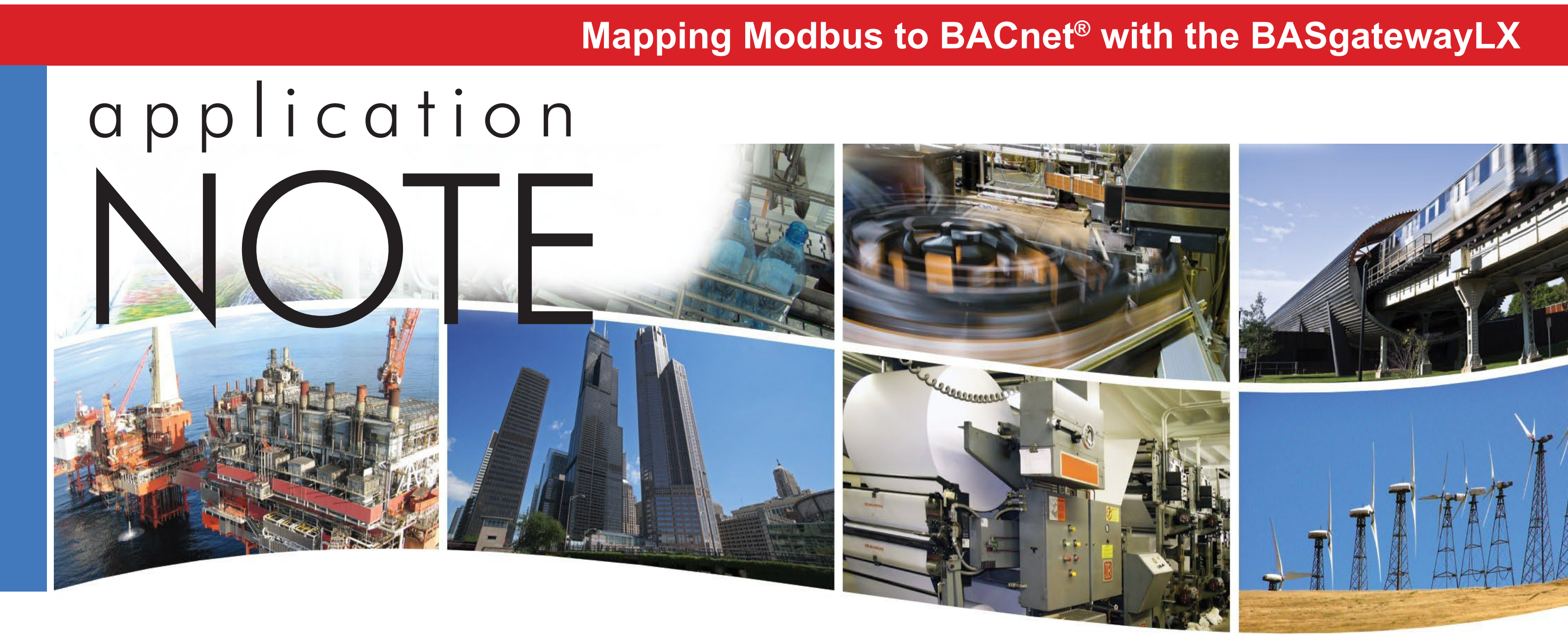
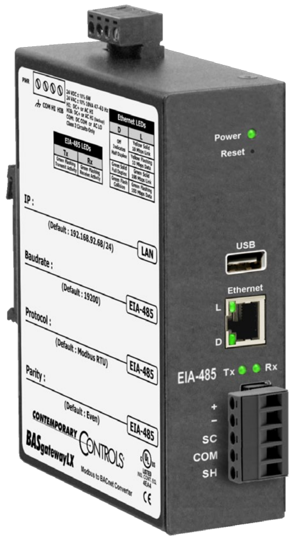


Mapping of Modbus Registers to BACnet® Objects Using the BASgatewayLX

Modbus remains a popular protocol and is frequently encountered as the default protocol used on mechanical equipment. It is simple to use and understand but it is not BACnet compliant. In order to make a Modbus device BACnet compliant, a Modbus to BACnet gateway is required such as Contemporary Controls' BASgatewayLX. This application note explains how the BASgatewayLX can be used to bring a Modbus device up to BACnet/IP. What is required is the development of a device profile for the particular Modbus device being accessed. Once the device profile is developed it is installed into the BASgatewayLX and a scan list is created by selecting those points on the device profile of interest. Only those enabled Modbus points on the scan list will be accessible from a BACnet client. Modbus device profiles are developed by Contemporary Controls as a convenience to its customers. However with care, the customer can modify the device profile to fit the unique requirements of an installation or develop a completely new device profile. This application note can assist on how this is done.

An important feature of the BASgatewayLX is that it supports virtual routing meaning that each connected Modbus device is treated by BACnet as a BACnet device with a unique device instance. This makes adding multiple Modbus devices with identical device profiles to the same BASgatewayLX easy. In addition, multiple Modbus devices with different device profiles are just as easily supported. The only restriction is that each BASgatewayLX can support up to 30 Modbus devices and a total of 1000 Modbus scanned points.

Modbus devices can be in the form of Modbus ASCII, Modbus RTU and Modbus TCP. Modbus ASCII is an older serial protocol which is seldom used. Modbus RTU is a very popular serial line interface while Modbus TCP operates over Ethernet. All forms of Modbus are supported by the BASgatewayLX.



Each BASgatewayLX can support up to 30 Modbus devices and a total of 1000 Modbus scanned points.

People have asked "what advantage does BACnet devices have over Modbus devices?" The simple answer is that BACnet devices allow their objects to be "discovered" while Modbus devices have no such capability. This feature speeds up commissioning. What this means is that BACnet devices are modelled as a collection of objects that are "network visible." By being able to view these BACnet objects and by understanding their predefined properties, much can be learned about the device. With Modbus, you will need to consult a user's manual to understand the meaning of the Modbus registers. However, there is a way of creating BACnet objects from Modbus registers using the BASgatewayLX. Because Modbus equipment is popular, being able to integrate these devices into a "single-seat BACnet system" is important. In order to understand how Modbus device profiles are created, a short review of Modbus is necessary.

Modbus Device Addressing

Modbus slaves are addressed from 1-247 while address 0 is reserved as the broadcast address – a message directed to all devices. The BASgatewayLX functions as the Modbus master and therefore has no address assignment. It has reserved Modbus slave address 247 for future use. Up to 30 Modbus serial devices can connect to the EIA-485 port on the BASgatewayLX with each one assigned a unique Modbus address in the allowable range. For Modbus TCP devices, each must attach to a port on an Ethernet switch located on the same subnet as the BASgatewayLX. As mentioned before, the BASgatewayLX supports virtual routing meaning that every attached Modbus slave (serial or TCP) is treated as a unique BACnet device. Therefore at a minimum, a BACnet device object must be created for each connected Modbus device. Other BACnet objects must be created to represent Modbus data which is in the form of 1-bit and 16-bit registers.

Modbus Data Addressing

Modbus data is considered to be segregated into four memory blocks—coils, discrete inputs, input registers and holding registers. Each memory location in each block can be accessed by a 16-bit address. Discrete inputs and coils are considered to be 1-bit registers while input registers and holding registers are 16-bit. A 32-bit register would require two memory locations. Traditional addressing practice uses 5-digit decimal references with the leading digit signifying a particular memory block as shown in Table 1. This approach is simpler to understand but it only addresses 10,000 points or registers instead of the 65,535 which are possible for each block. As the complexity of Modbus devices increased, the need to use more memory locations increased. Modern Modbus addressing incorporates 6-digit references with the first digit signifying the memory block address (0, 1, 3, and 4) and the remaining 5 digits representing the complete 16 bits of address space for that block. Contemporary Controls utilizes the 6-digit addressing scheme as shown in Table 2. Notice that the first register location within a block begins at 1. This is called PLC addressing which the original Modicon PLC used. Actually, the data is stored in computer memory beginning with 0 but Contemporary Controls uses PLC addressing references to avoid confusion.

Memory Block	Bits	Access	Address Range
Coils	1	Read/Write	00001-09999
Discrete inputs	1	Read-Only	10001-19999
Input registers	16	Read-Only	30001-39999
Holding registers	16	Read/Write	40001-49999

Table 1 — Traditional decimal representation of Modbus registers using a 5-digit addressing scheme.

Memory Block	Bits	Access	Address Range
Coils	1	Read/Write	000001-065535
Discrete inputs	1	Read-Only	100001-165535
Input registers	16	Read-Only	300001-365535
Holding registers	16	Read/Write	400001-465535

Table 2 — Modern decimal representation of Modbus registers using a 6-digit addressing scheme. This is the method used by Contemporary Controls.

Modbus Function Codes

In order to act upon the data within the Modbus memory blocks, the Modbus protocol defines a set of function codes but not all function codes are supported by all Modbus devices. The BASgatewayLX supports the following function codes shown in Table 3 which are adequate for accessing common Modbus devices. If a Modbus slave device does not support multiple register reads, the BASgatewayLX will communicate with the device using single register reads.

Function Code	Description
1	Read coils
2	Read discrete inputs
3	Read holding registers
4	Read input registers
5	Write single coil (available but not used)
6	Write single register (available but not used)
15	Write multiple coils
16	Write multiple registers

Table 3 — The BASgatewayLX supports the most common Modbus function codes

Assigning Modbus Registers to BACnet Objects

Looking at the four types of Modbus memory blocks and having an understanding of available BACnet objects it is obvious that Modbus discrete inputs should be assigned to BACnet binary inputs (BI) and Modbus input registers should be assigned to BACnet analog inputs (AI). The present value of both types of objects can only be read. Modbus coils should be assigned to BACnet binary outputs (BO). Modbus coils as well as BACnet binary outputs could be read or written. If a Modbus coil is to be locked-out from being written via the Modbus interface, it is best to treat this Modbus point as a BACnet BI and not as a BO. Holding registers could be inputs or outputs so assigning them as analog outputs (AO) make sense because they can be read or written. However, if it is obvious that holding register is to be treated only as an input, it should be assigned to a BACnet AI instead of an AO.

When assigning Modbus registers or points to a BACnet object, an understanding of the significance of the variable needs to be gained. Contemporary Controls does its best in interpreting the Modbus register tables provided by the equipment supplier. If a different interpretation is needed, it is possible to modify the Modbus device profile accordingly.

BACnet Object Properties

From the above discussion, only four BACnet object types have been selected (besides the required device object). The next step is to determine the required properties for these four objects. The abbreviated conformance code table (Table 4) shows what properties are required and what are optional.

BACnet Property	BI	BO	AI	AO
Object Identifier	R	R	R	R
Object Name	R	R	R	R
Object Type	R	R	R	R
Present Value	R	W	R	W
Status Flags	R	R	R	R
Event State	R	R	R	R
Out of Service	R	R	R	R
Polarity	R	R	NA	NA
Units	NA	NA	R	R
Priority Array	NA	R	NA	R
Relinquish Default	NA	R	NA	R
COV Increment	NA	NA	O	O

Table 4 — Abbreviated Conformance Code table

O = optional
R = required to be read
W = required being read and written
NA = not applicable to this object

Within one BACnet device, there can be several object types such as BI, BO; AI; etc. However, the Object Identifier for any of these types within the device must be unique. Likewise, the Object Names within the device must be unique. The Object Type will be set accordingly to represent the Modbus register as BI, BO, AI or AO. The Present Value represents the actual value of the point being read or written to by the BASgatewayLX. For BIs and BOs, the BACnet variable BACnetBinaryPV is used. For AIs and AOs, the BACnet variable type REAL is supported which follows IEEE floating point convention. Status Flags, Event State and Out of Service need not be configured. They are handled by the BASgatewayLX. Polarity is a required property for binary points. It was decided to preset all polarities to NORMAL so as to not invert the state of a Modbus coil or discrete input. Units of Measure for analog points need to be identified but it is possible to indicate “no units”. An attempt will be made to assign a BACnet unit to an equivalent Modbus unit otherwise “no units” will be inserted.

For output points the Priority Array and Relinquish Default must be set. Priority Array is set when the Present Value is written. The Present Value assumes the value of the Relinquish Default when there is a null Priority Array. In other words, the BASgatewayLX follows the BACnet rules for prioritizing the writing of outputs. A Modbus output will only be changed if a command is received with a higher priority from what is already written for a particular output.

The BASgatewayLX supports BACnet Change of Value (COV), so the COV increment must be set on analog points that will be subscribed. For analog points, a COV value of 0 will result in notification of any change in value of the point. It is not necessary to set an increment on a binary point. The BASgatewayLX can support up to 100 binary COV subscriptions and 100 analog COV subscriptions.

The first issue is to create a unique Object Instance for each Modbus register in the device profile. It was decided to simply assign sequential numbers beginning with 1. This is fine for 1-bit registers and single 16-bit word registers but if the data is represented as a double-word (32-bits) then the first register is assigned a BACnet object instance and the second register is skipped. Sequential numbering will then continue with the register following the skipped location. If the user decides to modify an object instance, care should be exercised to ensure that object instance uniqueness is maintained.

There is a special case called ‘bit-picking” where a single 16-bit word register represents from 1 to 16 discrete points. This register must then be mapped to individual BI and BO points. Therefore instead of having a single object instance for one Modbus 16-bit register, up to 16 sequential object instances are created depending upon the number of discrete points.

The next issue is the Object Name which must be unique. Any unique name would work but it is recommended to use the register description for the corresponding Modbus register. For example, if register 400001 means the “Voltage phase A to neutral,” that will be the name. However, this exact name cannot be used anywhere within the device profile. If two or more identical device profiles are loaded into the same BASgatewayLX, this is not an issue because each device profile corresponds to different devices. Therefore the uniqueness constraint is still met. However, to guard against the user modifying an object name that is already used within a device, the BASgatewayLX will save the change with an appended number in order to maintain uniqueness.

Creating Device Objects

Every BACnet device (virtual or real) must have one and only one device object. The BASgatewayLX is a real BACnet device and therefore must be represented as a device object. Each attached Modbus device is considered a virtual device and therefore must also be represented by a device object. The properties for a device object are much different than for a BI, BO, AI or AO and most of the required properties are preconfigured by Contemporary Controls and not accessible by the user. However, there are two properties that must be assigned for each Modbus device.

The Device Object Identifier must be unique BACnet internetwork-wide therefore the convention used was to take Contemporary Controls’ BACnet vendor ID (245), append a “0” and then append the 3-digit Modbus slave address. For example, if the Modbus slave address is 99, then the Device Object Identifier would be 2450099. This does not guarantee uniqueness BACnet internetwork-wide so it must be checked against all other devices in the network.

The Device Object Name must also be unique within the BACnet internetwork. The user can assign a name which has significance to the Modbus device being connected. The BASgatewayLX provides the default name “Modbus device name” and to ensure uniqueness appends the Device Object Identifier as shown above. For example, the default Device Object Name for the Device Object Identifier above would be “Default Modbus Name 2450099”.

Developing a Modbus Device Profile

The development of a Modbus Device Profile begins with the manufacturer’s product information in the form of a Modbus Register Table. Notice that in the following example that the registers in Table 5 are all holding registers with beginning location 40001. Although these are 5-digit references this is not a problem. Modbus uses offset bas+1 addressing but the actual data is stored in a base+0 address as shown. Although read/write holding registers are indicated by a 4xxxx reference, most of the registers are actually inputs so they will be mapped as AIs in BACnet. Notice register 40016. This register must be expanded because it really represents a collection of status bits that will be mapped to BIs in BACnet. You can see the detailed data in a second Modbus Register Table called Table 6.

Example Modbus Register Table for an Instrument

Register Address	Modbus Address	Data Type	Scaling	Comment
0x00	40001	Flow in Eng unit (low)	No	Mass flow in selected unit
0x01	40002	Flow in Eng unit (high)	No	
0x02	40003	Total (low)	No	Total in selected unit
0x03	40004	Total (High)	No	
0x04	40005	Temperature (low)	*10	Temperature in selected unit * 10
0x05	40006	Temperature (high)	*10	
0x06	40007	Elapsed time (low)	*10	Elapsed time in hours * 10
0x07	40008	Elapsed time (high)	*10	
0x08	40009	Velocity (Low)	No	Velocity in nm/hr
0x09	40010	Velocity (high)	No	
0x0A	40011	Flow in Eng unit * 10	10	Mass flow in selected unit * 10
0x0B	40012	Flow in Eng unit *100	100	Mass flow in selected init * 100
0x0C	40013	Total *100	100	Total in selected unit * 100
0x0D	40014	Total2 (low, 2 gas curves only)	No	Total #2 for 2 gas curves
0x0E	40015	Total2 (high, 2 gas curves only)	No	Total #2 for 2 gas curves
0x0F	40016	Status	No	Status
0x10	40017	Spare/ Not used		
0x11	40018	Spare/ Not used		
0x12	40019	Spare/ Not used		
0x13	40020	Flow in Eng Unit (float, upper 16 bits)	No	Mass flow in selected unit
0x14	40021	Flow in Eng Unit (float , lower 16 bits)	No	Mass flow in selected unit
0x15	40022	Total in Eng Unit (float, upper 16 bits)	No	Total in selected unit
0x16	40023	Total in Eng Unit (float, lower 16 bits)	No	Total in selected unit
0x17	40024	Total#2 for 2 gas curve (float, upper 16 bits)	No	Total in selected unit
0x18	40025	Total#2 for 2 gas curve (float, lower 16 bits)	No	Total in selected unit
0x19	40026	Temperature in selected unit (float, upper 16 bits)	No	Temperature in selected unit
0x1A	40027	Temperature in selected unit (float, lower 16 bits)	No	Temperature in selected unit
0x1B	40028	Elapsed time in hours (float, upper 16 bits)	No	Elapsed time in hours
0x1C	40029	Elapsed time in hours (float, lower 16 bits)	No	Elapsed time in hours
0x1D	40030	Velocity in selected unit (float, upper 16 bits)	No	Velocity in selected unit
0x1E	40031	Velocity in selected unit (float, lower 16 bits)	No	Velocity in selected unit
0x1F	40032	Spare/ Not used		
0x20	40033	Spare/ Not used		
0x21	40034	Spare/ Not used		
0x22	40035	Spare/ Not used		
0x23	40036	Spare/ Not used		

Table 5 — Modbus register table from an equipment vendor.

What follows is an expansion of register 40016. The complete register must be represented as 16 binary inputs.

Bit	Definition	Comment
0	Power up indication	Reset when out of the power up sequence
1	Flow rate reached high limit threshold	Set limit to zero to disable
2	Flow rate reached low limit threshold	Set limit to zero to disable
3	Temperature reached high limit threshold	Set limit to zero to disable
4	Temperature reached low limit threshold	Set limit to zero to disable
5	Sensor reading is out of range	Check sensor wiring
6	Velocity flow rate outside of calibration table	Check sensor wiring
7	Incorrect Settings	Check settings
8	In simulation mode	Set simulation value to 0 to disable
9	Frequency output is out of range	Check frequency output settings
10	Analog 4-20 mA for flow is out of range	Check analog output settings
11	Analog 4-20 mA for temperature is out of range	Check analog output settings
12	Anybus error	Check wiring from RS485 to Anybus IC
13	RTC error (only for FT2 with RTC)	Check RTC
14	CRC error	Check parameters and reset CRC
15	Tot Error	Reset total

Table 6 — Details of register 40016 as a collection of status bits.

Once all the Modbus points are determined and there is an understanding on how they are to be mapped to BACnet objects, the Modbus Device Profile can be generated using an Excel spreadsheet. With care the user can develop or modify an existing Modbus Device Profile with an understanding of its structure.

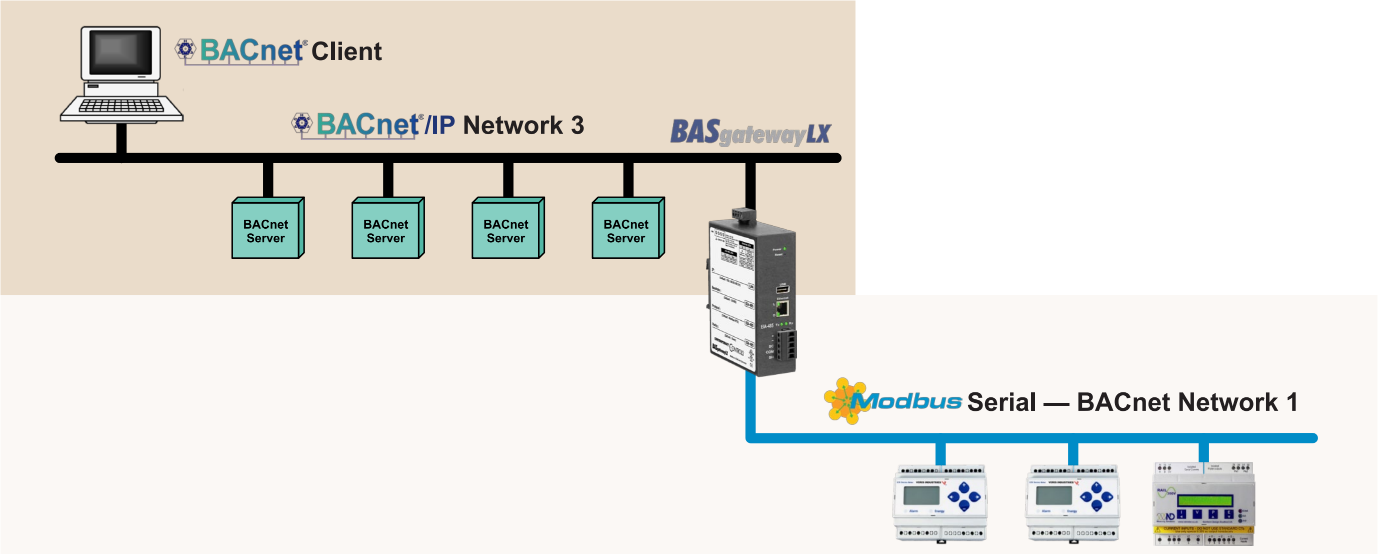
The Modbus Device Profile is actually a Comma Delimited Variable (CSV) file that is supported in Excel. It can also be viewed with a simple text editor such as Microsoft's Notepad. Each record ends with a line feed (LF). There is no start-of-record character. With this type of file, the user must be careful not to corrupt the file by inserting invalid characters. What follows is a Modbus Device Profile for the above device as viewed in an Excel spreadsheet. In addition, there is information to help the user in understanding the data structure of a Modbus Device Profile if modifications are intended.

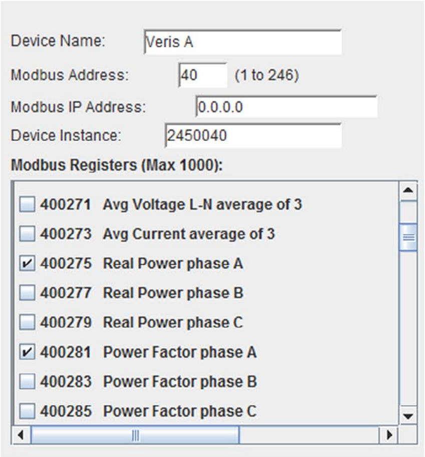
Virtual Routing

The BASgatewayLX incorporates a feature called virtual routing which allows each attached Modbus device to appear as a BACnet device. In the figure below, we have one BACnet client communicating to BACnet/IP server devices and to a BASgatewayLX. Each of these devices, including the BASgatewayLX, will have a unique device instance. The BASgatewayLX complies with the BACnet device profile B-ASC which means that it is an application specific controller functioning as a server.

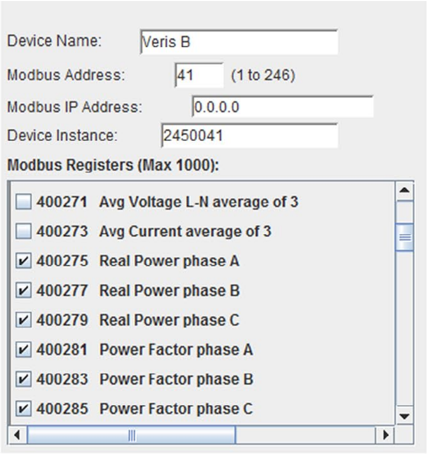
Attached to the EIA-485 port on the BASgatewayLX are three Modbus RTU energy meters. Two are identical Veris meters and the other is a Northern Design meter. Because of virtual routing, each of these meters will appear as BACnet devices with unique device instances. They will each inherit the same characteristics as the BASgatewayLX including the B-ASC device profile. Notice that these three devices are assigned to BACnet network 1 because they are considered to be residing on a separate network which in this case is a virtual network. The BASgatewayLX resides on network 3.

Shown at the bottom are three device profiles representing the three Modbus devices. The first two device profiles are the same because they represent the same model meter. However, not all points have been enabled for scanning. The remaining device profile is for a completely different meter. Notice that the device names and device instances are different. They must be unique BACnet internetwork-wide. The Modbus slave addresses are shown. Because these devices are not Modbus TCP devices, there will be no Modbus IP address assignment.

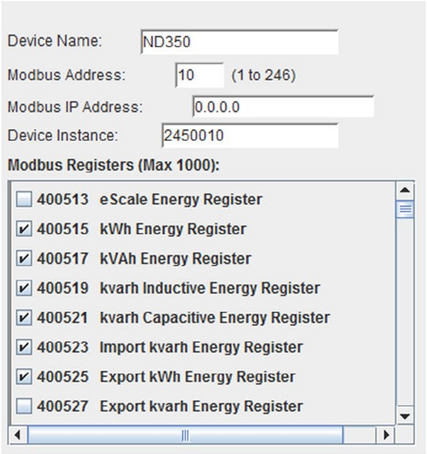



Device Name: Veris A
Modbus Address: 40 (1 to 246)
Modbus IP Address: 0.0.0.0
Device Instance: 2450040
Modbus Registers (Max 1000):

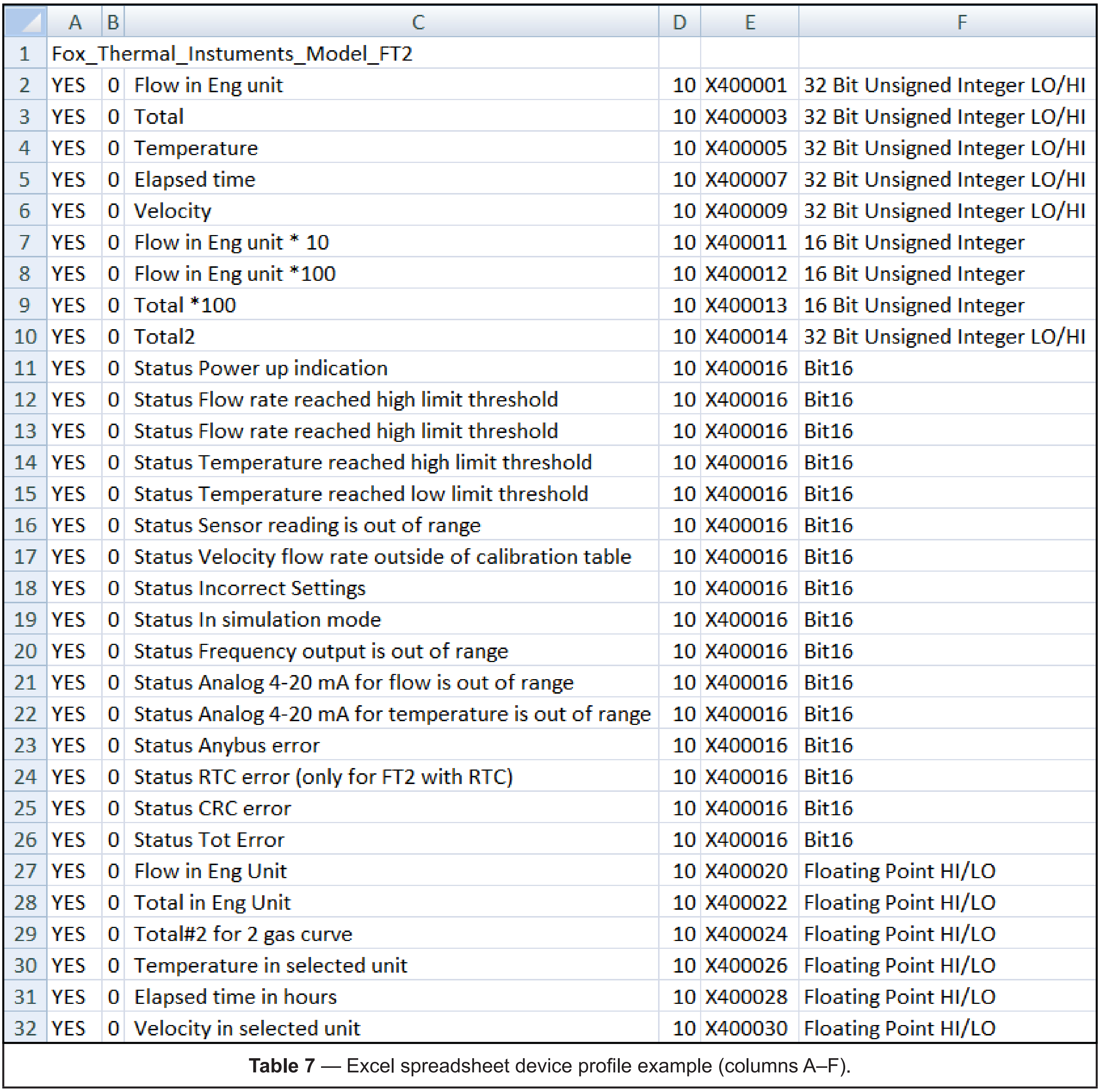
☐ 400271 Avg Voltage L-N average of 3
☐ 400273 Avg Current average of 3
☑ 400275 Real Power phase A
☐ 400277 Real Power phase B
☐ 400279 Real Power phase C
☑ 400281 Power Factor phase A
☐ 400283 Power Factor phase B
☐ 400285 Power Factor phase C


Device Name: Veris B
Modbus Address: 41 (1 to 246)
Modbus IP Address: 0.0.0.0
Device Instance: 2450041
Modbus Registers (Max 1000):

☐ 400271 Avg Voltage L-N average of 3
☐ 400273 Avg Current average of 3
☑ 400275 Real Power phase A
☑ 400277 Real Power phase B
☑ 400279 Real Power phase C
☑ 400281 Power Factor phase A
☑ 400283 Power Factor phase B
☑ 400285 Power Factor phase C

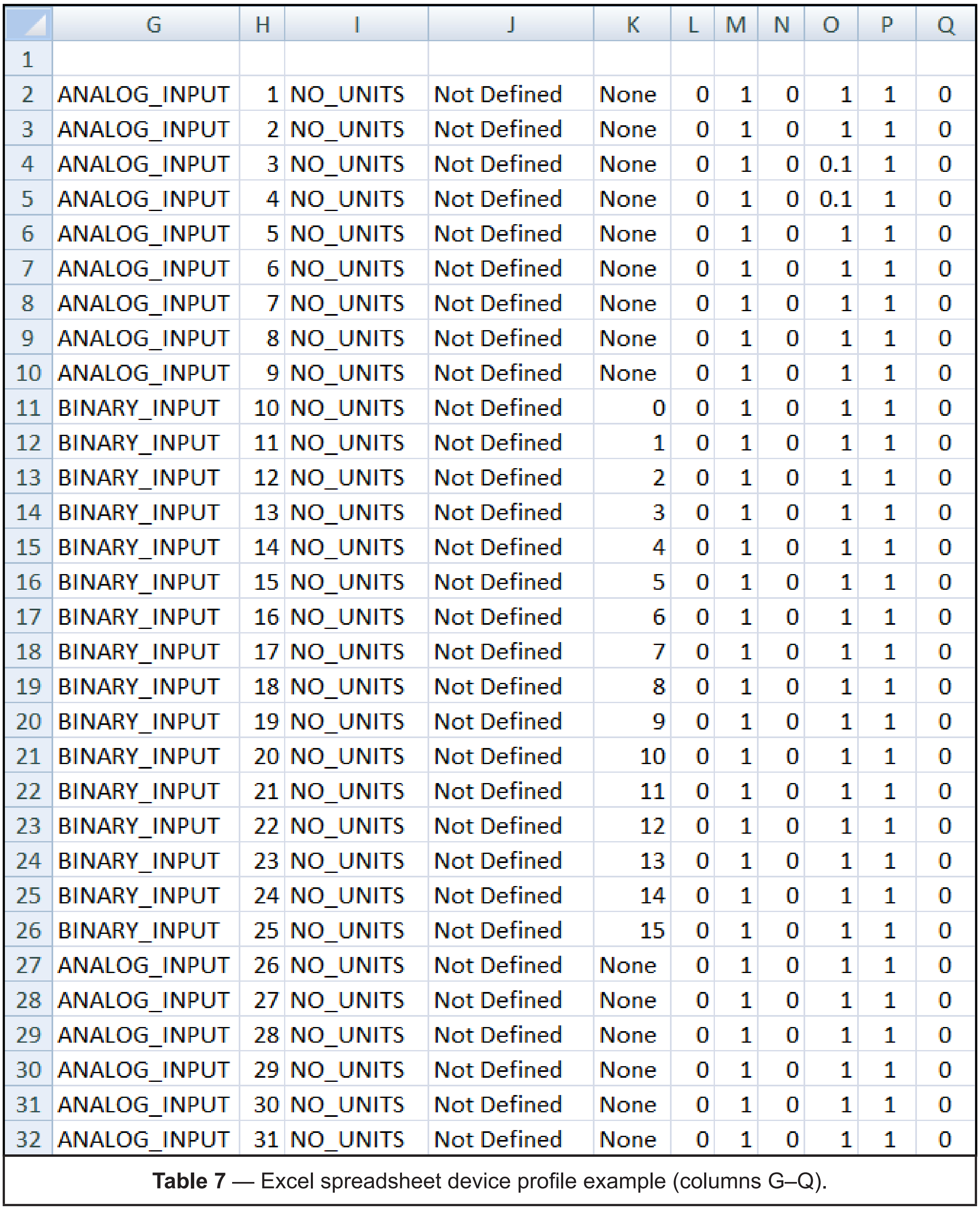

Device Name: ND350
Modbus Address: 10 (1 to 246)
Modbus IP Address: 0.0.0.0
Device Instance: 2450010
Modbus Registers (Max 1000):

☐ 400513 eScale Energy Register
☑ 400515 kWh Energy Register
☑ 400517 kVAh Energy Register
☑ 400519 kvarh Inductive Energy Register
☑ 400521 kvarh Capacitive Energy Register
☑ 400523 Import kvarh Energy Register
☑ 400525 Export kWh Energy Register
☐ 400527 Export kvarh Energy Register



	A	B	C	D	E	F
1	Fox_Thermal_Instuments_Model_FT2					
2	YES	0	Flow in Eng unit	10	X400001	32 Bit Unsigned Integer LO/HI
3	YES	0	Total	10	X400003	32 Bit Unsigned Integer LO/HI
4	YES	0	Temperature	10	X400005	32 Bit Unsigned Integer LO/HI
5	YES	0	Elapsed time	10	X400007	32 Bit Unsigned Integer LO/HI
6	YES	0	Velocity	10	X400009	32 Bit Unsigned Integer LO/HI
7	YES	0	Flow in Eng unit * 10	10	X400011	16 Bit Unsigned Integer
8	YES	0	Flow in Eng unit *100	10	X400012	16 Bit Unsigned Integer
9	YES	0	Total *100	10	X400013	16 Bit Unsigned Integer
10	YES	0	Total2	10	X400014	32 Bit Unsigned Integer LO/HI
11	YES	0	Status Power up indication	10	X400016	Bit16
12	YES	0	Status Flow rate reached high limit threshold	10	X400016	Bit16
13	YES	0	Status Flow rate reached high limit threshold	10	X400016	Bit16
14	YES	0	Status Temperature reached high limit threshold	10	X400016	Bit16
15	YES	0	Status Temperature reached low limit threshold	10	X400016	Bit16
16	YES	0	Status Sensor reading is out of range	10	X400016	Bit16
17	YES	0	Status Velocity flow rate outside of calibration table	10	X400016	Bit16
18	YES	0	Status Incorrect Settings	10	X400016	Bit16
19	YES	0	Status In simulation mode	10	X400016	Bit16
20	YES	0	Status Frequency output is out of range	10	X400016	Bit16
21	YES	0	Status Analog 4-20 mA for flow is out of range	10	X400016	Bit16
22	YES	0	Status Analog 4-20 mA for temperature is out of range	10	X400016	Bit16
23	YES	0	Status Anybus error	10	X400016	Bit16
24	YES	0	Status RTC error (only for FT2 with RTC)	10	X400016	Bit16
25	YES	0	Status CRC error	10	X400016	Bit16
26	YES	0	Status Tot Error	10	X400016	Bit16
27	YES	0	Flow in Eng Unit	10	X400020	Floating Point HI/LO
28	YES	0	Total in Eng Unit	10	X400022	Floating Point HI/LO
29	YES	0	Total#2 for 2 gas curve	10	X400024	Floating Point HI/LO
30	YES	0	Temperature in selected unit	10	X400026	Floating Point HI/LO
31	YES	0	Elapsed time in hours	10	X400028	Floating Point HI/LO
32	YES	0	Velocity in selected unit	10	X400030	Floating Point HI/LO

Table 7 — Excel spreadsheet device profile example (columns A–F).



	G	H	I	J	K	L	M	N	O	P	Q
1											
2	ANALOG_INPUT	1	NO_UNITS	Not Defined	None	0	1	0	1	1	0
3	ANALOG_INPUT	2	NO_UNITS	Not Defined	None	0	1	0	1	1	0
4	ANALOG_INPUT	3	NO_UNITS	Not Defined	None	0	1	0	0.1	1	0
5	ANALOG_INPUT	4	NO_UNITS	Not Defined	None	0	1	0	0.1	1	0
6	ANALOG_INPUT	5	NO_UNITS	Not Defined	None	0	1	0	1	1	0
7	ANALOG_INPUT	6	NO_UNITS	Not Defined	None	0	1	0	1	1	0
8	ANALOG_INPUT	7	NO_UNITS	Not Defined	None	0	1	0	1	1	0
9	ANALOG_INPUT	8	NO_UNITS	Not Defined	None	0	1	0	1	1	0
10	ANALOG_INPUT	9	NO_UNITS	Not Defined	None	0	1	0	1	1	0
11	BINARY_INPUT	10	NO_UNITS	Not Defined	0	0	1	0	1	1	0
12	BINARY_INPUT	11	NO_UNITS	Not Defined	1	0	1	0	1	1	0
13	BINARY_INPUT	12	NO_UNITS	Not Defined	2	0	1	0	1	1	0
14	BINARY_INPUT	13	NO_UNITS	Not Defined	3	0	1	0	1	1	0
15	BINARY_INPUT	14	NO_UNITS	Not Defined	4	0	1	0	1	1	0
16	BINARY_INPUT	15	NO_UNITS	Not Defined	5	0	1	0	1	1	0
17	BINARY_INPUT	16	NO_UNITS	Not Defined	6	0	1	0	1	1	0
18	BINARY_INPUT	17	NO_UNITS	Not Defined	7	0	1	0	1	1	0
19	BINARY_INPUT	18	NO_UNITS	Not Defined	8	0	1	0	1	1	0
20	BINARY_INPUT	19	NO_UNITS	Not Defined	9	0	1	0	1	1	0
21	BINARY_INPUT	20	NO_UNITS	Not Defined	10	0	1	0	1	1	0
22	BINARY_INPUT	21	NO_UNITS	Not Defined	11	0	1	0	1	1	0
23	BINARY_INPUT	22	NO_UNITS	Not Defined	12	0	1	0	1	1	0
24	BINARY_INPUT	23	NO_UNITS	Not Defined	13	0	1	0	1	1	0
25	BINARY_INPUT	24	NO_UNITS	Not Defined	14	0	1	0	1	1	0
26	BINARY_INPUT	25	NO_UNITS	Not Defined	15	0	1	0	1	1	0
27	ANALOG_INPUT	26	NO_UNITS	Not Defined	None	0	1	0	1	1	0
28	ANALOG_INPUT	27	NO_UNITS	Not Defined	None	0	1	0	1	1	0
29	ANALOG_INPUT	28	NO_UNITS	Not Defined	None	0	1	0	1	1	0
30	ANALOG_INPUT	29	NO_UNITS	Not Defined	None	0	1	0	1	1	0
31	ANALOG_INPUT	30	NO_UNITS	Not Defined	None	0	1	0	1	1	0
32	ANALOG_INPUT	31	NO_UNITS	Not Defined	None	0	1	0	1	1	0

Table 7 — Excel spreadsheet device profile example (columns G–Q).

Understanding the Data Structure of a Modbus Device Profile as Viewed in a Spreadsheet

Cell A1 contains the file name that will appear in the BASgatewayLX drop list box. It is a maximum of 40 characters and consists of alphanumeric character and the _ (underscore) character. No spaces are allowed and the maximum field size is 40 characters. All other cells in row 1 must be blank.

Example: CCC_Example

The remainder of the fields in row A should be left blank. They are reserved for future use.

Rows 2 and Beyond contain information in columns A through Q. In each of the data fields there can be no spaces or non-printable characters. In addition, the following characters are not allowed: & : ‘ “,

Column A represents the polling flag and contains the word YES or NO which are default conditions. The YES will be interpreted by the BASgatewayLX as a checked box to poll for the variable, while NO is an unchecked box indicating polling will not occur. After the file is loaded into the BASgatewayLX the boxes can be altered by the user by clicking on the box to switch from poll-mode to not-polled mode.

Column B contains the COV value (change of value). For a binary (on/off) variable it is always a 0 because a change of state is always transmitted if the host has established a COV connection. For an analog variable the value can be 0 (every change will be transmitted) or some other value like 10, 20.56; etc. which represents the COV increment.

Column C contains the BACnet object name. It must be unique and contains 1 to 64 characters maximum with no commas.

Column D is a place holder and contains no required data. This cell MUST have a value. Leave the value 10 in this cell.

Column E contains the data type and data table elements. The address is preceded by an uppercase X followed by the Modbus data type (0, 1, 3, 4), followed by the element number in the data table 1 through 65535. Data table addressing begins with 1 (called PLC addressing). However, data will actually be stored in locations that are one less than the PLC address.

Column F contains the data format.

- Bit — is for references 0aaaaa and 1aaaaa and for the bit picking of a 3aaaaa or 4aaaaa references. As mentioned before, a 6–digit addressing scheme is used with “aaaaa” representing an address from 1 to 65535.
- 16 Bit Unsigned Integer — 16 bits read as 0 to 65,535.
- 16 Bit Signed Integer — 16 bits read as –32,768 to 36,767.
- 32 Bit Unsigned Integer LO/HI — 32 bits read as 0 to 4,294,967,295, with the least significant word at the base address.
- 32 Bit Signed Integer LO/HI — 32 bits read as –2,147,483,648 to 2,147,483,647, with the least significant word at the base address.
- 32 Bit Unsigned Integer HI/LO — 32 bits read as 0 to 4,294,967,295, with the least significant word at the base address + 1.
- 32 Bit Signed Integer HI/LO — 32 bits read as –2,147,483,648 to 2,147,483,647, with the least significant word at the base address + 1.
- Floating Point LO/HI — 32 bits read as –3.4028235E+38 to 3.4028235E+38 with the least significant data at the base address.
- Floating Point HI/LO — 32 bits read as –3.4028235E+38 to 3.4028235E+38 with the least significant data at the base address +1.
- Bit16 — 16 bits read as 0 or 1 per bit, Bit number is specified in column K
- Bit32 LO/HI — 32 bits read as 0 or 1 per bit, Bit number is specified in column K with the least significant data at the base address.
- Bit32 HI/LO — 32 bits read as 0 or 1 per bit, Bit number is specified in column K with the least significant data at the base address +1.

Column G contains the BACnet object type.

- BINARY_INPUT — for use with all 1aaaaa, 0aaaaa that will be read-only, bit-pick of all 3aaaaa, and bit-pick of 4aaaaa that are read-only.
- BINARY_OUTPUT — for use with 0aaaaa that are read-write and bit-pick of 4aaaaa that are read/write.
- ANALOG_INPUT — for all 3aaaaa and 4aaaaa that are read-only.
- ANALOG_OUTPUT — for all 4aaaaa that are read/write.

Column H contains the BACnet object instance number. This sequential number is generated by the spreadsheet program beginning with 1. Each object instance is unique regardless of the object type (AI, AO, BI and BO).

Column I contains the BACnet valid unit of measure. A table of currently supported BACnet units can be found at:

www.ccontrols.com/support/basgatewaylx.htm

Enter NO_UNITS in this column if the unit of measure is unknown, not needed or unsupported by BACnet.

Column J contains the BACnet Object description. It can contain 64 characters. This is an optional field in the BACnet object and is not needed but do not leave the field blank. The field must contain some string.

Column K contains the bit pick selection if required. Typically the value is None. If a bit-pick will occur a value from 0 to 15 (16 Bit), or 0 to 31 (32 Bit), is specified with bit 0 being the least significant bit.

Columns L, M, N and O are for scaling of the incoming Modbus value. L and M represent two points (x_1, x_2) on the Modbus input scale while N and O represent two equivalent points (y_1, y_2) of the scaled result which will be presented to BACnet. A linear relationship is assumed between the input values and the output values. For example; the range of values in the Modbus register is 0 to 4800 representing a voltage, but we want the BACnet device to read 0 to 480.0. Set the value in Column L to 0, Column M to 1, Column N to 0 and Column O to .1 to scale the value. Any two points along the Modbus scale can be used while entering the equivalent BACnet values.

Assume the Modbus input reflects temperature in degrees Fahrenheit and we want to convert to degrees Centigrade (Celsius). Set L to 32, M to 212, N to 0 and O to 100. It does not matter if the full range of values entered for the Modbus scale will actually be encountered. The mathematical relationship is created by the four values entered. If no scaling is desired, set L, M, N, and O to 0, 1, 0 and 1 respectively.

Column P contains a 1 or 0. A 1 allows group reads of Modbus variables assuming the variables are contiguous. Entering a 0 does not allow grouping.

Column Q contains a 1 or 0 and only pertains to Modbus output registers. The value of 1 allows the last value written to a Modbus register by BACnet to be retransmitted when a Modbus device returns from an offline to online condition. For example, if the Modbus device was power cycled and returned to service the BASgatewayLX will resume reading Modbus registers in the device. However, register outputs may not be updated to the same state when the Modbus device first went off. If continuity of the output state of a particular output register is important after re-connection, set this bit to a 1. If the default state of the Modbus register is preferred when re-connecting the Modbus device to the BASgatewayLX leave this value at 0. There will be no attempt to resend the last value of the selected Modbus register after a re-connection. The default value for this column is 0.

BACnet Units Supported

The BASgatewayLX supports all of the BACnet Engineering Units as described in ASHRAE 135-2010 clause 21.

Editing Modbus Device Profiles in a Spreadsheet

Editing of Modbus Device Profiles can be accomplished with any spreadsheet program as long as the program can read a file with a ***.csv*** extension and can save a file with the same ***.csv*** extension. If column widths are changed for better reading and it is desirable to retain these changes for later edits, it is best to save the work in the native spreadsheet format. However, for loading into the BASgatewayLX, the file format MUST be comma separated variable and therefore must be saved as such.

For simple edits, a text editor such as Notepad can be used. shows the same Modbus Device Profile as viewed with a Notepad text editor.

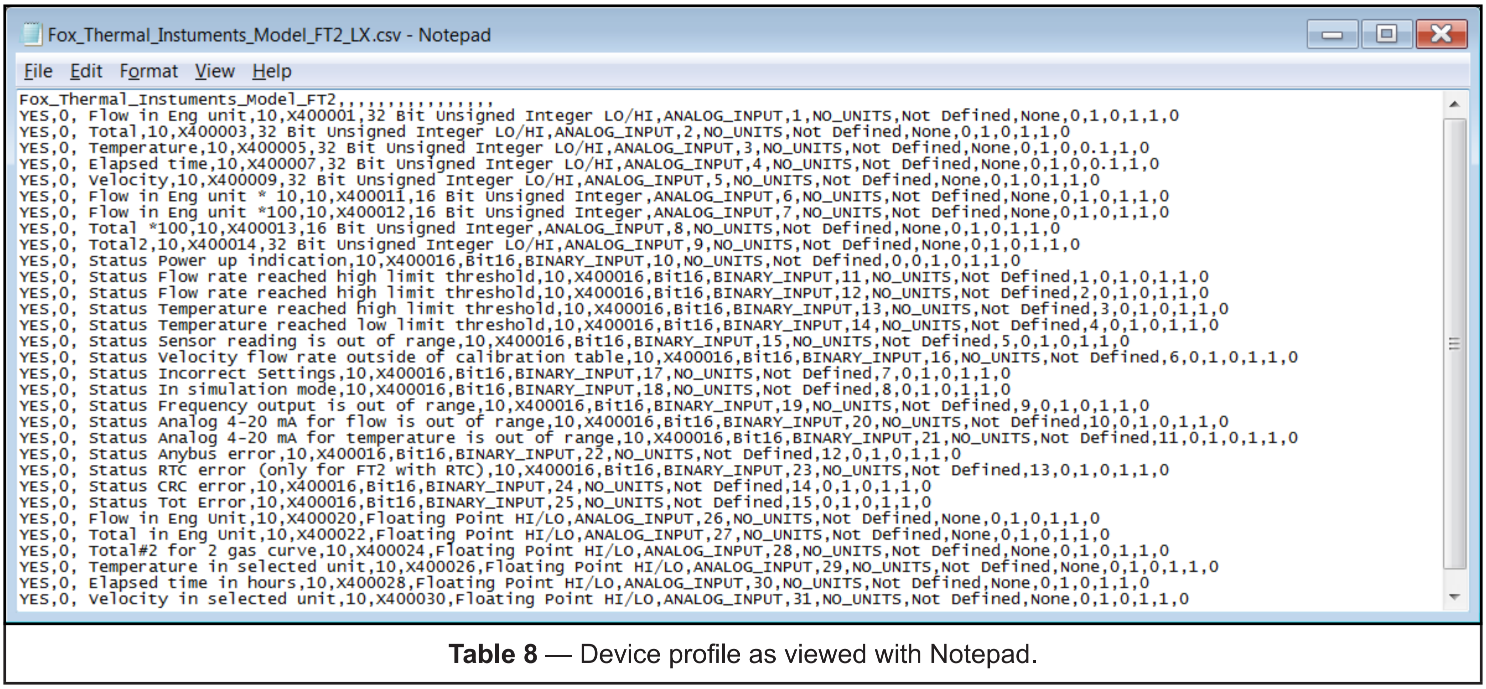
Using the BASgatewayLX

The BASgatewayLX web pages are accessible from any web browser with Java version 6.0 or above installed. Access is password protected and instructions on installing the unit can be found in the installation guide. Once access is achieved, resident help screens are available to assist the user. These help screens are reprinted below to assist the reader in understanding how to load and use device profiles. Although the product is shipped with some device profiles, Contemporary Controls maintains a library of device profiles at:

www.ccontrols.com/support/basgatewaylx.htm

Device profiles can be downloaded to the user's PC and then uploaded to the BASgatewayLX. Once device profiles are installed, they can be configured for scanning by simply clicking boxes adjacent to the desired registers. If more significant changes need to be made to the device profile, it can be modified on the user's PC and then uploaded with a different file name.

What follows is information on how to use the BASgatewayLX by examining the resident screens on the unit. However, the data on the screens will differ from that of the target device.

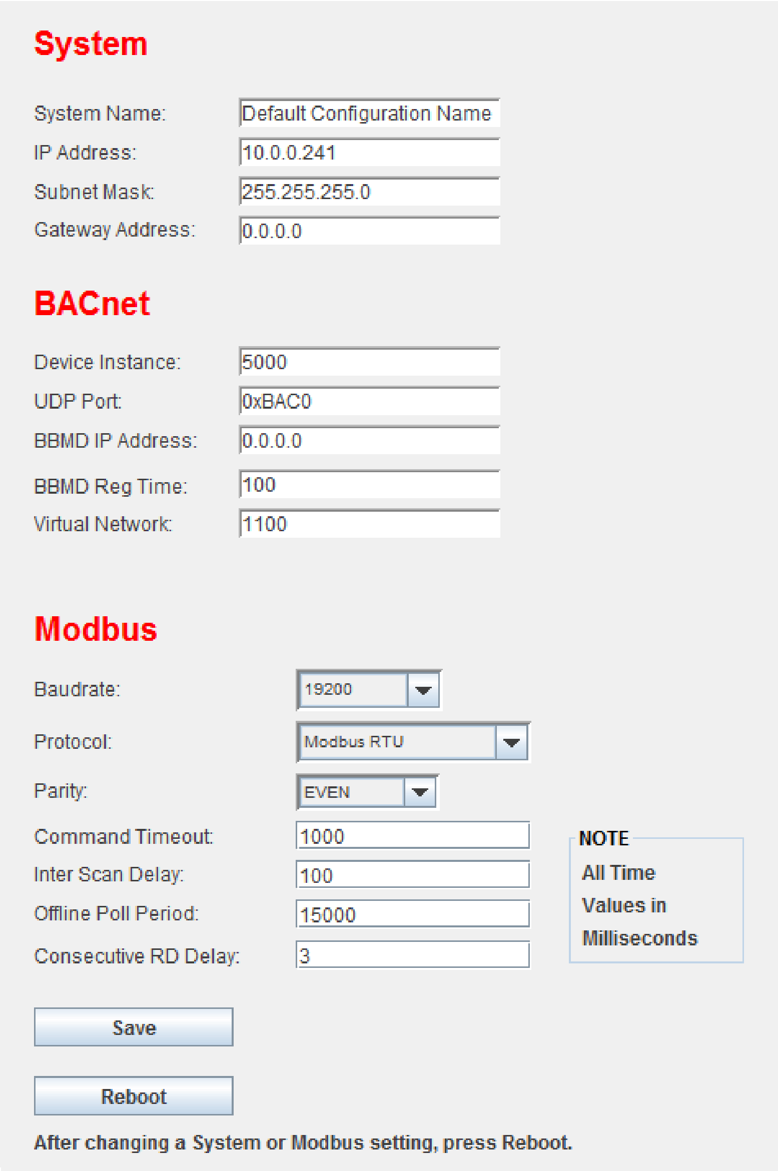


```
Fox_Thermal_Instuments_Model_FT2,,,,,,,,,,,,,,,
YES,0, Flow in Eng unit,10,X400001,32 Bit Unsigned Integer LO/HI,ANALOG_INPUT,1,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Total,10,X400003,32 Bit Unsigned Integer LO/HI,ANALOG_INPUT,2,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Temperature,10,X400005,32 Bit Unsigned Integer LO/HI,ANALOG_INPUT,3,NO_UNITS,Not Defined,None,0,1,0,0.1,1,0
YES,0, Elapsed time,10,X400007,32 Bit Unsigned Integer LO/HI,ANALOG_INPUT,4,NO_UNITS,Not Defined,None,0,1,0,0.1,1,0
YES,0, Velocity,10,X400009,32 Bit Unsigned Integer LO/HI,ANALOG_INPUT,5,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Flow in Eng unit * 10,10,X400011,16 Bit Unsigned Integer,ANALOG_INPUT,6,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Flow in Eng unit *100,10,X400012,16 Bit Unsigned Integer,ANALOG_INPUT,7,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Total *100,10,X400013,16 Bit Unsigned Integer,ANALOG_INPUT,8,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Total2,10,X400014,32 Bit Unsigned Integer LO/HI,ANALOG_INPUT,9,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Status Power up indication,10,X400016,Bit16,BINARY_INPUT,10,NO_UNITS,Not Defined,0,0,1,0,1,1,0
YES,0, Status Flow rate reached high limit threshold,10,X400016,Bit16,BINARY_INPUT,11,NO_UNITS,Not Defined,1,0,1,0,1,1,0
YES,0, Status Flow rate reached high limit threshold,10,X400016,Bit16,BINARY_INPUT,12,NO_UNITS,Not Defined,2,0,1,0,1,1,0
YES,0, Status Temperature reached high limit threshold,10,X400016,Bit16,BINARY_INPUT,13,NO_UNITS,Not Defined,3,0,1,0,1,1,0
YES,0, Status Temperature reached low limit threshold,10,X400016,Bit16,BINARY_INPUT,14,NO_UNITS,Not Defined,4,0,1,0,1,1,0
YES,0, Status Sensor reading is out of range,10,X400016,Bit16,BINARY_INPUT,15,NO_UNITS,Not Defined,5,0,1,0,1,1,0
YES,0, Status Velocity flow rate outside of calibration table,10,X400016,Bit16,BINARY_INPUT,16,NO_UNITS,Not Defined,6,0,1,0,1,1,0
YES,0, Status Incorrect Settings,10,X400016,Bit16,BINARY_INPUT,17,NO_UNITS,Not Defined,7,0,1,0,1,1,0
YES,0, Status In simulation mode,10,X400016,Bit16,BINARY_INPUT,18,NO_UNITS,Not Defined,8,0,1,0,1,1,0
YES,0, Status Frequency output is out of range,10,X400016,Bit16,BINARY_INPUT,19,NO_UNITS,Not Defined,9,0,1,0,1,1,0
YES,0, Status Analog 4-20 mA for flow is out of range,10,X400016,Bit16,BINARY_INPUT,20,NO_UNITS,Not Defined,10,0,1,0,1,1,0
YES,0, Status Analog 4-20 mA for temperature is out of range,10,X400016,Bit16,BINARY_INPUT,21,NO_UNITS,Not Defined,11,0,1,0,1,1,0
YES,0, Status Anybus error,10,X400016,Bit16,BINARY_INPUT,22,NO_UNITS,Not Defined,12,0,1,0,1,1,0
YES,0, Status RTC error (only for FT2 with RTC),10,X400016,Bit16,BINARY_INPUT,23,NO_UNITS,Not Defined,13,0,1,0,1,1,0
YES,0, Status CRC error,10,X400016,Bit16,BINARY_INPUT,24,NO_UNITS,Not Defined,14,0,1,0,1,1,0
YES,0, Status Tot Error,10,X400016,Bit16,BINARY_INPUT,25,NO_UNITS,Not Defined,15,0,1,0,1,1,0
YES,0, Flow in Eng Unit,10,X400020,Floating Point HI/LO,ANALOG_INPUT,26,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Total in Eng Unit,10,X400022,Floating Point HI/LO,ANALOG_INPUT,27,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Total#2 for 2 gas curve,10,X400024,Floating Point HI/LO,ANALOG_INPUT,28,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Temperature in selected unit,10,X400026,Floating Point HI/LO,ANALOG_INPUT,29,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Elapsed time in hours,10,X400028,Floating Point HI/LO,ANALOG_INPUT,30,NO_UNITS,Not Defined,None,0,1,0,1,1,0
YES,0, Velocity in selected unit,10,X400030,Floating Point HI/LO,ANALOG_INPUT,31,NO_UNITS,Not Defined,None,0,1,0,1,1,0
```

Table 8 — Device profile as viewed with Notepad.

Configure Settings

Use this page to make the System, Modbus Serial and BACnet settings.



System

System Name: Default Configuration Name

IP Address: 10.0.0.241

Subnet Mask: 255.255.255.0

Gateway Address: 0.0.0.0

BACnet

Device Instance: 5000

UDP Port: 0xBAC0

BBMD IP Address: 0.0.0.0

BBMD Reg Time: 100

Virtual Network: 1100

Modbus

Baudrate: 19200

Protocol: Modbus RTU

Parity: EVEN

Command Timeout: 1000

Inter Scan Delay: 100

Offline Poll Period: 15000

Consecutive RD Delay: 3

NOTE
All Time Values in Milliseconds

After changing a System or Modbus setting, press Reboot.

System

System Name: BACnet device object name which must be unique network-wide.

IP Address: Changing the default value of 192.168.92.68 is recommended.

Subnet Mask: The default value of 255.255.255.0 is adequate for most users.

Gateway Address: If your Ethernet LAN has a gateway or IP router, enter its address here.

BACnet

Device Instance: Give the BASgatewayLX a unique value (0–4,194,302). Default = 2459900.

UDP Port: The default of 0xBAC0 (47808 in decimal) should normally not be changed.

BBMD IP Address: If the local subnet has no BBMD and the BASgatewayLX must pass data to another subnet, it must register as a Foreign Device with a remote BBMD whose address is entered here.

BBMD Reg Time: Specify the time in seconds between successive foreign device registrations.

Virtual Network: Specify a unique network number for Modbus devices.

Modbus Serial

All devices on the EIA-485 bus must use the same Baudrate, Protocol and Parity.

Baudrate: Choose a value from 2,400 to 115,200. Default = 19,200.

Protocol: Choose RTU or ASCII. Default = RTU.

Parity: Specify NONE, ODD or EVEN. Default = EVEN.

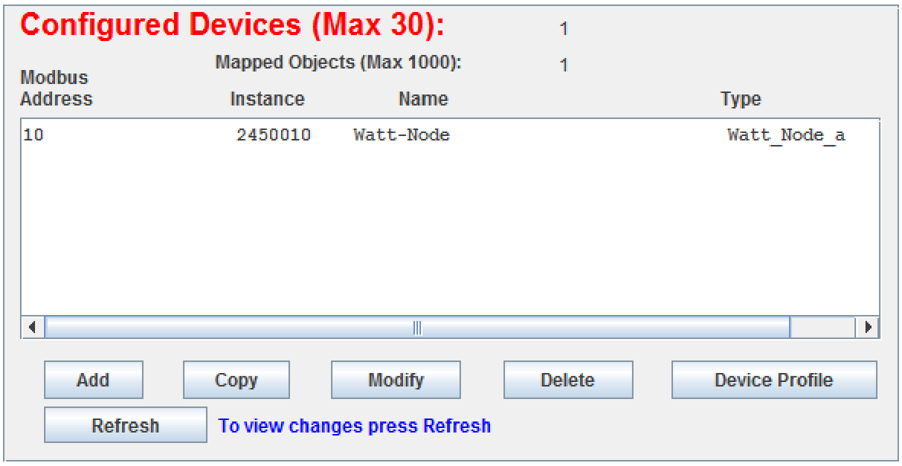
Command Timeout (ms): Specify how long the Master will wait for a slave to response (50–3000). Default = 1,000 ms. If a device fails to respond, it is put in a queue to be checked every Offline Poll Period.

Inter Scan Delay (ms): Specify the delay between each poll cycle (100-30,000). Default = 2,000 ms.

Offline Poll Period (ms): Set how often (2,000–30,000) the Master checks to see if a slave device is back online. Default = 15,000 ms.

Consecutive Access Delay (ms): Set the delay (0–1000) before back-to-back accesses of a slave. Default = 10 ms.

Mapping Configuration



Configured Devices (Max 30): 1

Mapped Objects (Max 1000): 1

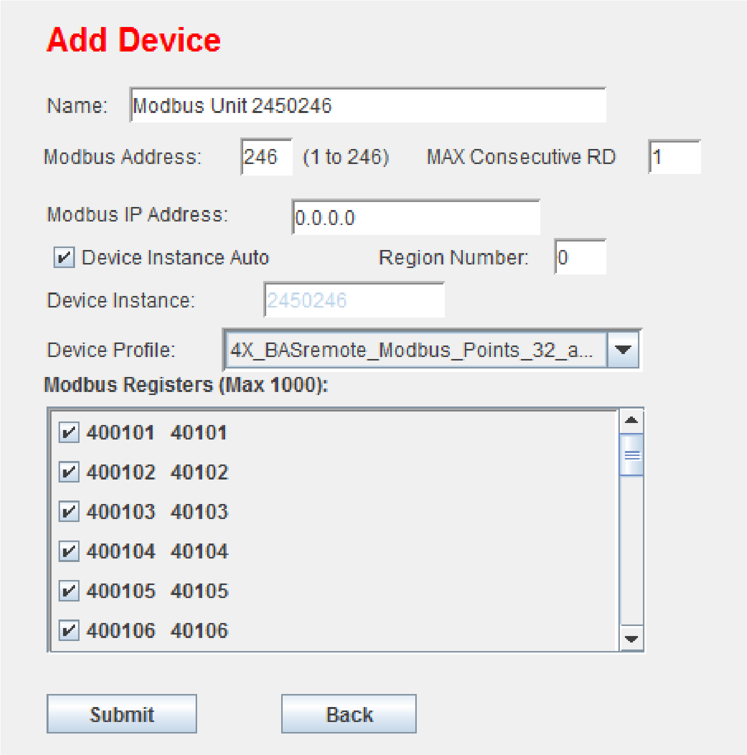
Modbus Address	Instance	Name	Type
10	2450010	Watt-Node	Watt_Node_a

Use this page to add Modbus devices along with device profiles. Each attached Modbus device must have its own device profile.

Use the **Add** button to add a device.

Use the **Copy** button to clone an existing device. Use the **Modify** button to modify an existing device. Use the **Delete** to remove a listed device. Double clicking an item under **Configured Devices** opens a read-only **Device Information** window.

Pressing **Add**, **Copy** or **Modify** opens a new screen where you can Name the device and set its **Modbus Address**. Leave the **Modbus IP Address** at 0.0.0.0 to disable Modbus TCP for this device. A nonzero value assumes a Modbus TCP device and not a Modbus Serial device.



Add Device

Name: Modbus Unit 2450246

Modbus Address: 246 (1 to 246) MAX Consecutive RD 1

Modbus IP Address: 0.0.0.0

☑ Device Instance Auto Region Number: 0

Device Instance: 2450246

Device Profile: 4X_BASremote_Modbus_Points_32_a...

Modbus Registers (Max 1000):

☑ 400101 40101
☑ 400102 40102
☑ 400103 40103
☑ 400104 40104
☑ 400105 40105
☑ 400106 40106

Automatic Device Instance assignments: By default, 7-digit BACnet device instances are assigned automatically beginning with a ‘245’ (Contemporary Controls' Vendor ID) followed by a region digit and a three-digit Modbus slave address (1 to 246). Modbus address 247 is reserved for Contemporary Controls. When using up to 10 BASgatewayLX units on the same BACnet internetwork, assign a different **Region Number** (0–9) for each BASgatewayLX to make auto-created **Device Instances** unique across the BACnet internetwork. Uncheck **Device Instance Auto** to manually enter a **Device Instance**. Any scheme can be used as long as there are not duplicated BACnet device instance assignments. **NOTE:** If your Modbus TCP devices use duplicate Modbus slave addresses, then disable **Device Instance Auto** and manually assign a **Device Instance**.

Device Profiles: Choose a **Device Profile** from the drop-down menu that matches your device. If no match is found, download more profiles from:

www.ccontrols.com/support/basgatewaylx.htm

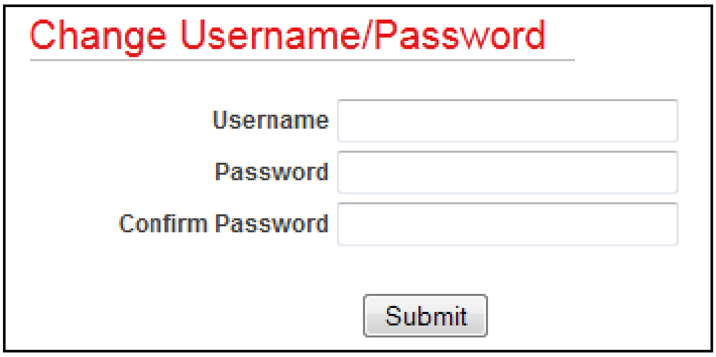
Then upload profiles of interest to the BASgatewayLX. After selecting a **Device Profile**, its **Modbus Registers** will fill the viewing window so you can choose which registers to map to BACnet objects. Only those registers that are clicked will be scanned and then converted to BACnet objects. The maximum number of **Modbus Registers** per BASgatewayLX is 1000 across all connected Modbus devices — TCP or serial. Once configuration is completed, click **Submit** which will take you to the main configuration screen. **NOTE:** To see changes, press the **Refresh** button.

The **Device Profile** button is used to display or delete Device Profiles stored in the BASgatewayLX.

Max Consecutive Read (ms): Set the number of registers (1–125) to be read as a group. Default = 1.

Authentication

A common web browser can be used to view the web pages in the BASgatewayLX but authentication is required to gain access. The default username is *admin* and the default password is *admin*. It is recommended that both the username and password be changed. Only alphanumeric characters can be used for both and the length of each must be five characters or more. If it is necessary to reset the username or password, the device can be reset by depressing a recessed pushbutton on the front of the unit.


Change Username/Password

Username

Password

Confirm Password

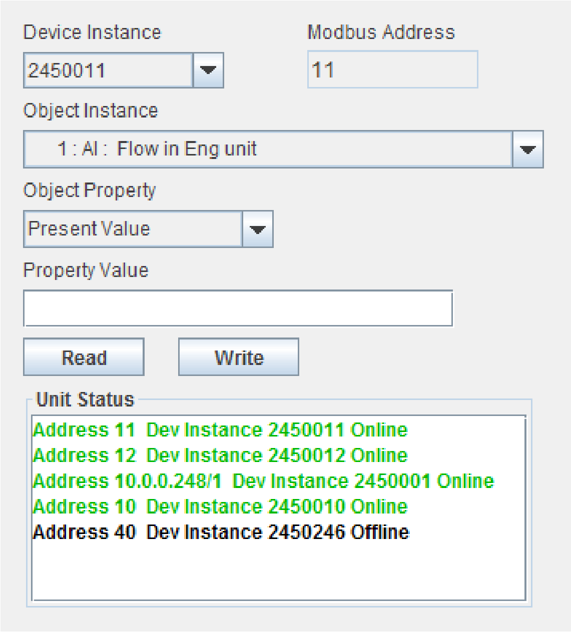
The BASgatewayLX incorporates Java applets so Java Runtime environment (JRE) 6.0 or later must be installed on the device that hosts your web browser.

Mapping Status

Use this page to view the status of mapped Modbus registers to BACnet objects.

Unit Status lists all the Modbus devices that are to be polled. Those listed in green indicate that the device is online and that every register marked for polling can be accessed. Those in black mean that device cannot be reached and therefore considered offline. If the colour changes from green to black frequently, this means that the device is online but a register marked for access cannot be reached. This usually indicates a faulty device profile.

Use the **Device Instance** drop-down to select a device to view or change. Its **Modbus Address** appears to the right. For Modbus serial devices, only the slave address is shown. For Modbus TCP devices, its IP address will precede its slave address. After choosing an **Object Instance** and **Object Property** of the device, click **Read** to view the **Property Value**. If the **Write** button is undimmed, you can enter a **Property Value** to be written. Enter a value and click the **Write** button. **CAUTION:** Understand the significance of making a change to an output before executing the command.


Device Instance: 2450011

Modbus Address: 11

Object Instance: 1 : AI : Flow in Eng unit

Object Property: Present Value

Property Value

Unit Status

Address 11 Dev Instance 2450011 Online
Address 12 Dev Instance 2450012 Online
Address 10.0.0.248/1 Dev Instance 2450001 Online
Address 10 Dev Instance 2450010 Online
Address 40 Dev Instance 2450246 Offline

Adding Device Profiles

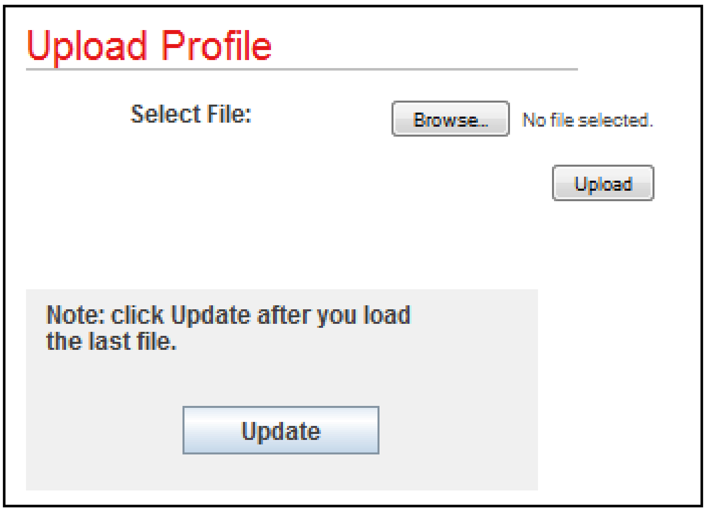
A library of Modbus Device Profiles can be found at:

www.ccontrols.com/support/basgatewaylx.htm

They have a ***.zip*** extension. As more become available, they will be added to this site. To use one of these device profiles, download the zipped file to your PC and unzip the file to reveal the profile in the proper CSV format with a ***.csv*** extension. At this point, you can edit the device profile or simply upload it to the BASgatewayLX.

While in the BASgatewayLX, browse for the file on your computer and once selected click **Upload**. After the file is uploaded click **Update** to see the new device profile on the Configure Modbus window.

Up to 100 device profiles can be stored within the BASgatewayLX — plus up to 30 more selected for scanning.


Upload Profile

Select File: Browse.. No file selected.

Upload

Note: click Update after you load the last file.

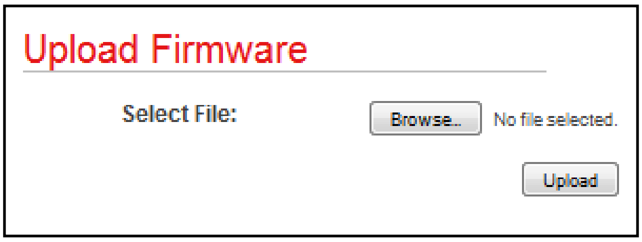
Update

Firmware Updates

The latest BASgatewayLX firmware can be found at:

www.ccontrols.com/support/basgatewaylx.htm

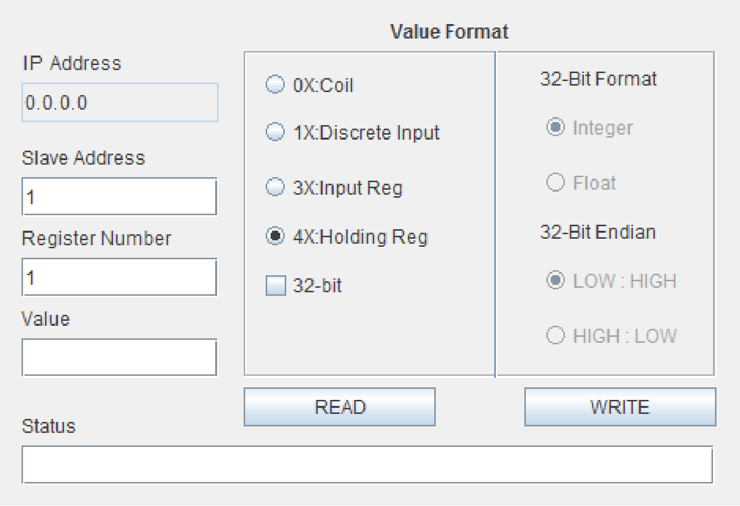
Download the latest firmware to a convenient location on your PC. The file extension is ***.tgz*** and it should not be renamed. While in the BASgatewayLX, browse for the file and click **Upload**.


Upload Firmware

Select File: Browse.. No file selected.

Upload

Modbus Utility



IP Address: 0.0.0.0

Slave Address: 1

Register Number: 1

Value:

Status:

Value Format

○ 0X:Coil
○ 1X:Discrete Input
○ 3X:Input Reg
● 4X:Holding Reg
□ 32-bit

32-Bit Format

● Integer
○ Float

32-Bit Endian

● LOW : HIGH
○ HIGH : LOW

As a convenience to the installer, a resident Modbus Utility (Java applet) is available to verify BASgatewayLX communication to attached Modbus serial and Modbus TCP devices. This utility is best used BEFORE entering device profiles. It is useful in confirming the data structure of Modbus registers within each device.

IP Address: (0.0.0.0 by default.) If the device is a Modbus TCP device, enter its IP address. For Modbus serial devices leave the 0.0.0.0 default address.

Slave Address: (1 by default.) Enter the slave address of the Modbus serial or Modbus TCP device to be accessed (1–246).

Register Number: (1 by default.) Enter only the 5-digit register address but not the leading 1-digit memory block address. This will be specified in the **Value Format** field. For example, to read register 400001, enter the register number 1 and click holding register 4X in the **Value Format** field. Notice that PLC addressing is being used.

Value Format: This reads or writes values for the register in the **Register Number** field. Select one of the memory blocks as follows:

0X: Coil: Check this if the addressed device is a Coil or Binary Output (0 or 1).

1X: Discrete Input: Use this to read a Binary Input (0 or 1).

3X: Input Reg: Use this to read a 16-bit register.

4X: Holding Reg: Use this to read or write a 16-bit register.

32-Bit: This displays a 32-bit value of a register pair and enables the following four buttons:

Integer: Use this for a value with no fractional content.

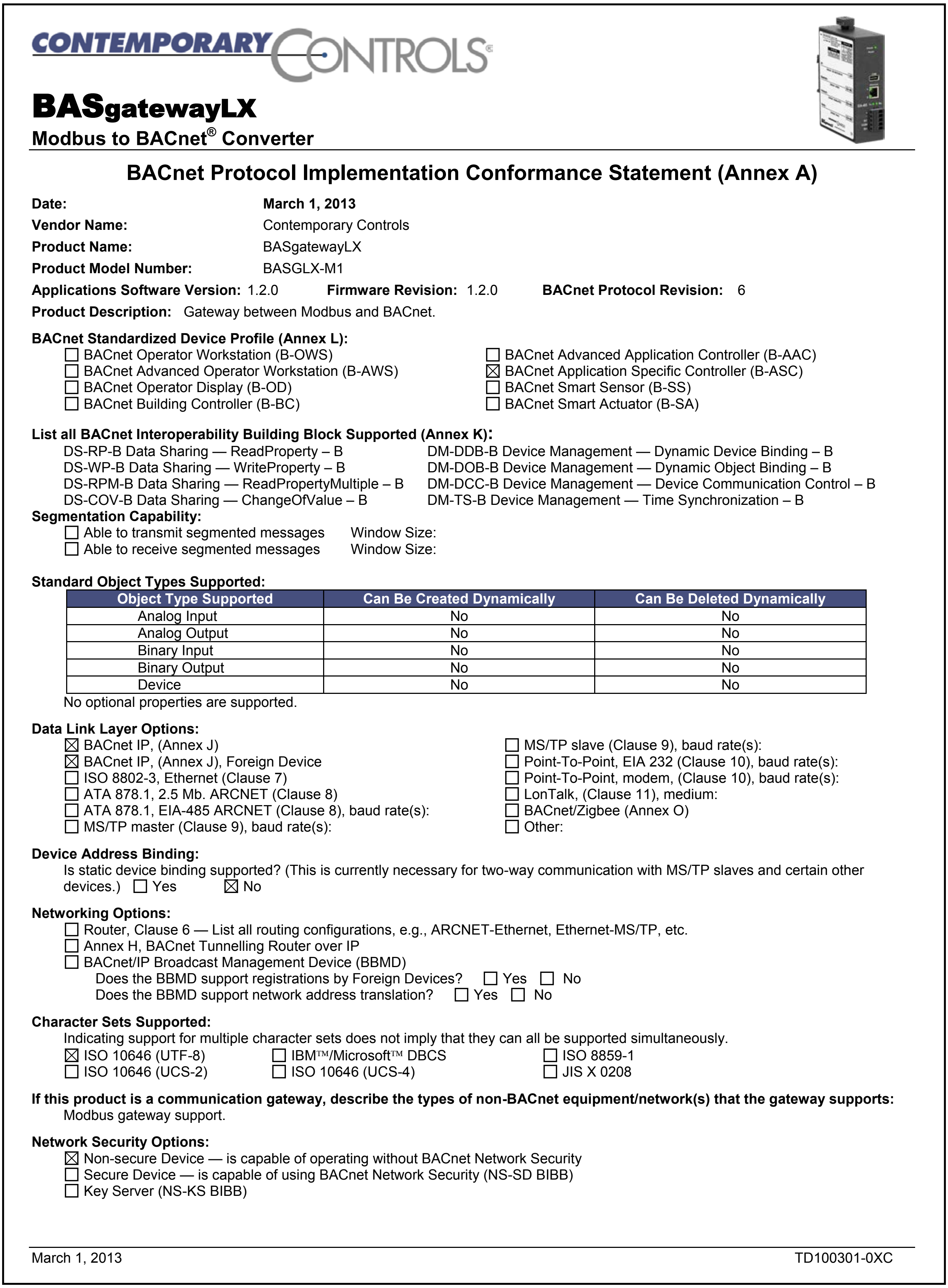
Float: Use this for a single-precision floating-point value.

LOW: HIGH: Use this for Little Endian format (low word is in the base register and high word is in the next-higher register).

HIGH: LOW: Use this for Big Endian format (high word is in the base register and low word is in the next-higher register).

Click on **Read** and the result will appear in the **Value** field. For a write operation, enter the proper value in the **Value** field and click **Write**.

Status: If you read or write a register successfully, “Success” appears here and **Value** shows the result. If a register access is unsuccessful, the error is reported in the **Value** field.



CONTEMPORARY CONTROLS

BASgatewayLX

Modbus to BACnet® Converter

BACnet Protocol Implementation Conformance Statement (Annex A)

Date:	**March 1, 2013**
Vendor Name:	Contemporary Controls
Product Name:	BASgatewayLX
Product Model Number:	BASGLX-M1

Applications Software Version: 1.2.0 **Firmware Revision:** 1.2.0 **BACnet Protocol Revision:** 6

Product Description: Gateway between Modbus and BACnet.

BACnet Standardized Device Profile (Annex L):

☐ BACnet Operator Workstation (B-OWS)
☐ BACnet Advanced Operator Workstation (B-AWS)
☐ BACnet Operator Display (B-OD)
☐ BACnet Building Controller (B-BC)
☐ BACnet Advanced Application Controller (B-AAC)
☒ BACnet Application Specific Controller (B-ASC)
☐ BACnet Smart Sensor (B-SS)
☐ BACnet Smart Actuator (B-SA)

List all BACnet Interoperability Building Block Supported (Annex K):

DS-RP-B Data Sharing — ReadProperty – B
DS-WP-B Data Sharing — WriteProperty – B
DS-RPM-B Data Sharing — ReadPropertyMultiple – B
DS-COV-B Data Sharing — ChangeOfValue – B
DM-DDB-B Device Management — Dynamic Device Binding – B
DM-DOB-B Device Management — Dynamic Object Binding – B
DM-DCC-B Device Management — Device Communication Control – B
DM-TS-B Device Management — Time Synchronization – B

Segmentation Capability:

☐ Able to transmit segmented messages Window Size:
☐ Able to receive segmented messages Window Size:

Standard Object Types Supported:

Object Type Supported	Can Be Created Dynamically	Can Be Deleted Dynamically
Analog Input	No	No
Analog Output	No	No
Binary Input	No	No
Binary Output	No	No
Device	No	No

No optional properties are supported.

Data Link Layer Options:

☒ BACnet IP, (Annex J)
☒ BACnet IP, (Annex J), Foreign Device
☐ ISO 8802-3, Ethernet (Clause 7)
☐ ATA 878.1, 2.5 Mb. ARCNET (Clause 8)
☐ ATA 878.1, EIA-485 ARCNET (Clause 8), baud rate(s):
☐ MS/TP master (Clause 9), baud rate(s):
☐ MS/TP slave (Clause 9), baud rate(s):
☐ Point-To-Point, EIA 232 (Clause 10), baud rate(s):
☐ Point-To-Point, modem, (Clause 10), baud rate(s):
☐ LonTalk, (Clause 11), medium:
☐ BACnet/Zigbee (Annex O)
☐ Other:

Device Address Binding:

Is static device binding supported? (This is currently necessary for two-way communication with MS/TP slaves and certain other devices.) ☐ Yes ☒ No

Networking Options:

☐ Router, Clause 6 — List all routing configurations, e.g., ARCNET-Ethernet, Ethernet-MS/TP, etc.
☐ Annex H, BACnet Tunnelling Router over IP
☐ BACnet/IP Broadcast Management Device (BBMD)
Does the BBMD support registrations by Foreign Devices? ☐ Yes ☐ No
Does the BBMD support network address translation? ☐ Yes ☐ No

Character Sets Supported:

Indicating support for multiple character sets does not imply that they can all be supported simultaneously.

☒ ISO 10646 (UTF-8)
☐ ISO 10646 (UCS-2)
☐ IBM™/Microsoft™ DBCS
☐ ISO 10646 (UCS-4)
☐ ISO 8859-1
☐ JIS X 0208

If this product is a communication gateway, describe the types of non-BACnet equipment/network(s) that the gateway supports:

Modbus gateway support.

Network Security Options:

☒ Non-secure Device — is capable of operating without BACnet Network Security
☐ Secure Device — is capable of using BACnet Network Security (NS-SD BIBB)
☐ Key Server (NS-KS BIBB)

March 1, 2013 TD100301-0XC

United States

**Contemporary Control
Systems, Inc.**
2431 Curtiss Street
Downers Grove, IL 60515
USA

Tel: +1 630 963 7070
Fax:+1 630 963 0109

info@ccontrols.com
www.ccontrols.com

China

**Contemporary Controls
(Suzhou) Co. Ltd**
11 Huoju Road
Science & Technology
Industrial Park
New District, Suzhou
PR China 215009

Tel: +86 512 68095866
Fax: +86 512 68093760

info@ccontrols.com.cn
www.ccontrols.asia

United Kingdom

**Contemporary Controls
Ltd**
14 Bow Court
Fletchworth Gate
Coventry CV5 6SP
United Kingdom

Tel: +44 (0)24 7641 3786
Fax:+44 (0)24 7641 3923

ccl.info@ccontrols.com
www.ccontrols.eu

Germany

**Contemporary Controls
GmbH**
Fuggerstraße 1 B
04158 Leipzig
Germany

Tel: +49 341 520359 0
Fax: +49 341 520359 16

ccg.info@ccontrols.com
www.ccontrols.eu