAC1 Heat Setting, 65
VAR,4, Lead System, 1
[equations_lua]

5```
if timerExp({((VAR.var2 > 200) or (VAR.var2 < -100)), 60}) then
OUTP.rly3 = OFF setEquVal (4)
elseif
timerExp({(VAR.var2 < VAR.var3),180}) then
OUTP.rly3 = ON setEquVal (1)
elseif
timerExp({VAR.var2 > (SPT.spt3 + 2),120}) then
OUTP.rly3 = OFF setEquVal (0)
end
```

### 11.3.5 Setpoint Limits

This equation will define a maximum and minimum value for setpoints to prevent a user from changing the value of the setpoint beyond defined limits.

Data to be included in the upload file:
[equations]
EQU, 3, Setpoint Limits, 0
SPT, 1, Cooling Setpoint, 78
SPT, 2, Cooling Off Delta, 4
SPT, 3, Heating Setpoint, 60

[equations_lua]
3```
SPT.spt1 = math.max(68,math.min(82,SPT.spt1)) SPT.spt2 = math.max(2,math.min(8,SPT.spt2))
SPT.spt3 = math.max(40,math.min(65,SPT.spt3))
```

12 File upload format

The ESB2 supports complete programming by uploading a file through the web server. The file is broken down into multiple segments and each starts with a heading in brackets. It is permissible to upload only certain sections at a time, but when uploading a section, all the options for that section must be sent. For example, if you send the [network] section, you must send all items in that section. For application programs sections such as [alarms], you must send all alarm programs since the new uploaded alarm section would replace all alarms that were in the unit. Some sections such as [alarms] or [equations_lua] require that the reference points (inputs, Modbus etc.) must be defined first. Therefore, they either have to be in the same file during the upload or have previously been defined.

The configuration settings in each section are comma separated values. You can use a text editor or a CSV application program to edit and or create the entire file or individual sections.

To become familiar with the file format, it is recommended that you download the file from the ESB2 by going to Settings/Advanced/Configuration Upload & Download and select "Download Configuration to browser". A text file will be downloaded from the system to your web browser.

12.1 Section Formats

Listed below are the various sections in the configuration file along with an explanation of what the section is used for and how it correlates to programming through the web server.

12.1.1 [network]

Contains the network settings that are configured through the Settings/General section of the web server for IPv4, IPv6 and IP General sections.

12.1.2 [site]

Site specific information from the Settings/General/Site Parameters section. In addition, there are fields for latitude, longitude, and URL text information that are for future use and should not be changed.

12.1.3 [timeserver]

Information for setting the system clock which is configured through Settings/General/Time Settings.

12.1.4 [timezone]

The time zone used for the system which is configured through Settings/General/Time Settings.

12.1.5 [snmp]

The SNMP general settings are configured through Settings/General/ "SNMP/SMTP General" Settings. This section also contains the heartbeat trap configuration which can be set through the web server Settings/Alarm Dispatch/Heartbeat Configuration section.

12.1.6 [smtp]

The general email settings that are configured through Settings/General/ "SNMP/SMTP General".

12.1.7 [email]

This section will contain the email dispatch configuration that is created through the Settings/Alarm Dispatch/Email Dispatch tab. The severity mask is as follows:

62 = All severities
56 = Critical, Major & Minor
48 = Critical and Major
32 = Critical only
0 = Disabled

12.1.8 [log]

This section is for internal use and should not be changed unless directed to do so by Quest Controls, Inc. personnel. Changes to this section can impact the performance of your system or render it inoperable.

12.1.9 [snmpv3]

This section contains the SNMPv3 settings that are entered through the web server Settings/General/SNMPv3 User Configuration tab.

12.1.10 [radius]

Settings for the RADIUS server connectivity that are entered through the web servers Settings/General/Radius Configuration tab.

12.1.11 [telnetssh]

Settings used for Telnet and SSH access that are entered through the web servers Settings/General/Telnet/SSH User Configuration.

12.1.12 [port]

The serial port settings for use with Modbus and alarm severities for communications. These settings are from the web servers Settings/General/Serial Ports tab.

12.1.13 [app_passwords]

The four-digit passcodes used for the color touchscreen on the front of the ESB.

12.1.14 [qmbus]

This section is for adding Quest expansion modules to the system. There will be a line for each board being defined that consists of the expansion board type and the address. Expansion boards can be added through the web server at the Program/Qmbus Module page.

<u>Options:</u>
#type = (uib|dob|aob),
addr = (UIB 0x60-0x6F (96-111); DOB 0x00-0x0F (0-15); AOB 0x70-0x7F (112-127))

<u>Example:</u>
[qmbus]
#type (uib|dob|aob),addr (UIB 0x60-0x6F (96-111); DOB 0x00-0x0F (0-15); AOB 0x70-0x7F (112-127))
dob, 1
uib,96

12.1.15 [inputs]

This section is for defining inputs that are connected to the ESB main board or Quest expansion boards. See Section 9.1 for additional information on defining inputs.

Options:

#pt = The point number where the point is connected to the system. from 1-272
Name = The 32-character display name defined for the point
Type = tempf, tempc, dryno, drync, wetno, wetnc, custom
Offset = offset value to correct the sensor reading.
custom low = When input is defined as custom this value is the low end of the scale.
custom high = When input is defined as custom this value is the high end of the scale.
log type = "COS" for change of state or "time" for interval time logging
log interval (min) = number of minutes for interval time logging or 0 when using COS.

Example:

[inputs]
#pt,name,type,offset,custom low,custom high,log type,log interval (min)
1,Outside Air,tempf,0,0,0,time,5
2,Zone 1,tempf,0,0,0,time,5
3,Humidity,custom,0,0,139,time,10

[outputs]

This section is for defining outputs that are tied to the ESB main board or Quest expansion boards. See Section 9.2 for additional information on defining outputs.

Options:

#pt = The point number where the point is connected to the system. from 1-144
Name = The 32-character display name defined for the point
default state = 0 for De-energized, 1 for Energized
follows = equations for equation control, CR-alarms for critical alarms, MJ-alarms for major alarms, MN for minor alarms, alarms for all alarms or TOD# where # is the schedule number.
ack input = for outputs following alarms add the input ID number i.e., UIN16

Example:

[outputs]
#pt,name,default state,follows,ack input
1,HVAC1_Fan,0,equations,
2,HVAC1_Cooling,0,equations,
3,HVAC1_Heating,0,equations,
9,Alarm_Output 9,1,CR-alarms,uin16
10,Lighting_Control,0,TOD1,

12.1.16 [analog_outputs]

This section is for defining analog outputs that are tied to the ESB main board or Quest expansion boards. See Section 9.3 for additional information on defining outputs.

Options:

#pt = The point number where the point is connected to the system. from 1-66
Name = The 32-character display name defined for the point
default % = 0-100 for the percentage output before an equation sets the value
converted low = the display reading when the output is at 0%
converted high = the display reading when the output is at 100%

Example:

```
[analog_outputs]
#pt,name,default %,converted low, converted high
1,HVAC1_Econ,0,0,100
2,HVAC2_Econ,0,0,100
```

12.1.17 [modbus1]

This section is for defining all the Modbus points that are connected to serial port 1. Each Serial port can support up to 512 Modbus registers. The section is made up of two parts. Part 1 is to define the Modbus device and part 2 is used for defining the Modbus points. See section 9.4 for additional information on defining Modbus devices and points.

Options for Modbus device:

#SN = prefix used to indicate a Modbus device is being defined.
address = the address number the device is configured to respond to.
Name = a 32-character display name for the device.

Options for Modbus points:

#MD = prefix used to indicate a Modbus device is being defined.
Id = the four-character unique id assigned to the point.
display name= a 32-character display name for the device.
Address = the address of the device where the point can be found
function code = the function code number used to gather the data 1-4
register = The register number used to request the data.
Bit = either 0 to use the entire register or 1-15 to use a single bit of the register.
Format = the type of register: byte, word, dword, sword or float.
Offset = offset value to calculate or correct the reading.
Multiplier = the multiplier used to convert the register reading.
log type = time, cos or none.
log interval (min) = minutes for interval logging or 0 when using cos or none.

<u>Example:</u>
[modbus1]
#SN, address, name
SN,1,HVAC controller
#MD,id,display name, address,function code,register,bit,format,offset,multiplier,log type,log interval (min)
MD,H111,onboard Zone Temp float,1,4,200,0,float,0,1,time,5
MD,H112,Remote Zone Temp float,1,4,202,0,float,0,1,time,5

12.1.18 [modbus2]
This section is for defining all the Modbus points that are connected to serial port 2. Refer to 12.1.17 above for details on defining points.

12.1.19 [schedules]
The schedules section is for the time schedule programming. There are eight time schedules and each schedule can have up to sixteen priorities. See section 9.8 for additional information on defining schedules.

<u>Options:</u>
#sched = The schedule number 1-8
Name = a 32-character name display name for each schedule.
Action-day mask 1 (action=bit7/su=bit0..sa=bit6) = The list of days or an alternate date for the priority line along with the action bit where 1 = on and 0 = off. Ie ON 7 days a week would be the first eight bits equal to 1 (11111111) which is 255 in decimal. Date Schedules use an additional byte where date schedule 1 = bit 7.
time (0-1435) = Time in minutes the action is to take place with 0 = midnight.

<u>Example:</u>
[schedules]
1,Lead_Switch,130,420,2,425, 32896,60,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Priority 1 is ON Monday at 7 am, Priority 2 is OFF Monday at 7:05 am and Priority 3 is ON when Date 1 is on(active) and the time is 1am.

12.1.20 [dates]
The dates section is for the alternate date programming to set action based on a calendar date. See section 9.7 for additional information on creating dates.

<u>Options:</u>
#date = The date schedule number
Name = a 32-character name display name for each schedule.
first date = the first date the schedule becomes active mm/dd
conjunction = and, to or none
second date = the second date the schedule is active mm/dd

Example:
[dates]
#date, name, first date, conjunction, second date
1,DateSchedule,1/1,to,1/30

12.1.21 [alarms]

The alarms section will contain all defined alarm conditions for inputs, Modbus points and control point types. See section 9.6 for additional information on defining date schedules.

Options
#Id = the four-character unique id assigned to the point.
display name= a 32-character display name for the device.
data id = the id number of the point used to create the alarm condition i.e., UIN1, M001 (Modbus id#)
severity = critical, major, minor, notice, warning, none
comparison = greater_than, less_than, equal_to, not_equal_to
value = value to compare the data id point with to determine the alarm condition
qualifier = additional qualifier such as a time schedule or another alarm program. The field is blank if not used.
alm delay = delay in seconds before the alarm is active when the comparison is true
clr delay = clear delay in seconds to wait after the condition is no longer true before clearing
nag = nag interval in minutes
tl1 type = env, eqptsa, eqptnsa

Example:
[alarms]
#id,disp name, data id, severity, comparison, value, qualifier, alm delay, clr delay,nag, tl1 type
a001,Testing an alarm for input 15,uin15,major,greater_than,0.000000,,60,5,0,env

12.1.22 [equations]

See section 11.1

12.1.23 [equations_lua]

See sections 11.2 - 11.3

12.2 Uploading config files.

once the upload file has been created go to the Settings/Advanced/Configuration Upload & Download tab. Click on the Choose File option to select the file from your computer. The click on the Upload Configuration from browser button. To load the file. The system will upload the file and then process all of the sections. If there are no errors the system will indicate a successful upload and ask you to click on the reboot button. Any errors found will be listed out in the upload status window. Correct the errors and then resend the file. See section 8.4.2 for additional information.

13 SNMP Sets & Gets

The ESB2 can be managed using SNMP Sets to review data and Gets to make changes. Contact Quest Controls, Inc. to obtain a copy of the product MIB. The System uses Quest's standard Alarms MIB for sending traps (see Appendix A - SNMP Trap Format). This section will focus on using the product MIB for gathering data.

13.1 Public SNMPv2

The ESB2 supports the standard v2-MIB OIDS for system branch. sysDescr, sysObjedID and sysUpTime are read only while sysContact, sysName and sysLocation support read and write.

13.2 Input table

The input table will contain all the defined inputs that are physically wired to the base ESB2 as well as all inputs that are attached to Universal Input Boards (UIB). The table index will the integer input number for each point. The table has the following objects available:

Object	Description
Input name	The defined 32-character name for the input
Input integer value	The input reading as an integer. Digital inputs will show 1 for On and 0 for Off
input integer value times 100	The input reading as an integer in 100ths. Divide the analog input value by 100 to convert the value. Digital inputs will show 1 for On and 0 for Off
Input string value	The Input value as a string value. Analog inputs will display values to the 10th. Digital inputs will display ON and OFF
Input causing alarm	Shows if the point is causing an alarm point to be active.
Input type	Shows the input definition type. I.e., Dry contact, thermistor, custom etc.
Input offset	For analog sensors this will display the offset value defined for the point
Input custom low	The defined low value for an input defined as CUSTOM. Used with the Custom High value to scale the sensor reading.
Input custom high	The defined high value for an input defined as CUSTOM. Used with the Custom low value to scale the sensor reading.
Input log type	The type of logging defined for the input. Either interval logging based on time or change of state (COS).
Input log interval	For inputs defined as interval logging this will be the number of minutes between log entries.
Input accumulated on time	Total accumulated on time for inputs defined as digital
Input interval on time	The interval on time for inputs defined as digital
Input interval off time	The interval off time for inputs defined as digital
Input event counter	The number of cycles (on/off events) for inputs defined as digital
Input event timer	The amount of time that has elapsed since all the timers were cleared
Input remote board address	The decimal address for the input expansion board that the input is located on. For inputs on the main IO board the value will be -1
Input remote board position	The physical location of the input on the remote board.

13.3 Alarm point table

The alarm point table will contain all the defined alarm points in the system. The alarms that have an "@" as the first character of the index are alarms that get created automatically when adding expansion modules or Modbus devices. The table index is a four-character ASCII index based upon the user defined point ID number. If your client doesn't support ASCII index, then use the decimal equivalent to

the ASCII characters. The Index number format will be the number 4 plus the decimal value for each character. For example, an alarm ID of a001 will have an ACII table index of a001 or the decimal value of 4.97.48.48.49. The table has the following objects available:

Object	Description
Alarm point name	The defined 32-character name for the alarm
Alarm point state	Either active meaning in alarm or the word clear meaning not in alarm.
Alarm point timestamp	The time of the last event for the point. Ie if in alarm then that is the time the alarm occurred or if clear, that is the time the clear occurred
Alarm point severity	The defined alarm severity for the alarm
Alarm point input name	The name of the input or point that is used to determine an alarm condition.
Alarm point string value	A string value showing the status of the point being monitored.
Alarm point comparison	The comparison type used for creating the alarm condition
Alarm point comparison value	The value that the input point is being compared to when determining an alarm condition.
Alarm point alarm delay	The delay in seconds that the comparison must be true before the alarm is active.
Alarm point clear delay	The delay in seconds that the comparison must no longer be true before the alarm is cleared.
Alarm point nag interval	The value in minutes that will cause an alarm dispatch if the alarm point is still active. A value of 0 means no nag function.
Alarm point tl1 type	The TL1 alarm type. env, eqptsa or eqptnsa

13.4 Modbus point table

The Modbus point table will contain all the defined Modbus points in the system. The table index is a four-character ASCII index based upon the user defined point ID number. If your client doesn't support ASCII index, then use the decimal equivalent to the ASCII characters. The Index number format will be the number 4 plus the decimal value for each character. For example, a Modus ID of b001 will have an ACII table index of b001 or the decimal value of 4.98.48.48.49. The table has the following objects available:

Object	Description
Modbus point name	The defined 32-character name for the input
Modbus integer value	The register reading as an integer. Digital points will show 1 for On and 0 for Off. This object is writable and will allow SNMP Gets to write to the Modbus register. It is up to the Modbus device to accept the write.
Modbus integer value times 100	The reading as an integer in 100ths for analog points. Divide the value by 100 to convert the value. Digital inputs will show 1 for On and 0 for Off. This object is writable and will allow SNMP Gets to write to the Modbus register. It is up to the Modbus device to accept the write.
Modbus String Value	The Input value as a string value.
Modbus point causing alarm	Shows if the point is causing an alarm point to be active.
Modbus point address	The Modbus address for the device that point is associated with
Modbus point register	The register number for reading the point from the Modbus device
Modbus point bit	Specifies to look at a single bit in the defined register. A value of 0 means to read the entire register.
Modbus point format	The format type for reading the register from the device.
Modbus point offset	The defined offset value applied to the register to correct the reading

Modbus point multiplier	The multiplier value used to convert the Modbus register to desired engineering units.
Modbus log type	The type of logging defined for the point. Either interval logging based on time or change of state (COS).
Modbus log interval	For points defined as interval logging this will be the number of minutes between log entries.
Modbus point function code	The defined Modbus function code used to read the point from the device.
Modbus point port number	The serial port used on the ESB2 to talk to the Modbus device
Modbus device name	The defined name of the Modbus device where the point is located
Input event counter	The number of cycles (on/off events) for inputs defined as digital
Input event timer	The amount of time that has elapsed since all the timers were cleared
Input remote board address	The decimal address for the input expansion board that the input is located on. For inputs on the main IO board the value will be -1
Input remote board position	The physical location of the input on the remote board.

13.5 Output table

The output table will contain all the defined outputs that are physically wired to the base ESB2 as well as all outputs that are attached to Digital Output Boards (DOB). The table index will show the integer output number for each point. The table has the following OIDs available:

Object	Description
Output name	The defined 32-character name for the output
Output logical state	The current logical state of the output. Note that an Active will be the inverse of the default state
Output control mode	The current control mode of the output. Values here will either be: Auto, bypOff (time left), bypOn (time left Note that the units of time are seconds and a time of FOREVER may be displayed This object is writable to create a bypass by using bypOn <seconds|forever>, bypOFF <seconds|forever> or auto
Output default state	The state the relay will be in when not commanded to be ON or Active.
Output follows	The defined use of the output. This can be an alarm severity, TOD schedule or equations
Output acknowledge input	For outputs following an alarm severity there can be an input used to acknowledge the alarm condition and turn off the output. You can also write the word “ack” to remotely acknowledge an alarm condition.
Output remote board address	The decimal address for the output expansion board that the point is located on. For outputs on the main IO board the value will be -1
Output remote board position	The physical location of the output on the remote board.

13.6 Active alarm point table

The active alarm point table will display any alarm that is currently in the active state. The alarms that have an “@” as the first character of the index are alarms that get created automatically when adding expansion modules or Modbus devices. The table index is a four-character ASCII index based upon the user defined point ID number. If your client doesn’t support ASCII index, then use the decimal equivalent to the ASCII characters. The Index number format will be the number 4 plus the decimal value for each character. For example, an alarm ID of a001 will have an ACII table index of a001 or the decimal value of 4.97.48.48.49. The table has the following objects available:

Object	Description
Active alarm point name	The defined 32-character name for the alarm
Active alarm point timestamp	The time when the point went into the active condition
Active alarm point severity	The defined alarm severity for the alarm
Active alarm point input name	The name of the input or point that is used to determine the alarm condition.
Active alarm point string value	A string value showing the status of the point in alarm.

13.7 Analog output table

The analog output table will contain all the defined analog outputs that are physically wired to the base ESB2 as well as all points that are attached to Analog Output Boards (AOB). The table index will show the integer output number for each point. The table has the following objects available:

Object	Description
Analog output name	The defined 32-character name for the point
Analog output percent value	0-100 indicating the percentage output status of the point
Analog output integer value	The scaled value of the output based on the percentage output value
Analog output integer times 100	The scaled value of the output based on the percentage output value in hundredths.
Analog output string value	The scaled value of the output based on the percentage output as a string value
Analog Output Control Mode	Shows either Automatic or bypassed with the time remaining (or forever). Write % output <sp> time (or forever) to bypass the output. Write the word auto to clear a bypass.
Analog Output remote board address	The decimal address for the output expansion board that the point is located on. For outputs on the main IO board the value will be -1
Analog output remote board position	The physical location of the output on the remote board.

13.8 Setpoint table

The setpoint table will contain all the user defined setpoints in the system. The table index is an up to 32-character ASCII index based upon the user defined point ID name. If your client doesn't support ASCII index, then use the decimal equivalent to the ASCII characters. The Index number format will be the number 4 plus the decimal value for each character. For example, a setpoint ID of s001 will have an ACII table index of s001 or the decimal value of 4.73.48.48.49. The table has the following objects available:

Object	Description
Setpoint integer value	The integer value of the setpoint. Sets to this OID will change the current value being used for the setpoint
Setpoint display name	The 32-character defined display name for the setpoint.

13.9 Scalar values

This section of the MIB contains the firmware version for the system. This is a string value that will contain the current firmware version of the main processor board and the main IO board.

14 Troubleshooting

The following section is designed to help you isolate the most likely system malfunctions that may occur. For additional help, contact Quest's Technical Support and Service Center.

14.1 Power Up

PROBLEM

- Green Power Light is not blinking every second.

SOLUTION

1) Use a voltmeter and verify that the incoming power to the ESB2 is between 18 and 65VDC.
2) Verify input power polarity. If using an A&B feed, make sure both feeds have correct polarity.
3) Replace the ESB2.

14.2 Communication

PROBLEM

- I experienced a communication failure with the TELSEC® ESB2 through my Ethernet connection.

SOLUTION

1) Verify you have a physical link by looking at the Green LED on the Ethernet jack. Reconnect or replace any defective IP/Ethernet cables.
2) Verify the router the TELSEC® ESB2 is connected to is operable and properly configured.
3) Verify the IP address has not been changed. Connect to the TELSEC® ESB2 with a crossover cable using the programmed address or try the default address of 192.168.1.31
4) Replace the Ethernet cable.
5) Power cycle the ESB2

14.3 Sensor Reading

PROBLEM

- I am not receiving data from one of my sensors.

SOLUTION

1) Verify wiring is correct from the sensor to the TELSEC® ESB2.
2) Use a known good sensor and replace the suspected bad sensor.
3) If the known good sensor doesn't work, then disconnect the field wiring from the TELSEC® ESB2 and connect it directly. If it works now, then correct or replace field wiring.
4) If the known good sensor doesn't read when connected directly to the TELSEC® ESB2 then try the sensor on another input. Be sure to define the input accordingly.
5) If the input is bad, then either move the sensor to a spare input or replace the unit.

14.4 Control Point Failure

PROBLEM

- One of my control points is not turning off.

SOLUTION

1) Verify that the program is correct and has the proper program assigned to the output.
2) Verify the relay on the TELSEC® ESB2 opens and closes when using the bypass command from the web server or via SNMP set to the output.
3) Verify the wiring to the control point is correct.
4) Verify the control source voltage (typically an external 24 VAC transformer) is operable and supplying proper voltage.

14.5 Alarm Notification Failure

PROBLEM

- I am not receiving email alarms.

SOLUTION

1) Verify network connection.
2) Check the setting in the Email Alarm notification page.
3) Verify the EMAIL SMTP server is operational and that the proper username and password (if required) have been entered.
4) Check for any firewalls and or network rule sets preventing emails being sent from the location to the SMTP server.
5) Do a test email by clicking on the Send Test Alarm button.

PROBLEM

- I am not receiving traps.

SOLUTION

1) Verify network connection.
2) Check the settings for the trap server on the TELSEC® ESB2 and make sure they match the requirements of the trap server including SNMP version and community string or username (v3).
3) Check for any firewalls and or rule sets preventing traps being sent from the location to the SMTP server.
4) Send a test trap by clicking on the Send Test Alarm button.

14.6 Battery

PROBLEM

- The system keeps losing history memory and clock settings.

SOLUTION

1) Verify the battery insulating tab was removed at install to enable the battery circuit.
2) Power down the system, remove the cover and change the system battery.

14.7 Modbus Communication and Status

PROBLEM

- Modbus data is not updating, or you notice that Alarm Points referencing Modbus points show COMM ERROR as the status.

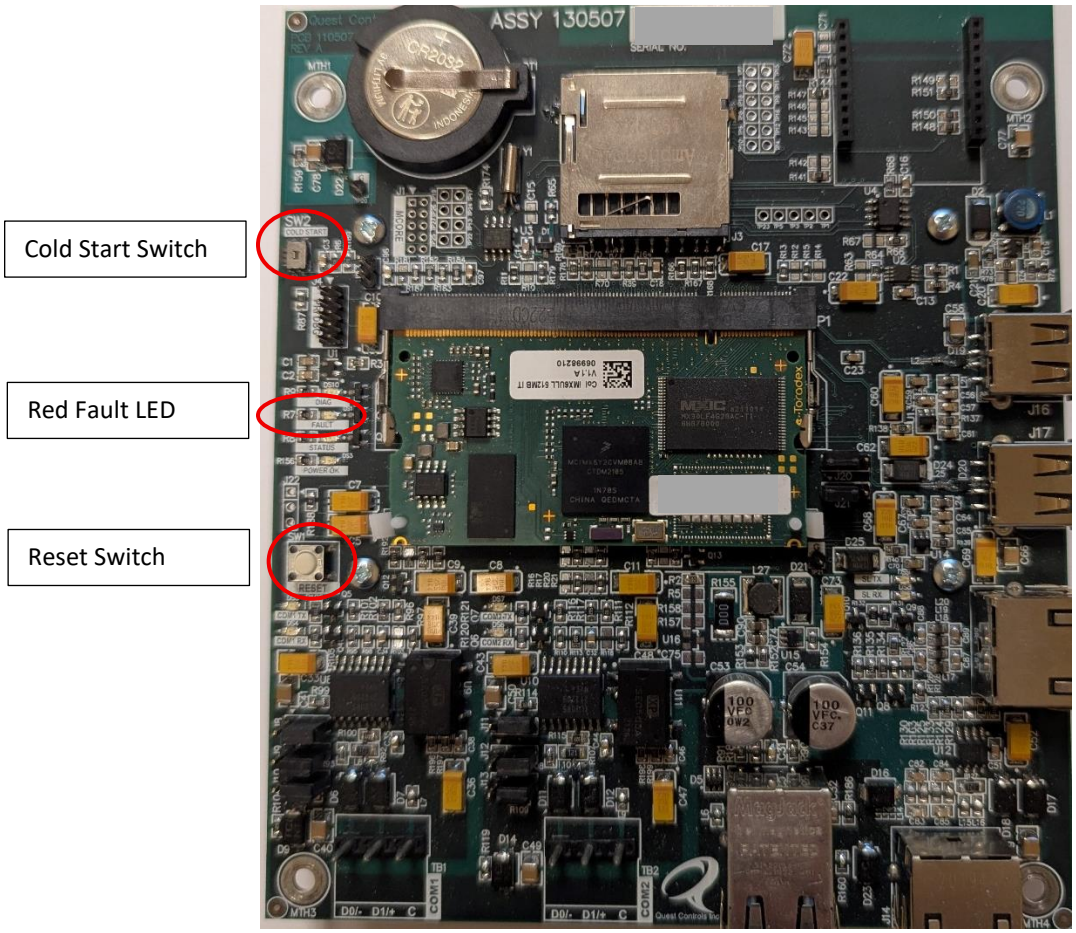
SOLUTION

1) Verify that every Slave device has a unique address and that all devices including the ESB2 are using the same baud rate and parity.
2) Verify that the End of Line (EOL) and line biasing jumpers on the ESB2 are in the proper position.
3) Verify that the ESB2 is communicating over the bus. Remove the cover and look at the Tx and Rx lights. If you see Tx and no Rx activity then the system is attempting to talk to devices, but none of them are responding.
4) Verify the proper wiring between the ESB2 and all slave devices. Isolate the bus to one device first to see if you can get data from that device and then move on to the next device in the chain.

14.8 Return to Factory Default

The following process will erase all memory in the ESB2 and return the system to factory default settings. Prior to doing this process it is advised to have a backup of the current program. Go to Settings/Advanced/Configuration Upload & Download and click on the Download Configuration to browser button. This will download the current program to your Downloads directory on your PC.

1. Locate the Cold Start switch (SW2),the Reset switch (SW1) and the Fault LED on the left edge of the main processor board.
2. Press and hold the Cold Start switch while pressing and releasing the Reset switch. You must hold in the Cold Start switch the entire time during the cold start process.
3. Continue to hold the Cold Start switch until you see the red fault LED come on, flash a few times and then stay on solid. This will take about 20 seconds.
4. Let go of the Cold Start



Appendix A - SNMP Trap Format

The TELSEC® ESB2 can send traps or informs to four different trap servers. The user can program the system to send v1 traps, v2 traps/informs or v3 traps/informs to the server. When the ESB2 sends a trap, it will be sent with an OID based on the severity of the alarm. There will also be multiple variable bindings (varabinds) sent to provide detailed information of the event that occurred. For proper trap reception, you will need the alarm MIB. Please contact Quest for the current revision of the alarm MIB.

TRAP OIDs

The following table shows the available trap OID names and numbers for each trap type:

Alarm Severity	OID Number	OID Name
Critical	11476.100.0.5	alarmCritical
Major	11476.100.0.4	alarmMajor
Minor	11476.100.0.3	alarmMinor
Info	11476.100.0.2	alarmInfo
Clear	11476.100.0.1	alarmClear

When a point goes into alarm, it will generate a trap using the critical, major, minor or info OID. Once the alarm point goes out of alarm or clears, the ESB2 will send a trap with the clear OID.

Trap Variable Bindings

The ESB2 will send the following bindings with every trap that is sent:

Varabind Name	OID Number	Value Type	Value
sysUpTime	1.3.6.1.2.1.1.3	Timeticks	This binding is only available with v2 and v3 traps and will contain the system up time since the ESB2 started up.
snmpTrapOID	1.3.6.1.6.3.1.1.4.1	Object Identifier	This binding is only available with v2 and v3 traps and will contain the OID value for the alarm severity.
notifyProduct	11476.100.1.2	Text String	The product type that is alarming. The value is TELSEC_ESB2
notifyCriticality	11476.100.1.3	Text String	The alarm severity in Text. Available values are: CRITICAL, MAJOR, MINOR, INFO and CLEAR
notifyType	11476.100.1.4	Text String	Alarm type. For TELSEC® ESB2's the type value is SYS for inputs and MOD for Modbus alarms
notifyTime	11476.100.1.5	Timeticks	The time the alarm was generated.
notifyName	11476.100.1.6	Text String	The thirty-two-character name of the alarm point
notifyValue	11476.100.1.7	Text String	The string value of the sensor that caused the alarm.
notifyInputName	11476.100.1.8	Text String	The thirty-two-character name of the input referenced in the alarm point
notifySiteId	11476.100.1.9	Text String	The programmed site identification value for the system.
notifyPointOID	11476.100.1.10	Object Identifier	The OID of the value of the point in alarm

Appendix B - RADIUS Server Configuration

For the RADIUS server configuration, you must use the “Filter-ID” attribute for each user to have access. Inside the “Filter-ID” you must have QuestESB2:<number>

<number> is an encoded value for access to the different sections of the web server. The sections and values are:

Home page – 1
Status – 2
Bypass – 4
Program – 8
Settings – 16
Add up the access level for the user and use that number. Example if you want a user to have full access then the value to enter would be “QuestESB2:31”.

Appendix C – SD Card Update

Overview

The TELSEC® ESB2 can be upgraded using a standard SD memory card. The system supports firmware upgrade as well as loading application programs with an SD card. Use this procedure when you need to change or upgrade the application program of your system. Please contact Quest prior to performing the upgrade if you have any questions or concerns regarding the upgrade files and/or process.

Items needed:

1. SD memory card 2-8 GB in size formatted for use with Windows file format FAT32. (Cards typically come this way)
2. PC or laptop with an SD card reader to allow transfer of files.
3. Firmware upgrade file – Contact Quest for the most current firmware.
4. Application program file (optional)

Process:

1. Obtain the appropriate files from Quest.
2. The program files are in a zip file (compressed) format. Open the file with "File Explorer" and copy the "update" folder with its contents to the SD card.
3. When completed, your SD card should have a file folder with the name "update" on it and in that folder will be all the files necessary to perform the update.
4. If you want to load the application program only using an SD card, then create a folder called "update" and put the application file in this folder. The name of the file must be "allconfig.txt" to work properly.
5. Remove the card from your PC and insert it into the SD slot on the ESB2 processor board. Make sure it clicks into place.
6. The system will take approximately 10-15 seconds to shut down the current processing and then look for the update information.
7. The system will then reboot and rapidly flash the power LED to indicate it is performing the update.
8. The system will reboot again after the update is completed and begin operating with the new parameters. Note this process could be repeated if you are doing both a firmware upgrade and loading a new application program.
9. Remove the SD card by pressing on it to unlatch the spring-loaded connection.

Note the ESB2 will write a file in the update directory that has the name of the MAC address for this system. The file is unique to this system and is to prevent the ESB2 from attempting to load the configuration file again on reboot. The file with the MAC address name can be erased if you need to upgrade this system in the future.

Appendix D – Email Alarm Format

The TELSEC® ESB2 will generate text-based emails for alarm conditions. The email notification will follow a specific format which is outlined below.

Format:

Subject line:

ESB2 <email type> (<severity>) - <user defined subject line>

Email body:

ESB2 named <site name> generated <email type>
Description of <alarm name>
Timestamped at <day month year hh:mm:ss -UTC>
Severity of <severity>
Value of <current value>
Data point name of <data point name>

Where:

<email type> = ALARM or CLEAR
<severity> CRITICAL, MAJOR, MINOR OR WARNING
<user defined subject line> = Subject field entered when defining the Email dispatch.
<site name> = The site name entered under Site Parameters
<alarm name> = The name of the alarm strategy that generated the alarm
<data point current value> = The current value of the data point causing the alarm condition
<data point name> = The name of the data point causing the alarm condition.

Example:

Subject: ESB2 ALARM (Critical) - TELSEC® ESB2 Email notification
ESB2 named ESB2_Site_101 generated ALARM
Description of Tower Lights are not on
Timestamped at 01 Mar 2019 13:30:04 -0700
Severity of Critical
Value of 0.07
Data point name of Light Level Sensor

Appendix E – TL1 Alarms and Commands

TL1 Commands and Alarm:

TL1 stands for "Transaction Language 1." The ESB2 supports TL1 alarm messaging for all alarms being sent out. When an Alarm Point becomes active, the system will generate an alarm. The alarm will be sent using SNMP, SMTP (Email) and/or TL1 depending on how the ESB2 is configured. TL1 alarms assumes the receiving system has established a Telnet session to either the standard IP telnet port or the port number defined in the TL1 Port setup (see section 8.1.4). In addition to sending TL1 autonomous alarm messages, the ESB2 will respond to the following read only commands:

RTRV-HDR:

The RTRV-HDR command is used to retrieve the system identification which is configured in the System Site Configuration (see section 8.1.1). The command format is as follows:

RTRV-HDR:<TID>::<CTAG>;

TID Target identifier (the first ID line). This can be omitted on direct connect.

CTAG Correlation tag. Used to match the response to the command with who sent the command.

<u>Examples:</u>

Command: RTRV-HDR:::K;

Response:
```
  MONITOR 19-10-23 09:57:37
M  K COMPLD
;
```

Command: RTRV-HDR:ESB2_MONITOR::K;

Response:
```
  MONITOR 19-10-23 09:57:37
M  K COMPLD
;
```

RTRV-ALM:

The RTRV-ALM command is used to review active alarms. By issuing the command, the ESB2 will show all currently active alarms. The format for RTRV-ALM command is as follows:

RTRV-ALM-<TYPE>:<TID>::<CTAG>;

Where:

<u>TYPE</u>

ALL Will show active equipment alarms.
EQPT Will show active equipment alarms.
ENV Shows active environmental alarms.

TID Target identifier (first line of the ID). This can be left out for direct connection.
CTAG Correlation tag to match the request with response.

Examples:

Command: RTRV-ALM-ENV:::X123;

Response:

```
  ESB2_MONITOR 19-10-23 09:59:48
M  X123 COMPLD
*  1 REPT ALM ENV
  "ALM_A002:MN,BTL_LTMP,10-04,15-46-10:\"BLD002 LOW TEMPERATURE ALARM,ON\""
*  2 REPT ALM ENV
  "ALM_A004:MN,BHL_LHUM,10-04,15-46-10:\"BLD011 LOW HUMIDITY ALARM,ON\""
;
```

Automatic TL1 Alarms

Once a point is determined to be in alarm, the controller will automatically send the alarm and log the event in the alarm log. When the alarm goes away, the controller will automatically send out a clear message in TL1 format. The clear will also be logged in the alarm log.

TL1 ALARM MESSAGE

FORMAT:

<cr><lf><lf>

<sp><sp><sp>TID<sp>YY-MM-DD<sp>HH:MM:SS<cr><lf>

ALMCDE<sp><ALARM SEQUENCE NUMBER><sp>REPT<sp>ALM<sp>ENV <cr><lf>

<sp><sp><sp>”ALM_<REF#>:<NOTIFICATION CODE>, <ALARM NAME>,MM-DD,HH:MM:SS,<ALARMMESSAGE>”<cr><lf>

;

<u><ALMCDE></u>

*C = CRITICAL ALARM

** = MAJOR ALARM

*<sp> = MINOR ALARM

A<sp> = AUTOMATIC used for clear alarm and heartbeat messages

<ALARM SEQUENCE NUMBER> = sequential sequence number assigned to every TL1 autonomous message.

<REF#>= The four-digit alarm reference number that was assigned when defining the alarm.

<NOTIFICATION CODE>
CR = Critical alarm
MJ = Major alarm
MN = Minor alarm
CL = Cleared alarm
NA = Non Alarm used for notifications.

<ALARM NAME> = The 32-character name that was defined for the alarm.

< ALARMMESSAGE> = The 32-character name of the point causing the alarm followed by a comma and the current status of the alarm.

Sample alarm:

```
  SITE401 19-10-26 11:43:39
** 1 REPT ALM ENV
  "ALM_A005:MJ,POWER_Failure,10-26,11:43:39,COMMERCIAL POWER FAILURE"
;
```

Sample Clear Alarm:

```
  SITE401 19-10-26 11:43:59
A  2 REPT ALM ENV
  " ALM_A005:CL,POWER_Failure,10-26,11:43:59,COMMERCIAL POWER FAILURE"
;
```

Sample Heartbeat message:

```
  RDNGPAREH72_ESB2 19-10-23 09:48:05
A  13808 REPT EVT ENV
  "HEARTBEAT:NA,HEARTBEAT,10-23,09-48-05:\"User defined Heartbeat message\""
;
```

Appendix F – Network Settings and Supported Protocols

Physical Interface:

Ethernet 100Base-T

Switch port settings:

Recommend Auto Negotiate. The system will typically operate at 100Mb full duplex.

Additional port settings:

Disable spanning tree protocol (or set it to port fast) for the port the product is connected to.

IP protocols supported:

Protocol	Port #	User Changeable
IPv4		Enable/Disable
IPv6		Enable/Disable
DHCP	67/68	No
DNS	58	No
HTTP	80	Yes
HTTPS	443	Yes
NTP	123	No
RADIUS	1812	Yes
SMTP	25	Yes
SNMP	161	Yes
SNMP Traps/Informs	162	No
SSH	22	Yes
Telnet	23	Yes

Contact Information

Website:

www.questcontrols.com

Email:

General information: info@questcontrols.com

Technical support: CustomerService@questcontrols.com

Mailing Address:

Quest Controls, Inc.

208 9th Street Dr. West, Palmetto FL 34221

Key Phone Numbers

Main Number: (941) 729-4799

Sales/Sales support: option 1.

Technical & Applications Support: option 3.

Repair and RMA Support: option 5.

Fax: (941) 729-5480