

## BACnet Protocol Implementation Conformance Statement

**Date:** 12-January-2012

**Vendor Name:** Control Solutions, Inc.

**Product Name:** Babel Buster BB2-3010

**Product Model Number:** BB2-3010

**Applications Software Version:** 3.61 **Firmware Revision:** 3.61 **BACnet Protocol Revision:** 7

### Product Description:

Network gateway allowing Modbus RTU slave devices to be accessed via BACnet IP as a BACnet slave.

### BACnet Standardized Device Profile (Annex L):

- BACnet Operator Workstation (B-OWS)
- 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 Blocks Supported (Annex K):

DS-RP-A DS-RP-B DS-RPM-B DS-WP-A DS-WP-B DS-WPM-B DS-COV-B DS-COVP-B  
DM-DDB-A DM-DDB-B DM-DOB-B DM-DCC-B DM-RD-B DM-R-B

### Segmentation Capability:

- Segmented requests supported                      Window Size: 16
- Segmented responses supported                      Window Size: 16

### Standard Object Types Supported:

An object type is supported if it may be present in the device. For each standard Object Type supported provide the following data:

- 1) Whether objects of this type are dynamically creatable using the CreateObject service
- 2) Whether objects of this type are dynamically deletable using the DeleteObject service
- 3) List of the optional properties supported
- 4) List of all properties that are writable where not otherwise required by this standard
- 5) List of proprietary properties and for each its property identifier, datatype, and meaning
- 6) List of any property range restrictions

### Object types: AI, AO, AV, BI, BO, BV, MSI, MSO, MSV, DEV (all static)

<u>Property (AI, AO, AV, BI, BO, BV, MSI, MSO, MSV)</u>	<u>Encoding</u>
Object_Identifier (75)	BACnetObjectIdentifier
Object_Name (77) (W)	CharacterString "Analog Input <i>n</i> "
Object_Description (28) (W)	Character String Same as Object_Name, is only alias for Object_Name
Object_Type (79)	BACnetObjectType ENUMERATED: analog-input (0)

	<p>analog-output (1)  analog-value (2)  binary-input (3)  binary-output (4)  binary-value (5)  multi-state-input (13)  multi-state-output (14)  multi-state-value (19)</p>
Present_Value (85) (W)	<p>REAL (analog objects)  ENUMERATED (binary objects)  Unsigned (multi-state objects)  (no index)  (priority required when writing commandable objects)  (input objects writeable only when out of service)</p>
Status_Flags (111)	<p>BACnetStatusFlags  BIT STRING: fault(1), out-of-service(3)</p>
Event_State (36)	<p>BACnetEventState  ENUMERATED: normal(0), fault(1)</p>
Reliability (103)	<p>BACnetReliability  ENUMERATED: normal(0)  <i>Vendor specific:</i>  no response (64)  crc error (65)  exception, illegal function code (66)  exception, illegal data address (67)  exception, illegal data value (68)  exception, code+65, rarely used (69..79)  configuration property fault (80)  exception, code not recognized (81)  BACnet client read/write timeout (82)  BACnet client received error response from slave (83)</p>
Out_Of_Service (81) (W)	<p>BOOLEAN</p>
COV_Increment (22) (W)	<p>REAL (analog objects only)</p>
Priority_Array (87)	<p>BACnetPriorityArray (commandable objects only)  SEQUENCE SIZE (16) OF BACnetPriorityValue  REAL (each element, analog output objects)  ENUMERATED (each element, binary output objects)  Unsigned (each element, multi-state output objects)</p>
Relinquish_Default (104) (W)	<p>REAL (analog objects)  ENUMERATED (binary objects)  Unsigned (multi-state objects)</p>
Polarity (84)	<p>BACnetPolarity (binary objects only)  ENUMERATED: normal(0)</p>
Number_Of_States (74)	<p>Unsigned (multi-state objects only)</p>
Units (117)	<p>BACnetEngineeringUnits (analog objects only)</p>
<p><u><i>Vendor Specific Object Properties:</i></u></p>	

Object_Map_Usage (801) (W)	<b>BIT STRING:</b> (0) object is mapped to Modbus (1) object included in packed group (2) object included in mixed type packed group (3) reserved (4) set default on power-up (5) set default on comm. fail (6) read periodic (7) write periodic (8) write on delta (9) enable max quiet (10) object is mapped to BACnet
Poll_Period (802) (W)	Unsigned Modbus poll/update time in seconds
Note: Properties 803 through 808 can only be written when bit 0 of the Object_Map_Usage bit string is set. Properties 819 through 826 can only be written when bit 10 of the Object_Map_Usage bit string is set. Bits 0 and 10 of the Object_Map_Usage bit string cannot both be set at the same time (doing so will cause unpredictable behavior).	
Register_Number (803) (W)	Unsigned Modbus register number 1..65535
Register_Type (804) (W)	<b>ENUMERATED:</b> none(0), coils(1), discrete-input(2), input-register(3), holding-registers(4), coil (5), (write single, FC5) holding-register (6) (write single, FC6)
Register_Format (805) (W)	<b>ENUMERATED:</b> none(0), signed-integer(1), unsigned-integer(2), double-signed-integer(3), double-unsigned-integer(4), floating-point(5), bit(6), double-signed-integer-swapped(7), double-unsigned-integer-swapped(8), floating-point-swapped(9)
Modbus_Slave_ID (806) (W)	<b>ENUMERATED:</b> 1..MAX_RTU_DEVICE
Register_Bit_Mask (807) (W)	Unsigned
Register_Bit_Fill (808) (W)	Unsigned
Slope (809) (W)	<b>REAL</b> BACnet = Modbus * slope + intercept Modbus = (BACnet – intercept) / slope
Intercept (810) (W)	<b>REAL</b>
Default_Value (811) (W)	<b>REAL</b> (analog objects)

	ENUMERATED (binary objects) Unsigned (multi-state objects)
Max_Quiet_Time (812) (W)	Unsigned
BACnet_Slave_Timeout (813) (W)	Unsigned
Max_Read_Fail_Count (814) (W)	Unsigned
Send_On_Delta (815) (W)	REAL (analog objects only)
Initial_COV_Increment (816) (W)	REAL (analog objects only)
Initial_Relinquish_Default (818) (W)	REAL (analog objects) ENUMERATED (binary objects) Unsigned (multi-state objects)
BACnet_Slave_Device (819) (W)	Unsigned
Slave_Object_Type (820) (W)	ENUMERATED
Slave_Object_Instance (821) (W)	Unsigned
Slave_Index (822) (W)	Unsigned (value is index+1, 0=no index)
Slave_Priority (823) (W)	ENUMERATED (1..16)
Slave_Property_Type (824) (W)	ENUMERATED (see BACnet protocol specification for type codes)
Slave_Data-Encoding (825) (W)	ENUMERATED (0) Null, (1) Boolean, (2) Unsigned integer, (3) Signed Integer, (4) Real (floating point), (8) Bit string, (9) Enumerated
Slave_Bit_Position (826) (W)	ENUMERATED
Error_Class (830)	Unsigned
Error_Code (831)	Unsigned
Config_Write_Command (832) (W)	ENUMERATED (1) write config to EEPROM

<u>Property (DEV)</u>	<u>Encoding</u>
Object_Identifier (75)	BACnetObjectIdentifier
Object_Name (77)	CharacterString
Object_Type (79)	BACnetObjectType ENUMERATED: device (8)

System_Status (112)	BACnetDeviceStatus
Vendor_Name (121)	CharacterString
Vendor_Identifier (120)	Unsigned16 (should always return 208)
Model_Name (70)	CharacterString
Fimrware_Revision (44)	CharacterString
Application_Software_Version (12)	CharacterString
Protocol_Version (98)	Unsigned
Protocol_Revision (139)	Unsigned
Protocol_Services_Supported (97)	BACnetServicesSupported
Protocol_Object_Types_Supported (96)	BACnetObjectTypesSupported
Object_List (76)	BACnetARRAY[N] of BACnetObjectIdentifier
Max_APDU_Length_Accepted (62)	Unsigned
Segmentation_Supported (107)	BACnetSegmentation
APDU_Timeout (11)	Unsigned
Number_Of_APDU_Retries (73)	Unsigned
Device_Address_Binding (30)	List of BACnetAddressBinding
Database_Revision (155)	Unsigned
<i>Vendor Specific Object Properties:</i>	
RTU_Port_Baud_Rate (851) (W)	ENUMERATED: (4800, 9600, 19200, 38400)
RTU_Character_Format(852) (W)	BITSTRING: (0) double registers are swapped (1) gateway is slave (2) enable parity (3) set odd parity (even otherwise if parity enabled) (4) enable two stop bits (no parity)
RTU_Slave_Timeout (853) (W)	Unsigned Index: 1..MAX_RTU_DEVICE
RTU_Slave_Address (854) (W)	ENUMERATED (1..247)
RTU_Pre_Delay (855) (W)	Unsigned
RTU_Message_Count (861) (W*)	Unsigned Index: 1..MAX_RTU_DEVICE
RTU_Exception_Count (862) (W*)	Unsigned Index: 1..MAX_RTU_DEVICE

RTU_Bad_CRC_Count (863) (W*)	Unsigned Index: 1..MAX_RTU_DEVICE
RTU_No_Response_Count (864) (W*)	Unsigned Index: 1..MAX_RTU_DEVICE
W* -- Registers are "writeable" for purposes of resetting error counts. Writing any value will reset count to zero. Total message count will stop at 65535. Error counts will stop at 255. Counting will resume when reset to zero.	
RTU_Error_Scan (865) (W)	Unsigned Index: 1..MAX_RTU_DEVICE (index is starting point to scan for next device with errors) (returns next index and error count, or zero) (write index to reset errors for that device, or zero to reset all)
MSTP_Port_Baud_Rate (1201) (W)	ENUMERATED: (9600, 19200, 38400, 76800) NOTE: Changes to port settings will take effect only after a COLDSTART or WARMSTART command is issued.
MSTP_Station_ID (1202) (W)	Unsigned
Enable_Auto_Reset (1203) (W)	BOOLEAN (True if reliability codes should auto return to zero)
Enable_Alt_Map (1204) (W)	BOOLEAN Enables alternate Modbus slave map if set.

**Data Link Layer Options:**

- BACnet IP, (Annex J)
- BACnet IP, (Annex J), Foreign Device
- ISO 8802-3, Ethernet (Clause 7)
- ANSI/ATA 878.1, 2.5 Mb. ARCNET (Clause 8)
- ANSI/ATA 878.1, RS-485 ARCNET (Clause 8), baud rate(s) \_\_\_\_\_
- MS/TP master (Clause 9), baud rate(s): 9600, 19200, 38400,76800
- MS/TP slave (Clause 9), baud rate(s): 9600, 19200, 38400,76800
- Point-To-Point, EIA 232 (Clause 10), baud rate(s): \_\_\_\_\_
- Point-To-Point, modem, (Clause 10), baud rate(s): \_\_\_\_\_
- LonTalk, (Clause 11), medium: \_\_\_\_\_
- 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 Tunneling Router over IP
- BACnet/IP Broadcast Management Device (BBMD)  
Does the BBMD support registrations by Foreign Devices?  Yes  No

**Character Sets Supported:**

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

- |   |   |                                     |
|---|---|-------------------------------------|
| <input checked="" type="checkbox"/> ANSI X3.4 | <input type="checkbox"/> IBM™/Microsoft™ DBCS | <input type="checkbox"/> ISO 8859-1 |
| <input type="checkbox"/> ISO 10646 (UCS-2)    | <input type="checkbox"/> ISO 10646 (UCS-4)    | <input type="checkbox"/> JIS C 6226 |

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

Modbus RTU: BACnet slave device functions as Modbus RTU master or slave. Can also function as MS/TP client with Modbus RTU master.