3.4 XML Operation

Custom computer applications may be created to monitor and control the X-300™ without using a web browser. Monitoring the state of the device is done by sending an HTTP GET request for the file state.xml. This file returns the current status of the device and is different depending on the current mode of the device. The device can also be controlled by passing parameters along with the request for the file state.xml similarly to how an HTML form would send name/value pairs to a server when the submit button is pressed. This can be demonstrated by entering commands into the URL line of a web browser.

To simply request the state of the device, enter: http://192.168.1.2/state.xml

To set the set temperature of the device, enter: http://192.168.1.2/state.xml?setTemp=75.0

To set the set temperature of the device and turn the fan on, enter:

http://192.168.1.2/state.xml?setTemp=75.0&fanMode=0

To turn the first relay on, enter: http://192.168.1.2/state.xml?relay1State=1

In response to each request, an XML file will be returned with the current state of the device. The format of the XML file will differ depending on the mode of the X-300TM (Thermostat or Temperature Monitor). Also, some parameters only work in *Thermostat* mode while others only work in *Temperature Monitor* mode.

3.4.1 Thermostat Mode

Entering the above address in *Thermostat* mode will return the following XML page:

```
<datavalues>
        <units>F</units>
        <indoorTemp>x.x</indoorTemp>
        <outdoorTemp>x.x</outdoorTemp>
        <setTemp>65</setTemp>
        <heat>0</heat>
        <cool>0</cool>
        <fan>0</fan>
        <hold>00</hold>
        <minTemp>00:00 300.0</minTemp>
        <maxTemp>00:00 -300.0</maxTemp>
        <minTempY>00:00 0.0</minTempY>
        <maxTempY>00:00 0.0</maxTempY>
        <heatMode>0</heatMode>
        <fanMode>1</fanMode>
        <filtChng>60</filtChng>
        <maxSTemp>85.0</maxSTemp>
        <minSTemp>65.0</minSTemp>
        <mode>6</mode>
        <time>1235406753</time>
        <serialNumber>00:0C:00:00:00:00</serialNumber>
</datavalues>
```

The tags <units>, <indoorTemp>, <wallTemp>, <outdoorTemp>, <setTemp>, <heat>, <cool>, <fan>, <hold>, <minTemp>, <maxTemp>, <heatMode>, <fanMode>, <filtChng>, <maxSTemp>, <minSTemp>, <mode>, and <time> indicate the current state of the X-300™ unit. Values for each are described below.

<units>	F (Degrees Fahrenheit)
	C (Degrees Celsius)

<indoorTemp>	x.x (indicates that no digital temperature sensor is associated with the indoor temperature sensor) 80.1 (temperature reading from digital temperature sensor associated with the indoor temperature sensor)
<outdoorTemp>	x.x (indicates that no digital temperature sensor is associated with the outdoor temperature sensor) 80.1 (temperature reading from digital temperature sensor associated with the outdoor temperature sensor)
<setTemp>	65 (indicates the set temperature)
<heat>	Indicates whether the heat relay is energized (0 = off,1 = on).
<cool>	Indicates whether cool relay is energized (0 = off,1 = on)
<fan>	Indicates whether fan relay is energized (0 = off,1 = on)
<hold>	Indicates whether the *Hold* or 7-day program is enabled. The first number shows whether hold is enabled and the second number indicates whether the 7-day programming is enabled. (00 = both hold and 7-day program are disabled, 01 = hold is disabled; 7-day program is enabled, 11 = both hold and 7-day program are enabled, 10 = hold is enabled, 7-day program is disabled)
<minTemp>	12:48 80.1 (indicates minimum 24 hour temperature reading [time temp])
<maxTemp>	12:48 80.1 (indicates maximum 24 hour temperature reading [time temp])
<minTempY>	80.1 (indicates minimum temperature reading for yesterday [time temp])
<maxTempY>	80.1 (indicates maximum temperature reading for yesterday [time temp])
<heatMode>	Indicates current ***HEAT MODE*** (0 = off, 1 = heat only, 2 = cool only, 3 = auto)
<fanMode>	Indicates current ***FAN MODE*** (0 =on, 1 =auto)
<filtChng>	60 (indicates number of days until filter should be changed)
<maxSTemp>	85 (indicates maximum allowable set temperature)
<minSTemp>	65 (displays minimum allowable set temperature)
<mode>	Indicates current ***Mode*** (6 = *Thermostat* mode,7 = *Temperature Monitor* mode)
<time>	1235406753 = time displayed in “epoch time” (number of seconds since January 1, 1970)
<serialNumber>	00:0C:00:00:00:00 Serial Number (MAC address) of X-300

In *Thermostat* mode there are five parameters that can be sent to the X-300™ to control it. The current set temperature, heat mode, fan mode, hold state, and filter change counter can all be set/configured through the use of HTTP GET requests for the file state.xml.

The supported parameters are:

1. setTemp
   Set the set temperature. Valid range must be withing the min and max set temp values.

   Example: `http://192.168.1.2/state.xml?setTemp=75.0`

2. heatMode
   Set the heat mode of the X-300™. Possible values are:
   0 – off
   1 – heat
   2 – cool
   3 – auto
   Example: http://192.168.1.2/state.xml?heatMode=1

3. hold
   Hold the current set temperature. Essentially toggles the 7 day programming on/off. Possible values are:
   1 - toggle hold state
   Example: http://192.168.1.2/state.xml?hold=1

4. fanMode
   Change the fan mode. Possible values are:
   0 - Fan always on
   1 - Fan works automatically
   Example: http://192.168.1.2/state.xml?fanMode=1

5. rstFilt
   Reset the filter counter to the filter change interval in the setup pages. Possible values are:
   1 – reset
   Example: http://192.168.1.2/state.xml?rstFilt=1
   Note: This parameter will only reset the filter counter once the counter has reached zero.

These parameters can be sent individually or chained together in one request. The previous parameters, for example, could all be sent in one request as follows:

http://192.168.1.2/state.xml?setTemp=75&heatMode=1&fanMode=1&hold=1

Note that when commands are sent to the X-300™, its current sate is returned in the form of an XML page. The command can also be sent without having the X-300™ return the XML page. This is accomplished by adding the noReply parameter as follows:

http://192168.1.2/state.xml?setTemp=75&noReply=1

Password :
If the control password is enabled in the temperature module and the XML page is requested through a browser, the browser will prompt the user for the password. If the XML request is sent from another application and not a browser, the HTML request will need to contain the password encoded using the base 64 encoding scheme. The HTML request header without the password looks like this:

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

The HTML request header with the password looks like this:

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

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

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

3.4.2 *Temperature Monitor* Mode

To request the current state enter:

http://192.168.1.2/state.xml

In *Temperature Monitor* mode the following XML page will be returned:

```
-<datavalues>
    <units>F</units>
    <sensor1temp>x.x</sensor1temp>
    <sensor2temp>x.x</sensor2temp>
    <sensor3temp>x.x</sensor3temp>
    <sensor4temp>x.x</sensor4temp>
    <sensor5temp>x.x</sensor5temp>
    <sensor6temp>x.x</sensor6temp>
    <sensor7temp>x.x</sensor7temp>
    <sensor8temp>x.x</sensor8temp>
    <relay1state>0</relay1state>
    <relay2state>0</relay2state>
    <relay3state>0</relay3state>
    <mode>7</mode>
</datavalues>
```

The tags <units> , <sensorXtemp>, <relayXstate>, and <mode> indicate the current state of the X-300™ unit. Values for each tag are described below.

<units>	F (Degrees Fahrenheit) C (Degrees Celsius)
<sensorXtemp>	x.x (indicates that no digital temperature sensor is associated with the sensor number) 80.1 (temperature reading from digital temperature sensor associated with the sensor number)
<relayXstate>	Indicates whether relay is energized (0 = off, 1 = on)
<mode>	Indicates current ***Mode*** (6 = *Thermostat* mode, 7 = *Temperature Monitor* mode)

In *Temperature Monitor* mode there are six parameters that can be sent to the device to control it. Using these parameters, the relay states can be changed.

The supported parameters are:

1. relay1State
2. relay2State
3. relay3State

    Change the state of relays 1 – 3. Possible values are:
    1 – On
    0 – Off
    2 – Pulse
    5 – Toggle

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

4. pulseTime1
5. pulseTime2
6. pulseTime3

Used to change the pulse time for a pulse command. Used in conjunction with a relayXState=2 command.
Valid range: 0.1 to 86400 seconds or 0.1 seconds to 1 day.

Example: http://192.168.1.2/state.xml?relay1State=2&pulseTime1=5

Note that the PulseTimeX command does not change the pulse time specified in the setup pages, nor is it stored or recorded.

Note that when commands are sent to the X-300™, its current sate is returned in the form of an XML page. The command can also be sent without having the X-300™ return the XML page. This is accomplished by adding the noReply parameter as follows:

http://192168.1.2/state.xml?setTemp=75&noReply=1

Password:
If the control password is enabled in the temperature module and the XML page is requested through a browser, the browser will prompt the user for the password. If the XML request is sent from another application and not a browser, the HTML request will need to contain the password encoded using the base 64 encoding scheme. The HTML request header without the password looks like this:

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

The HTML request header with the password looks like this:

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

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

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

3.4.3 XML Diagnostics

There is a special XML page that can be requested for diagnostic purposes called diagnostics.xml. A request for this file returns the following:

```
<datavalues>
    <internalTemp>76.6</internalTemp>
    <vin>15.0</vin>
    <fiveVolt>5.0</fiveVolt>
    <memoryPowerUpFlag>1</memoryPowerUpFlag>
    <devicePowerUpFlag>1</devicePowerUpFlag>
    <powerLossCounter>1</powerLossCounter>
</datavalues>
```

The tags <internalTemp>, <vin>, <fiveVolt>, <memoryPowerUpFlag>, <devicePowerUpFlag>, and <powerLossCounter> are helpful parameters when trying to diagnose a problem with the X-300™.

<internalTemp> Indicates the internal temperature of the device

<vin> Indicates the DC voltage that is applied to the Vin+ and Vin- terminals.

<fiveVolt>	Indicates the 5V power supply.
<memoryPowerUpFlag>	Indicates a loss of power to the capacitor backed real time clock. This field can be set to 0. A value of 1 indicates that the real time clock lost power. This should only happen if the device has lost power for several days. There is an internal capacitor that can power the real time clock for an extended period of time even if the main power is lost. If the real time clock looses power, the time will have to be reset. By default this will read 1 until it is set to 0.
<devicePowerUpFlag>	Indicates a loss of power to the device. This field can also be set to 0. A value of 1 means the X-300™ has lost power at least one time since the flag was set to 0.
<powerLossCounter> count	A count of how many times the device has lost power. This field is a of how many times the X-300™ has lost main power.

In both *Thermostat* mode and *Temperature Monitor* mode the devicePowerUpFlag and memoryPowerUpFlag can be cleared by requesting the file diagnotics.xml and passing parameters along with this request. For example, to clear both flags the following request can be sent to the device:

http://192.168.1.2/diagnostics.xml?memoryPowerUpFlag=0&devicePowerUpFlag=0

The powerLossCounter can also be cleared by using the request:

http://192.168.1.2/diagnostics.xml?powerLossCounter=0

Requests made for the diagnostics.xml file do not require a password.

3.5 Modbus Operation

The X-300™ can be controlled (and read) using Modbus/TCP protocol. This provides a standard means of using the temperature module in conjunction with devices and software from other manufacturers. This section contains the information necessary to communicate with the X-300™ using Modbus/TCP. This is not a tutorial on Modbus and it is assumed that the reader is already familiar with Modbus. Detailed Modbus information can be found at http://www.modbus.org.

It is important to note that when the control password in the X-300™ is enabled, Modbus/TCP communications are disabled. This is because Modbus/TCP does not provide a mechanism for password protection. Make sure the control password is disabled (default) before using Modbus with the X-300™.

The X-300™ functions as a Modbus server (slave). Client (master) devices open a connection with the temperature module on port 502 (unless another Modbus port is selected) and send commands or requests to read the state of the relays, read the current value of the temperature sensors, or change the relay states. When the X-300™ receives a command, it will perform the desired function and return a response. The following commands are available:

Read Coils (Modbus function 01 (0x01)) - read the status of the relays.
Read Holding Registers (Modbus Function Code 03 (0x03)) – read temperature values of sensors
Write Single Coil (Modbus function 05 (0x05)) - change one relay state
Write Multiple Coils (Modbus Function Code 15 (0x0F)) - change one or more relay states
Write Multiple Registers (Modbus function 16 (0x10)) - pulse the relay(s)

Multiple commands may be sent without closing and re-opening the connection, but if no data is

transferred for 50 seconds the connection will time out. To keep the connection open, a read request can be sent periodically. Two TCP sockets are available for Modbus communications. If two masters each have a socket open (or a single master using two sockets) and a third master attempts to open a socket for Modbus communications, the request to open a third socket will be rejected.

When errors occur, an error code is sent. This code is comprised of the function code sent plus 0x80. For example, an error during the read coils function 0x01 would return 0x81. Each error has a qualifying exception number. The following are the possible exception codes and their meanings.

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

3.5.1 Read Coils (Modbus Function Code 01 (0x01))

This function returns the state of the relay coils. 0 denotes that the coil is not energized, 1 denotes that the coil is energized.

Valid Starting Address and Quantity of Coil Combinations

The following gives the starting address range of coils and the number of possible coils to read. Some examples are also shown of using the given addressing and quantities.

Starting Address: 0x00 to 0x02
Quantity: 0x01 to 0x03

Examples:

Starting address 0x00 quantity of coils 0x01 (read relay 1 state only)
Starting address 0x01 quantity of coils 0x01 (read relay 2 state only)
Starting address 0x02 quantity of coils 0x01 (read relay 3 state only)
Starting address 0x00 quantity of coils 0x02 (read relay 1 and 2 state)
Starting address 0x01 quantity of coils 0x02 (read relay 2 and 3 state)
Starting address 0x00 quantity of coils 0x03 (read relay 1, 2, and 3 state)

The response of the unit will be given in a binary value which determines the state of the relays. The following chart gives all possible responses from the unit.

Data Byte (hex)	LS Data Bits (binary)	Relay 3 State	Relay 2 State	Relay 1 State
0x00	000	off	off	off
0x01	001	off	off	on
0x02	010	off	on	off
0x03	011	off	on	on
0x04	100	on	off	off
0x05	101	on	off	on
0x06	110	on	on	off
0x07	111	on	on	on

3.5.2 Read Holding Registers (Modbus Function Code 03 (0x03))

This function is used to read the temperature values of the sensors. One, multiple, or all values can be read at the same time using this function.

Valid Starting Addresses and Quantity of Coil Combinations
Reading these registers requires that the address be in the range of 16 to 30 (0x10 and 0x1E). Also, the address must be evenly divisible by 2. In other words, only address 0x10, 0x12, 0x14, 0x16, 0x18, 0x1A, 0x1C, and 0x1E are valid. Each value is returned as two registers in IEEE 754 floating point format. The four data bytes are treated as two individual big endian 16-bit words with the least significant word being sent first. In other words, the 32 bit floating point number represented as '1234 ABCD' is sent as 'CDAB 3412'. For example, if we were to read two registers starting at address 0x10, and sensor 1 measured the temperature to be 81.25 degrees, then we would get the following value 800042A2. The least significant word would be 8000 hex and the most significant word would be 42A2. Using a conversion utility we could then convert this hexadecimal value into the temperature reading of 81.25 degrees. The following table lists the addresses that correspond to each input.

The device returns a value of 0xffffffff (NaN) if there isn't a valid reading or when the sensor isn't active.

Starting Address: 0x10 to 0x1E
Quantity: 0x02 to 0x10 (must be divisible by 2)

The following chart gives the addresses of each of the temperature sensors.

Sensor	Address
1	0x10
2	0x12
3	0x14
4	0x16
5	0x18
6	0x1A
7	0x1C
8	0x1E

3.5.3 Write Single Coil (Modbus Function Code 05 (0x05))

This command allows to change the state of a single relay coil. The valid addresses and quantities correspond to to the relay coils described in section 3.5.1.

Starting Address: 0x00 to 0x02
Quantity: 0x01 to 0x03

3.5.4 Write Multiple Coils (Modbus Function Code 15 (0x0F))

This function allows you to control relays 1, 2, or 3 simultaneously using one command.

Valid starting Address and Quantity of Coil Combinations
Valid starting address can fall in the range of 0x00 and 0x02. The only restriction is that the sum of the starting address and the quantity of coils must not be greater than 0x03. For example, if you want to control two relays, then the starting address can be be 0x01 or 0x00. On the other hand, you may want to control the third relay only. This can be achieved by starting at address 0x02 and using a quantity of 0x01.

Starting Address: 0x00 to 0x02
Quantity: 0x01 to 0x03

3.5.5 Write Multiple Registers (Modbus Function Code 16 (0x10))

This is used to pulse the relays for a specified time. When the X-300™ receives this command, it immediately turns the relay coil on (if it is not on already on) and starts the pulse timer. The pulse time is specified in the register value field and the time can range from 0.1 seconds to 86400 seconds (1 day). When the pulse time expires, the relay coil will be turned off. If a pulse time command is sent with a value greater than 86400, the pulse timer will be set to 86400. If a pulse time command is sent with a value less than 0.1, the pulse timer will be set to 0.1. If any commands are sent to the X-300™ (commands may be Modbus, XML, HTML) before the pulse timer has expired, the pulse timer will be canceled immediately and the new command will be executed.

Valid Starting Addresses and Quantity of Coil Combinations

Writing to these register requires that the address be in the range of 0x16 and 0x1A. Also, the address must be evenly divisible by 2. In other words, only address 0x16, 0x18 and 0x1A are valid addresses. When the address 0x16 is used, all three relays can be sent pulse times (12 bytes), or just the first relay can be sent a pulse time (4 bytes.) If the address 0x1A is used, only relay 3 will be sent a pulse time (4 bytes.)

Note that the pulse time is provided in IEEE 754 floating point format. The four data bytes are treated as two individual big endian 16-bit words but the least significant word is sent first. In other words, the 32-byte floating point number represented as 'ABCD 1234' is sent as 'CDAB 3412'. In the example shown below, the relay will pulse for 10 seconds. Ten seconds is represented using a floating point number of 41200000. It is transferred as 00004120.

Starting Address: 0x10 to 0x12
Quantity: 0x01 to 0x03

3.6 Special Functions

3.6.1 Email Alerts

Each of the eight sensors on the X-300™ can be configured to send an email message to up to five email addresses when alarm conditions occur. The header, descriptions and temperature units are taken from the entries in the setup pages.

The message is shown below with default text settings. The text shown in bold indicates that it can be changed by the user.

X-300 - Email Notification

Email alert triggered by high alarm from **Sensor 1**: 87.0° **F**

Sensor 1 Current Temp: 87.4° **F**
Sensor 2 Current Temp: xx.x
Sensor 3 Current Temp: xx.x
Sensor 4 Current Temp: xx.x
Sensor 5 Current Temp: xx.x
Sensor 6 Current Temp: xx.x
Sensor 7 Current Temp: xx.x
Sensor 8 Current Temp: xx.x

Quite a bit of the text above is shown in bold. These items can be changed by the user using the fields described below from the setup pages.

X-300 – This text is set in the ***Main Header Text*** field under the **Control Page Setup** tab.

F – The units are set through the radio button field ***Units*** under the **Main** tab.
Sensor 1-8 – This text is set in the ***Sensor Description*** field for the selected sensor under the **Sensors** tab.

*Note: The email message will only contain the temperature for sensors that have the *Display Temperature/State*' option selected under the **Control Page Setup** tab.

Email notification requires that the following fields are set. See Section 2 for a description of each field.

Network Tab:
IP Addressing
Netmask
Broadcast
Gateway
DNS Addresses (Only necessary if SMTP server is not listed as an IP address)
Mail Server (SMTP)
Port
Domain
Return Email
Email 1 Address (Email 2,3,4,5 Address are optional)

Sensor Tab:		**Thermostat Setup** Tab
Email Options	-or-	Notification Email
Use Email Address		

**Testing Email: Once the email settings are entered, the email functionality can be tested by entering the following command.
http://192.168.1.2/state.xml?testEmail=1

This will cause an email message to be sent immediately. If the message doesn't get through, the system log file may have some clues to assist in diagnosing the problem (http://192.168.1.2/syslog.txt).

3.6.2 Alarm Conditions

Diagram 3.4 below shows a visualization a high alarm set point and a low alarm set point when one of the temperature sensors is configured to send out emails or remote messages based on the set points. When the temperature goes above the high alarm set point a high alarm occurs. This high alarm condition remains until the temperature drops below the deadband. The same is true when the temperature falls below the low alarm set point. In this situation the low alarm condition remains until the temperature rises past the deadband. Red dots indicate high and low alarms while the green dots indicate a return to the normal operating condition. Note that relays are only changed when the alarm condition changes. Users can still manually change relays using the web page.

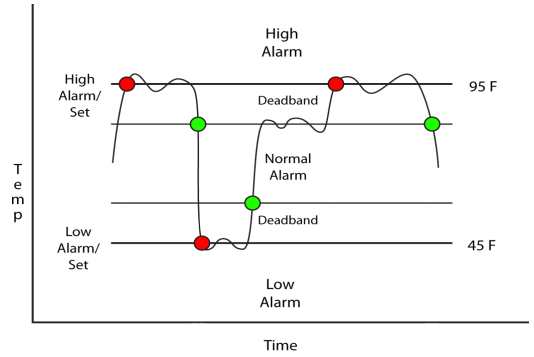

Figure 3.4 – High, Low, and Normal Alarm Diagram