

MQTT Topics and JSON Data Format

User Manual



ADVANTECH

B+B
SMARTWORX



USA Headquarters

Advantech B+B SmartWorx
707 Dayton Road
Ottawa, IL 61350 USA

Phone (815) 433-5100 -- **General Fax** (815) 433-5105

www.advantech-bb.com
support@advantech-bb.com

European Headquarters

Advantech B+B SmartWorx
Westlink Commercial Park
Oranmore, Co. Galway, Ireland

Phone +353 91-792444 -- **Fax** +353 91-792445

REVISION HISTORY

Version	Date	Description
2.0	06/17/2015	Initial Release
3.0	01/14/2016	Added the sections describing the topics and data for setting configuration parameters using MQTT. Added notes about "ahi" needing to be set greater than "alo" and that these values need to be between the "zp" and "sp" settings.

CONTENTS

REVISION HISTORY.....	3
About This Document.....	6
MQTT Format.....	7
Prefixes and Identifiers.....	7
MQTT Topic Format.....	7
JSON Format.....	7
Sensor Data Format.....	8
Sensor Control.....	11
Sensor Configuration.....	12
1 – GET THE Dust Network Configuration.....	12
2 – SET THE Dust Network Configuration.....	13
3 – GET THE Bluetooth LE Configuration.....	14
4 – SET THE Bluetooth LE Configuration.....	15
5 - GET THE Sensor Platform Configuration.....	16
6 - SET THE Sensor Platform Configuration.....	18
7 – GET THE Sensor List – Sensors Available on the Sensor Platform.....	19
8 – GET AN Analog Input Configuration.....	20
9 – SET AN Analog Input Configuration.....	22
10 – GET A Digital Input Configuration.....	24
11 – SET A Digital Input Configuration.....	27
12 – GET A Digital Output Configuration.....	30
13 – SET A Digital Output Configuration.....	31
14 – GET A Temperature Input Configuration.....	32
15 – SET A Temperature Input Configuration.....	34
16 – GET THE Accelerometer Configuration.....	36

17 – SET THE Accelerometer Configuration.....	37
18 – GET THE Geolocation Configuration.	38
19 – SET THE Geolocation Configuration.....	40
20 – GET THE Supply Voltage Configuration.....	42
21 – SET THE Supply Voltage Configuration.	43
Quality Codes.....	44

ABOUT THIS DOCUMENT

This document contains the messaging details for accessing the published data and controls that are available on the Wzzard Sensor. The sensor uses the MQTT protocol containing data in a JSON format for all of the data being sent across the sensor network.

MQTT is a lightweight broker based publish/subscribe messaging protocol designed for use on low bandwidth networks. JSON is an open standard format that contains data objects consisting of attribute-value pairs in human readable text.

In order to have the 'data' and 'configuration' items sent to a MQTT client, a user will need to subscribe to the broker handling the Wzzard Sensor network so that the data being published by the sensor will be forwarded to the client. The subscribe message will contain the 'topic' that is being requested. This 'topic' represents the area (or grouping) of data that the user wants to see or collect. This document contains the 'topic' values along with what data is available for each 'topic'.

If the user wants to change one of the digital outputs on a sensor, then a 'publish' will need to be sent to the broker handling the Wzzard Sensor network with the new setting. This is handled via the 'ctl' topics supported by the sensor and is covered in the 'Sensor Control' section of this document.

If the user wants to change a configuration setting on a sensor, then a 'publish' will need to be sent to the broker handling the Wzzard Sensor network with the new setting. This is handled via the 'set' topics supported by the sensor and is covered in the ['Sensor Configuration'](#) section of this document.

MQTT FORMAT

PREFIXES AND IDENTIFIERS

1. The prefix for each MQTT topic is in this format: **mfg_id/asset_id/...**
2. **mfg_id**: the OEM identifier for the manufacturer. For B & B Electronics this will be "BB". The mfg_id cannot contain the following MQTT reserved characters: plus ('+'), forward slash ('/'), number sign ('#').
3. **asset_id**: this is the unique identifier for the sensor module. The asset_id cannot contain the following MQTT reserved characters: plus ('+'), forward slash ('/'), number sign ('#').

IMPORTANT: The JSON data sent to each MQTT topic must be 75 bytes or less to be transferred over the wireless network to the sensor platform.

MQTT TOPIC FORMAT

mfg_id/asset_id/topic

JSON FORMAT

S: a sequence number that updates each time a sensor reading is made

t: ISO 8601 timestamp of the UTC time when the sensor reading was made

q: a value that describes the quality of the sensor reading. This value is a bitwise-or of the quality code and the quality limit. For a description of the values, see [Quality Codes](#).

C: a number that is incremented whenever there is a configuration change affecting the reading (e.g., scaling, calibration)

SENSOR DATA FORMAT

This section describes all the data possible for every variant of the Wzzard. An individual Wzzard will have a subset of this data corresponding to the sensors available on the specific model. This data is published on a time based interval that's configured using the 'publish interval' setting of the system configuration.

Data Format (Specified in JSON Schema)

```
{ "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "Sensor Data",
  "description": "Wireless sensor input data.",
  "type": "object",
  "additionalProperties": false,
  "required": ["s", "t", "q", "c"],
  "properties": {
    "s": { "title": "Sequence Number", "description": "A number incremented for every publish of sensor data.",
      "type": "integer", "minimum": 0, "maximum": 9 },
    "t": { "title": "Timestamp", "description": "An ISO 8601 timestamp of the UTC time for the sensor reading.",
      "type": "string", "format": "date-time" },
    "q": { "title": "Quality Code", "description": "A value that indicates quality of the sensor reading.",
      "type": "integer", "minimum": 0, "maximum": 255 },
    "c": { "title": "Configuration Index", "description": "A number increment when the configuration changes.",
      "type": "integer", "minimum": 1, "maximum": 9 },
    "x": { "title": "Accelerometer 1", "description": "Accelerometer x-axis value.",
      "$ref": "#/definitions/accel_value" },
    "y": { "title": "Accelerometer 1", "description": "Accelerometer y-axis value.",
      "$ref": "#/definitions/accel_value" },
    "z": { "title": "Accelerometer 1", "description": "Accelerometer z-axis value.",
      "$ref": "#/definitions/accel_value" },
    "ai1": { "title": "Analog Input 1", "description": "The value of analog input 1.",
      "$ref": "#/definitions/ai_value" },
    "ai2": { "title": "Analog Input 2", "description": "The value of analog input 2.",
      "$ref": "#/definitions/ai_value" },
    "ai3": { "title": "Analog Input 3", "description": "The value of analog input 3.",
      "$ref": "#/definitions/ai_value" },
    "di1": { "title": "Digital Input 1", "description": "The value of digital input 1.",
      "$ref": "#/definitions/di_value" },
    "di2": { "title": "Digital Input 2", "description": "The value of digital input 2.",
      "$ref": "#/definitions/di_value" },
    "do1": { "title": "Digital Output 1", "description": "The state of digital output 1.",
      "$ref": "#/definitions/do_value" },
    "do2": { "title": "Digital Output 2", "description": "The state of digital output 2.",
      "$ref": "#/definitions/do_value" },
```



```

"temp1":{"title":"Temperature Input 1","description":"The value of temperature input 1.",
"$ref":"#/definitions/temp_value"},
"temp2":{"title":"Temperature Input 2","description":"The value of temperature input 2.",
"$ref":"#/definitions/temp_value"},
"tempint":{"title":"Internal Temperature","description":"The internal temperature of the sensor
platform.",
"$ref":"#/definitions/temp_value"},
"vbatt":{"title":"Supply Voltage","description":"The voltage supplied to the sensor platform.",
"$ref":"#/definitions/vbatt_value"}
},
"definitions":{
"ai_value":{"type":"number"},
"di_value":{"oneOf":[{"type":"boolean"},{"type":"number"}]},
"do_value":{"type":"boolean"},
"temp_value":{"type":"number"},
"vbatt_value":{"type":"number"},
"accel_value":{"type":"number","minimum":-9.999,"maximum":9.999}
}
}

```

Description	Example MQTT Topic	Example JSON Data	Notes
Example of the data for accelerometer.	BB/0013430C981F /data	{ "s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "x":-1.234,"y":0.234, "z":-0.234}	x: the x-axis value y: the y-axis value z: the z-axis value The accelerometer values will be four digits, a decimal point and possibly a negative sign.
Example of the data for analog input.	BB/0013430C981A /data	{ "s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "ai1":-1.234567, "ai2":-123.4567} {s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "ai3":-123456.7}	The value of the sensor in engineering units
Example of the data for digital input when configured for a Boolean input.	BB/0013430C981B /data	{ "s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "di1":false,"di2":true}	The value of di1 and di2 can be: false, true.

Example of the data for digital input when configured for counter or frequency.	BB/0013430C981B /data	{ "s":1, "t":"2014-07_11T15:26:37Z", "q":192,"c":1, "di1":1.234567, "di2":123456.7}	The value of di1 and di2 can be: counts or frequency.
Example of the data for digital output.	BB/0013430C981C /data	{ "s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "do1":false,"do2":true}	The value of do1 and do2 can be: false, true
Example of the data for temperature sensor.	BB/0013430C981D /data	{ "s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "temp1":23.6, "temp2":-273.0}	The temperature in the configured engineering units
Example of the data for supply.	BB/0013430C981F /data	{ "s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "vbatt":3.1}	The supply voltage.
Example of the data for internal temperature.	BB/0013430C981ff /data	{ "s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "tempint":-25.2}	Temperature in the configured engineering units

SENSOR CONTROL

These topics are used to control the digital outputs on the sensors that support them. These requests need to be published to the broker handling the Wzzard Sensor network.

Description	MQTT Topic Format	Example JSON Data	Data Format (Specified in JSON Schema)	Notes
Generalized format for setting sensor data.	mfg_id/asset_id/ctl/sensor_id/sensor_instance	Example1: { "v": false }	{ "\$schema": "http://json-schema.org/draft-04/schema#", "description": "Set a wireless sensor output", "type": "object", "additionalProperties": false, "required": ["v"], "properties": { "v": { "description": "The sensor output value specified in engineering units", "type": "boolean" } } }	v: the value in engineering units.

Description	Example MQTT Topic	Example JSON Data	Notes
Set the value of a digital output.	BB/0013430C981F/ctl/dout/1 BB/0013430C981F/ctl/dout/2 BB/0013430C981F/ctl/dout/3 BB/0013430C981F/ctl/dout/4	{ "v": true }	MQTT: do not set the retain bit when publishing messages to this topic; otherwise, an old retained message may change the state of the output.

SENSOR CONFIGURATION

The following sections describe all the configuration settings for every variant of the Wzzard. An individual Wzzard will have a subset of this data corresponding to the sensors available on the specific model. This data is published when the sensor first connects to the network and again when a configuration setting is changed.

1 – GET THE DUST NETWORK CONFIGURATION.

Description	Get the Dust Network configuration.
Example MQTT Topic	BB/0013430C981F/cfg/dust
Example JSON Data	<pre>{ "id":1981, "jd":25.0, "fw":"1.2.0.0", "mac":"0013430C981F", "ol":false}</pre>
JSON Schema	<pre>{ "\$schema":"http://json-schema.org/draft-04/schema#", "type":"object", "additionalProperties":true, "properties":{ "id":{"description":"Dust Network Identifier", "type":"integer","minimum":1,"maximum":65534}, "jd":{"description":"Dust Network Join Duty Cycle (%)", "type":"number","minimum":0.0,"maximum":100.0}, "fw":{"description":"Dust Network firmware version", "type":"string","maxLength":16}, "mac":{"description":"Dust Network MAC address", "type":"string","minLength":16,"maxLength":16,"pattern":"^[0-9A-Fa-f]{16}\$"}, "ol":{"description":"Enable Over-the-Air-Programming", "type":"boolean"}} } }</pre>
Notes	<p>The sensor platform never publishes the Dust Network Join Key.</p> <p>Note: none of the name-value elements are required. This allows a program to send only the name-value elements that have changed.</p> <p>"jd" values will be rounded to the nearest x.0 value in all FW releases beyond version 1.3.x.</p> <p>"ol" is read-only and reserved for future use.</p>

2 – SET THE DUST NETWORK CONFIGURATION.

Requires Wizzard Firmware Version 1.6.x or newer.

Description	Set the Dust Network configuration.
Example MQTT Topic	BB/0013430C981F/set/dust
Example JSON Data	<pre>{"id":1981, "jd":25.0, "jk":"0102030405060708090A0B0C0D0E0F10", "ol":false}</pre>
JSON Schema	<pre>{"\$schema":"http://json-schema.org/draft-04/schema#", "type":"object", "additionalProperties":true, "properties":{ "id":{"description":"Dust Network Identifier", "type":"integer","minimum":1,"maximum":65534}, "jd":{"description":"Dust Network Join Duty Cycle (%)", "type":"number","minimum":0.0,"maximum":100.0}, "jk":{"description":"Dust Network Join Key", "type":"string","minLength":32,"maxLength":32,"pattern":"^[0-9A-Fa-f]{32}\$"}, "ol":{"description":"Enable Over-the-Air-Programming", "type":"boolean"}} }</pre>
Notes	<p>The sensor platform never publishes the Dust Network Join Key.</p> <p>Note: none of the name-value elements are required. This allows a program to send only the name-value elements that have changed.</p> <p>"jd" values will be rounded to the nearest x.0 value in all FW releases beyond version 1.3.x.</p> <p>"ol" is read-only and reserved for future use.</p>

3 – GET THE BLUETOOTH LE CONFIGURATION.

Description	Get the Bluetooth LE configuration.
Example MQTT Topic	BB/0013430C981F/cfg/ble
Example JSON Data	<pre>{ "ae":false, "ar":32, "fw":"1.2.1.3.1.0-PAN1720", "mac":"0013430C981F", "un":"username" }</pre>
JSON Schema	<pre>{ "\$schema":"http://json-schema.org/draft-04/schema#", "type":"object", "additionalProperties":true, "properties":{ "ae":{"description":"Bluetooth LE Advertise Enable", "type":"boolean"}, "ar":{"description":"Bluetooth LE Advertise Rate (Seconds)", "type":"number","minimum":10,"maximum":86400}, "ee":{"description":"Bluetooth LE Enable Extra Encryption", "type":"boolean"}, "fw":{"description":"Bluetooth LE Firmware Version", "type":"string","maxLength":24}, "mac":{"description":"Bluetooth LE MAC Address", "type":"string","minLength":12,"maxLength":12,"pattern":"^[0-9A-Fa-f]{12}\$"}, "un":{"description":"Bluetooth LE Username", "type":"string","maxLength":16,"pattern":"^[^,\\n\\r]+\$"} } }</pre>
Notes	"ee" is reserved for future use.

4 – SET THE BLUETOOTH LE CONFIGURATION.

Requires Wzzard Firmware Version 1.6.x or newer.

Description	Set the Bluetooth LE configuration.
Example MQTT Topic	BB/0013430C981F/set/ble
Example JSON Data	<pre>{ "ae":false, "ar":32, "pw":"password", "un":"username" }</pre>
JSON Schema	<pre>{ "\$schema":"http://json-schema.org/draft-04/schema#", "type":"object", "additionalProperties":true, "properties":{ "ae":{"description":"Bluetooth LE Advertise Enable", "type":"boolean"}, "ar":{"description":"Bluetooth LE Advertise Rate (Seconds)", "type":"number","minimum":10,"maximum":86400}, "ee":{"description":"Bluetooth LE Enable Extra Encryption", "type":"boolean"}, "ek":{"description":"Bluetooth LE Encryption Key", "type":"string","length":64}, "pw":{"description":"Password", "type":"string","maxLength":16,"pattern":"^[^,\n\r]+\$"}, "un":{"description":"Bluetooth LE Username", "type":"string","maxLength":16,"pattern":"^[^,\n\r]+\$"} } }</pre>
Notes	"ee" & "ek" are reserved for future use.

5 - GET THE SENSOR PLATFORM CONFIGURATION

Description	Get the sensor platform system configuration..
Example MQTT Topic	BB/0013430C981F/cfg/sys
Example JSON Data	<pre>{ "aid": "0013430C981F", "fw": "1.2.3", "hw": 6, "md": "2015-02-24", "mi": 86400, "mid": "bb", "mn": "WSD1MA2", "oem": "B+B SmartWorx", "pi": 86400, "schv": 2, "sn": "1234567890123456", "udf": "1234567890123456789012345678901234567890", "bid": "2015-02-25", "ft": "WSD " }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "aid": { "description": "Sensor Platform Asset Identifier", "type": "string", "minLength": 1, "maxLength": 23, "pattern": "^[^#+/{1,23}\$" }, "fw": { "description": "Sensor Platform Firmware Version", "type": "string", "maxLength": 16 }, "hw": { "description": "Sensor Platform Hardware Revision", "type": "number", "minimum": 1, "maximum": 99 }, "md": { "description": "Sensor Platform Manufacture Date", "type": "string" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mid": { "description": "Manufacturer Identifier", "type": "string", "minLength": 1, "maxLength": 8, "pattern": "^[^#+/{1,8}\$" }, "mn": { "description": "Sensor Platform Model Number", "type": "string", "maxLength": 16 }, "oem": { "description": "Manufacturer Name", "type": "string", "maxLength": 16 }, "pi": { "description": "Publish Interval (Seconds)", "type": "integer", "minimum": 10, "maximum": 86400 }, "qos": { "description": "MQTT Quality of Service", "type": "integer", "minimum": 0, "maximum": 2 }, "schv": { "description": "Sensor Platform Schema Version", "type": "integer", "minimum": 1, "maximum": 99 }, "sn": { "description": "Sensor Platform Serial Number", </pre>

	<pre> "type":"string","maxLength":16}, "udf":{"description":"User-Defined Field.", "type":"string","maxLength":40}, "bid":{"description":"Battery Install Date (yyyy-mm-dd)", "type":"string","maxLength":10}, "ft":{"description":"Firmware Type", "type":"string","maxLength":8}} } </pre>
Notes	<p>constraint: publication interval >= measurement interval "qos" is reserved for future use. Do not implement.</p>

6 - SET THE SENSOR PLATFORM CONFIGURATION

Requires Wizzard Firmware Version 1.6.x or newer.

Description	Set the sensor platform system configuration..
Example MQTT Topic	BB/0013430C981F/set/sys
Example JSON Data	<pre>{"mi":86400, "pi":86400, "udf":"1234567890123456789012345678901234567890"}</pre>
JSON Schema	<pre>{"\$schema":"http://json-schema.org/draft-04/schema#", "type":"object", "additionalProperties":true, "properties":{ "mi":{"description":"Measurement Interval (Seconds)", "oneOf":[{"type":"integer","minimum":10,"maximum":86400}, {"type":"integer","minimum":0,"maximum":0}]}, "pi":{"description":"Publish Interval (Seconds)", "type":"integer","minimum":10,"maximum":86400}, "qos":{"description":"MQTT Quality of Service", "type":"integer","mimimum":0,"maximum":2}, "udf":{"description":"User-Defined Field.", "type":"string","maxLength":40}} }</pre>
Notes	constraint: publication interval >= measurement interval "qos" is reserved for future use. Do not implement.

7 – GET THE SENSOR LIST – SENSORS AVAILABLE ON THE SENSOR PLATFORM.

Description	Get the sensors available on the sensor platform.
Example MQTT Topic	BB/0013430C981F/cfg/sensor_list
Example JSON Data	["ain/1", "ain/2", "ain/3", "din/1", "din/2", "dout/1", "dout/2", "temp/1", "temp/2", "temp/int", "accel/1", "geoloc", "vbatt"]
JSON Schema	{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "array", "uniqueItems": true, "items": { "description": "Sensors Available on the Sensor Platform", "type": "string", "pattern": "^[^#+]+\$" } }
Notes	<p>Notice that this is a JSON array instead of a JSON object, so it begins and ends with square brackets (instead of curly braces).</p> <p>Only the sensors available on the specific sensor platform are returned in the array. The array entries can be used as the suffix for the topics "BB/0013430C981F/data" and "BB/0013430C981F/cfg/sensor".</p>

8 – GET AN ANALOG INPUT CONFIGURATION.

Description	Get an analog input configuration.
Example MQTT Topic	BB/0013430C981F/cfg/sensor/ain/1 BB/0013430C981F/cfg/sensor/ain/2 BB/0013430C981F/cfg/sensor/ain/3 BB/0013430C981F/cfg/sensor/ain/4 BB/0013430C981F/cfg/sensor/ain/5 BB/0013430C981F/cfg/sensor/ain/6 BB/0013430C981F/cfg/sensor/ain/7 BB/0013430C981F/cfg/sensor/ain/8
Example JSON Data	<pre>{ "ae":true, "en":true, "mu":"mA", "sl":"12345678901234567890123456789012", "ahi":350.0, "alo":150.0, "sp":400.0, "zp":100.0, "typ":"4-20mA"} </pre>
JSON Schema	<pre> {"\$schema":"http://json-schema.org/draft-04/schema#", "type":"object", "additionalProperties":true, "properties":{ "ae":{"description":"Enable Sensor Advertisement over Bluetooth LE", "type":"boolean"}, "en":{"description":"Sensor Enable", "type":"boolean"}, "mi":{"description":"Measurement Interval (Seconds)", "oneOf":[{"type":"integer","minimum":10,"maximum":86400}, {"type":"integer","minimum":0,"maximum":0}]}, "mu":{"description":"Measurement Unit", "type":"string","maxLength":8}, "pi":{"description":"Publish Interval (Seconds)", "oneOf":[{"type":"integer","minimum":10,"maximum":86400}, {"type":"integer","minimum":0,"maximum":0}]}, "sl":{"description":"Sensor Label", "type":"string","maxLength":32}, "ahi":{"description":"Alert High", "\$ref":"#/definitions/alert_value"}, "alo":{"description":"Alert Low", "\$ref":"#/definitions/alert_value"}, "sp":{"description":"Sensor Span Point", "type":"number","multipleOf":0.01}, </pre>

	<pre> "typ":{"description":"Analog Input Type", "type":"string","enum":["0-20mA","4-20mA","0-5V","1-5V"]}, "zp":{"description":"Sensor Zero Point", "type":"number","multipleOf":0.01}}, "definitions":{ "alert_value":{ "description":"An Alert Value", "oneOf":[{"type":"number","multipleOf":0.01}, {"type":"null"}]}} } </pre>
Notes	<p>Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.</p> <p>The "ahi", "alo", "sp" & "zp" values only support 2 decimal places.</p>

9 – SET AN ANALOG INPUT CONFIGURATION.

Requires Wzzard Firmware Version 1.6.x or newer.

Description	Set an analog input configuration.
Example MQTT Topic	BB/0013430C981F/set/sensor/ain/1 BB/0013430C981F/set/sensor/ain/2 BB/0013430C981F/set/sensor/ain/3 BB/0013430C981F/set/sensor/ain/4 BB/0013430C981F/set/sensor/ain/5 BB/0013430C981F/set/sensor/ain/6 BB/0013430C981F/set/sensor/ain/7 BB/0013430C981F/set/sensor/ain/8
Example JSON Data	<pre>{ "ae":true, "en":true, "mu":"mA", "sl":"12345678901234567890123456789012", "ahi":350.0, "alo":150.0, "sp":400.0, "zp":100.0, "typ":"4-20mA" }</pre>
JSON Schema	<pre>{ "\$schema":"http://json-schema.org/draft-04/schema#", "type":"object", "additionalProperties":true, "properties":{ "ae":{"description":"Enable Sensor Advertisement over Bluetooth LE", "type":"boolean"}, "en":{"description":"Sensor Enable", "type":"boolean"}, "mi":{"description":"Measurement Interval (Seconds)", "oneOf":[{"type":"integer","minimum":10,"maximum":86400}, {"type":"integer","minimum":0,"maximum":0}]}, "mu":{"description":"Measurement Unit", "type":"string","maxLength":8}, "pi":{"description":"Publish Interval (Seconds)", "oneOf":[{"type":"integer","minimum":10,"maximum":86400}, {"type":"integer","minimum":0,"maximum":0}]}, "sl":{"description":"Sensor Label", "type":"string","maxLength":32}, "ahi":{"description":"Alert High", "\$ref":"#/definitions/alert_value"}, "alo":{"description":"Alert Low", "\$ref":"#/definitions/alert_value"}, "sp":{"description":"Sensor Span Point", "type":"number","multipleOf":0.01}, </pre>

	<pre> "typ":{"description":"Analog Input Type", "type":"string","enum":["0-20mA","4-20mA","0-5V","1-5V"]}, "zp":{"description":"Sensor Zero Point", "type":"number","multipleOf":0.01}}, "definitions":{ "alert_value":{ "description":"An Alert Value", "oneOf":[{"type":"number","multipleOf":0.01}, {"type":"null"}]}} } </pre>
Notes	<p>Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.</p> <p>The "ahi", "alo", "sp" & "zp" values only support 2 decimal places.</p> <p>The "ahi" setting must be set to a value greater than the "alo" setting and the "ahi" and "alo" settings must be set within the "zp" and "sp" span for proper operation.</p>

10 – GET A DIGITAL INPUT CONFIGURATION.

Description	Get a digital input configuration.
Example MQTT Topic	BB/0013430C981F/cfg/sensor/din/1 BB/0013430C981F/cfg/sensor/din/2 BB/0013430C981F/cfg/sensor/din/3 BB/0013430C981F/cfg/sensor/din/4
Example JSON Data	<pre>{ "ae": true, "en": true, "mu": "bool", "sl": "12345678901234567890123456789012", "ahi": 1, "alo": 0, "typ": "bool", "tsc": { "inv": false, "fmsg": "closed", "tmsg": "open" } }</pre> <pre>{ "ae": true, "en": true, "mu": "[count]", "sl": "12345678901234567890123456789012", "ahi": null, "alo": null, "typ": "cnt", "tsc": { "inv": false, "mot": 0.001, "mult": 1.5 } }</pre> <pre>{ "ae": true, "en": true, "mu": "[Hz]", "sl": "12345678901234567890123456789012", "ahi": 100.0, "alo": 5.0, "typ": "rate", "tsc": { "period": 1.0, "mot": 0.001, "mult": 1.5 } }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement unit", "type": "string", "maxLength": 8 }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "ahi": { "description": "Alert High", "oneOf": [{ "\$ref": "#/definitions/alert_value_rate" }, </pre>

	<pre> {"\$ref":"#/definitions/alert_value_bool_high"}}, "alo":{"description":"Alert Low", "oneOf":[{"\$ref":"#/definitions/alert_value_rate"}, {"\$ref":"#/definitions/alert_value_bool_low"}]}, "typ":{"description":"Type of Digital Input", "type":"string","enum":["bool","cnt","rate"]}, "tsc":{"description":"Type-Specific Configuration", "oneOf":[{"\$ref":"#/definitions/bool_cfg"}, {"\$ref":"#/definitions/cnt_cfg"}, {"\$ref":"#/definitions/rate_cfg"}]}, "definitions":{ "alert_value_rate":{ "description":"An Alert Value For The Rate", "oneOf":[{"type":"number","multipleOf":0.01}, {"type":"null"}]}, "alert_value_bool_high":{ "description":"An Alert Value For Boolean Alert High", "oneOf":[{"type":"number","minimum":1,"maximum":1}, {"type":"null"}]}, "alert_value_bool_Low":{ "description":"An Alert Value For Boolean Alert Low", "oneOf":[{"type":"number","minimum":0,"maximum":0}, {"type":"null"}]}, "bool_cfg":{ "description":"Boolean Input Configuration", "type":"object", "additionalProperties":false, "required":["inv","fmsg","tmsg"], "properties":{ "inv":{"description":"Invert Input", "type":"boolean"}, "fmsg":{"description":"False Message", "type":"string","maxLength":8}, "tmsg":{"description":"True Message", "type":"string","maxLength":8}}, "cnt_cfg":{ "description":"Counter Input Configuration", "type":"object", "additionalProperties":false, "required":["inv","mot","mult"], "properties":{ "inv":{"description":"Invert Input", "type":"boolean"}, "mot":{"description":"Minimum On Time (Seconds)", "type":"number","minimum":0.000001}, "mult":{"description":"Multiplier", </pre>
--	---

	<pre> "type":"number","minimum":0.000001}}, "rate_cfg":{ "description":"Counter Input Configuration", "type":"object", "additionalProperties":false, "required":["period","mot","mult"], "properties":{ "period":{"description":"Period (Seconds)", "type":"number","minimum":0.01,"multipleOf":0.01}, "mot":{"description":"Minimum On Time (Seconds)", "type":"number","minimum":0.000001}, "mult":{"description":"Multiplier", "type":"number","minimum":0.000001}}}} </pre>
Notes	<p>Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.</p> <p>For the rate type: the "period" option is currently read-only and is set to 1 second. The "ahi" & "alo" values only support 2 decimal places.</p> <p>For the rate and counter types: the "mot" period is reserved for future use. Do not implement.</p>

11 – SET A DIGITAL INPUT CONFIGURATION.

Requires Wzzard Firmware Version 1.6.x or newer.

Description	Set a digital input configuration.
Example MQTT Topic	BB/0013430C981F/set/sensor/din/1 BB/0013430C981F/set/sensor/din/2 BB/0013430C981F/set/sensor/din/3 BB/0013430C981F/set/sensor/din/4
Example JSON Data	<pre>{ "ae": true, "en": true, "mu": "bool", "sl": "12345678901234567890123456789012", "ahi": 1, "alo": 0, "typ": "bool", "tsc": { "inv": false, "fmsg": "closed", "tmsg": "open" } }</pre> <pre>{ "ae": true, "en": true, "mu": "[count]", "sl": "12345678901234567890123456789012", "ahi": null, "alo": null, "typ": "cnt", "tsc": { "inv": false, "mot": 0.001, "mult": 1.5 } }</pre> <pre>{ "ae": true, "en": true, "mu": "[Hz]", "sl": "12345678901234567890123456789012", "ahi": 100.0, "alo": 5.0, "typ": "rate", "tsc": { "period": 1.0, "mot": 0.001, "mult": 1.5 } }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement unit", "type": "string", "maxLength": 8 }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "ahi": { "description": "Alert High", "oneOf": [{ "\$ref": "#/definitions/alert_value_rate" }, </pre>

	<pre> {"\$ref":"#/definitions/alert_value_bool_high"}}, "alo":{"description":"Alert Low", "oneOf":[{"\$ref":"#/definitions/alert_value_rate"}, {"\$ref":"#/definitions/alert_value_bool_low"}]}, "typ":{"description":"Type of Digital Input", "type":"string","enum":["bool","cnt","rate"]}, "tsc":{"description":"Type-Specific Configuration", "oneOf":[{"\$ref":"#/definitions/bool_cfg"}, {"\$ref":"#/definitions/cnt_cfg"}, {"\$ref":"#/definitions/rate_cfg"}]}, "definitions":{ "alert_value_rate":{ "description":"An Alert Value For The Rate", "oneOf":[{"type":"number","multipleOf":0.01}, {"type":"null"}]}, "alert_value_bool_high":{ "description":"An Alert Value For Boolean Alert High", "oneOf":[{"type":"number","minimum":1,"maximum":1}, {"type":"null"}]}, "alert_value_bool_Low":{ "description":"An Alert Value For Boolean Alert Low", "oneOf":[{"type":"number","minimum":0,"maximum":0}, {"type":"null"}]}, "bool_cfg":{ "description":"Boolean Input Configuration", "type":"object", "additionalProperties":false, "required":["inv","fmsg","tmsg"], "properties":{ "inv":{"description":"Invert Input", "type":"boolean"}, "fmsg":{"description":"False Message", "type":"string","maxLength":8}, "tmsg":{"description":"True Message", "type":"string","maxLength":8}}}, "cnt_cfg":{ "description":"Counter Input Configuration", "type":"object", "additionalProperties":false, "required":["inv","mot","mult"], "properties":{ "inv":{"description":"Invert Input", "type":"boolean"}, "mot":{"description":"Minimum On Time (Seconds)", "type":"number","minimum":0.000001}, "mult":{"description":"Multiplier", </pre>
--	--

	<pre> "type":"number","minimum":0.000001}}, "rate_cfg":{ "description":"Counter Input Configuration", "type":"object", "additionalProperties":false, "required":["period","mot","mult"], "properties":{ "period":{"description":"Period (Seconds)", "type":"number","minimum":0.01,"multipleOf":0.01}, "mot":{"description":"Minimum On Time (Seconds)", "type":"number","minimum":0.000001}, "mult":{"description":"Multiplier", "type":"number","minimum":0.000001}}}} </pre>
Notes	<p>Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.</p> <p>For the rate type: the "period" option is currently read-only and is set to 1 second. The "ahi" & "alo" values only support 2 decimal places.</p> <p>The "ahi" setting must be set to a value greater than the "alo" setting for proper operation.</p> <p>For the rate and counter types: the "mot" period is reserved for future use. Do not implement.</p>

12 – GET A DIGITAL OUTPUT CONFIGURATION.

Description	Get a digital output configuration.
Example MQTT Topic	BB/0013430C981F/cfg/sensor/dout/1 BB/0013430C981F/cfg/sensor/dout/2 BB/0013430C981F/cfg/sensor/dout/3 BB/0013430C981F/cfg/sensor/dout/4
Example JSON Data	<pre>{ "ae": false, "en": true, "sl": "12345678901234567890123456789012", "inv": false }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "inv": { "description": "Invert Output", "type": "boolean" } } }</pre>
Notes	Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.

13 – SET A DIGITAL OUTPUT CONFIGURATION.

Requires Wzzard Firmware Version 1.6.x or newer.

Description	Set a digital output configuration.
Example MQTT Topic	BB/0013430C981F/set/sensor/dout/1 BB/0013430C981F/set/sensor/dout/2 BB/0013430C981F/set/sensor/dout/3 BB/0013430C981F/set/sensor/dout/4
Example JSON Data	{ "ae":false, "en": true, "sl":"12345678901234567890123456789012", "inv":false}
JSON Schema	{ "\$schema":"http://json-schema.org/draft-04/schema#", "type":"object", "additionalProperties":true, "properties":{ "ae":{"description":"Enable Sensor Value Advertisement over Bluetooth LE", "type":"boolean"}, "en":{"description":"Sensor Enable", "type":"boolean"}, "mi":{"description":"Measurement Interval (Seconds)", "oneOf":[{"type":"integer","minimum":10,"maximum":86400}, {"type":"integer","minimum":0,"maximum":0}]}, "pi":{"description":"Publish Interval (Seconds)", "oneOf":[{"type":"integer","minimum":10,"maximum":86400}, {"type":"integer","minimum":0,"maximum":0}]}, "sl":{"description":"Sensor Label", "type":"string","maxLength":32}, "inv":{"description":"Invert Output", "type":"boolean"}} }
Notes	Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.

14 – GET A TEMPERATURE INPUT CONFIGURATION.

Description	Get a temperature input configuration.
Example MQTT Topic	BB/0013430C981F/cfg/sensor/temp/1 BB/0013430C981F/cfg/sensor/temp/2 BB/0013430C981F/cfg/sensor/temp/int
Example JSON Data	<pre>{ "ae": true, "en": true, "mu": "C", "sl": "12345678901234567890123456789012", "ahi": 100.0, "alo": 5.0, "typ": "K" }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement Unit", "type": "string", "maxLength": 8, "enum": ["Cel", "[degF]", "C", "F"] }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor label", "type": "string", "maxLength": 32 }, "ahi": { "description": "Alert High", "\$ref": "#/definitions/alert_value" }, "alo": { "description": "Alert Low", "\$ref": "#/definitions/alert_value" }, "typ": { "description": "Type of External Temperature Input", "type": "string", "enum": ["J", "K"] } }, "definitions": { "alert_value": { "description": "An Alert Value", "oneOf": [{ "type": "number", "multipleOf": 0.1 }, { "type": "null" }] } } }</pre>

Notes	<p>Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic. "typ" is only for the external temperature sensors and will not be reported for the internal temperature.</p> <p>The "ahi" & "alo" values only support 1 decimal place.</p>

15 – SET A TEMPERATURE INPUT CONFIGURATION.

Requires Wzzard Firmware Version 1.6.x or newer.

Description	Set a temperature input configuration.
Example MQTT Topic	BB/0013430C981F/set/sensor/temp/1 BB/0013430C981F/set/sensor/temp/2 BB/0013430C981F/set/sensor/temp/int
Example JSON Data	<pre>{ "ae": true, "en": true, "mu": "C", "sl": "12345678901234567890123456789012", "ahi": 100.0, "alo": 5.0, "typ": "K" }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement Unit", "type": "string", "maxLength": 8, "enum": ["Cel", "[degF]", "C", "F"] }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor label", "type": "string", "maxLength": 32 }, "ahi": { "description": "Alert High", "\$ref": "#/definitions/alert_value" }, "alo": { "description": "Alert Low", "\$ref": "#/definitions/alert_value" }, "typ": { "description": "Type of External Temperature Input", "type": "string", "enum": ["J", "K"] } }, "definitions": { "alert_value": { "description": "An Alert Value", "oneOf": [{ "type": "number", "multipleOf": 0.1 }, { "type": "null" }] } } }</pre>

Notes	<p>Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic. "typ" is only for the external temperature sensors and will not be reported for the internal temperature.</p> <p>The "ahi" & "alo" values only support 1 decimal place.</p> <p>The "ahi" setting must be set to a value greater than the "alo" setting for proper operation.</p>

16 – GET THE ACCELEROMETER CONFIGURATION.

Description	Get the accelerometer configuration.
Example MQTT Topic	BB/0013430C981F/cfg/sensor/accel/1
Example JSON Data	<pre>{ "ae": true, "en": true, "mu": "[geoloc]", "sl": "12345678901234567890123456789012", "accg": "2G", "acci": "100Hz" }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement Unit", "type": "string", "maxLength": 8 }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "accg": { "description": "Accelerometer Granularity", "type": "string", "enum": ["2G", "4G", "8G"] }, "acci": { "description": "Accelerometer Interval", "type": "string", "enum": ["12.5Hz", "25Hz", "50Hz", "100Hz", "200Hz", "400Hz"] } } }</pre>
Notes	Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.

17 – SET THE ACCELEROMETER CONFIGURATION.

Requires Wizzard Firmware Version 1.6.x or newer.

Description	Set the accelerometer configuration.
Example MQTT Topic	BB/0013430C981F/set/sensor/accel/1
Example JSON Data	<pre>{ "ae": true, "en": true, "mu": "[geoloc]", "sl": "12345678901234567890123456789012", "accg": "2G", "acci": "100Hz" }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement Unit", "type": "string", "maxLength": 8 }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "accg": { "description": "Accelerometer Granularity", "type": "string", "enum": ["2G", "4G", "8G"] }, "acci": { "description": "Accelerometer Interval", "type": "string", "enum": ["12.5Hz", "25Hz", "50Hz", "100Hz", "200Hz", "400Hz"] } } }</pre>
Notes	Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.

18 – GET THE GEOLOCATION CONFIGURATION.

Description	Get the geolocation configuration.
Example MQTT Topic	BB/0013430C981F/cfg/sensor/geoloc
Example JSON Data	<pre>{ "ae": false, "en": true, "mu": "deg", "sl": "12345678901234567890123456789012", "ovr": { "la": -90.0, "lo": -180.0, "el": 0.0 } }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement Unit", "type": "string", "maxLength": 8 }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "ovr": { "description": "Local Override of Sensor Value", "oneOf": [{ "\$ref": "#/definitions/geolocation" }, { "type": "null" }] }, "definitions": { "geolocation": { "type": "object", "additionalProperties": false, "required": ["la", "lo", "el"], "properties": { "la": { "description": "Latitude", "oneOf": [{ "type": "null" }, { "type": "number", "minimum": -90, "maximum": 90, "multipleOf": 0.0001 }] } } } } } }</pre>

	<pre> "lo":{"description":"Longitude", "oneOf":[{"type":"null"}, {"type":"number","minimum":-180,"maximum":180,"multipleOf":0.0001}]}, "el":{"description":"Elevation", "oneOf":[{"type":"null"}, {"type":"number","multipleOf":0.1}}]} </pre>
Notes	<p>Sensor specific "ae", "en", "mi" and "pi" are reserved for future use. Do not implement. The "ae" and "en" items will be used for sensors with an active GPS type sensor. Use the "mi" and "pi" sensor platform system configuration topic.</p> <p>The "la" & "lo" values support 4 decimal places and the "el" value supports 1 decimal place.</p>

19 – SET THE GEOLOCATION CONFIGURATION.

Requires Wizzard Firmware Version 1.6.x or newer.

Description	Set the geolocation configuration.
Example MQTT Topic	BB/0013430C981F/set/sensor/geoloc
Example JSON Data	<pre>{ "ae": false, "en": true, "mu": "deg", "sl": "12345678901234567890123456789012", "ovr": { "la": -90.0, "lo": -180.0, "el": 0.0 } }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor Enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement Unit", "type": "string", "maxLength": 8 }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "ovr": { "description": "Local Override of Sensor Value", "oneOf": [{ "\$ref": "#/definitions/geolocation" }, { "type": "null" }] }, "definitions": { "geolocation": { "type": "object", "additionalProperties": false, "properties": { "la": { "description": "Latitude", "oneOf": [{ "type": "null" }, { "type": "number", "minimum": -90, "maximum": 90, "multipleOf": 0.0001 }] }, "lo": { "description": "Longitude", "oneOf": [{ "type": "null" }] } } } } } }</pre>

	<pre> {"type":"number","minimum":-180,"maximum":180,"multipleOf":0.0001} }], "el":{"description":"Elevation", "oneOf":[{"type":"null"}, {"type":"number","multipleOf":0.1}}]}} </pre>
Notes	<p>Sensor specific "ae", "en", "mi" and "pi" are reserved for future use. Do not implement. The "ae" and "en" items will be used for sensors with an active GPS type sensor. Use the "mi" and "pi" sensor platform system configuration topic.</p> <p>The "la" & "lo" values support 4 decimal places and the "el" value supports 1 decimal place.</p>

20 – GET THE SUPPLY VOLTAGE CONFIGURATION.

Description	Get the supply voltage configuration.
Example MQTT Topic	BB/0013430C981F/cfg/sensor/vbatt
Example JSON Data	<pre>{ "ae": true, "en": true, "mu": "V", "sl": "12345678901234567890123456789012", "alo": 2.8, "ahi": 3.5 }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement Unit", "type": "string", "maxLength": 8, "enum": ["V"] }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "ahi": { "description": "Alert High", "oneOf": [{ "type": "number", "minimum": 3.0, "maximum": 4.0, "multipleOf": 0.01 }, { "type": "null" }] }, "alo": { "description": "Alert Low", "oneOf": [{ "type": "number", "minimum": 2.4, "maximum": 3.0, "multipleOf": 0.01 }, { "type": "null" }] } } }</pre>
Notes	<p>Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.</p> <p>The "ahi" & "alo" values only support 2 decimal places.</p>

21 – SET THE SUPPLY VOLTAGE CONFIGURATION.

Requires Wzzard Firmware Version 1.6.x or newer.

Description	Set the supply voltage configuration.
Example MQTT Topic	BB/0013430C981F/set/sensor/vbatt
Example JSON Data	<pre>{ "ae": true, "en": true, "mu": "V", "sl": "12345678901234567890123456789012", "alo": 2.8, "ahi": 3.5 }</pre>
JSON Schema	<pre>{ "\$schema": "http://json-schema.org/draft-04/schema#", "type": "object", "additionalProperties": true, "properties": { "ae": { "description": "Enable Sensor Value Advertisement over Bluetooth LE", "type": "boolean" }, "en": { "description": "Sensor enable", "type": "boolean" }, "mi": { "description": "Measurement Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "mu": { "description": "Measurement Unit", "type": "string", "maxLength": 8, "enum": ["V"] }, "pi": { "description": "Publish Interval (Seconds)", "oneOf": [{ "type": "integer", "minimum": 10, "maximum": 86400 }, { "type": "integer", "minimum": 0, "maximum": 0 }] }, "sl": { "description": "Sensor Label", "type": "string", "maxLength": 32 }, "ahi": { "description": "Alert High", "oneOf": [{ "type": "number", "minimum": 3.0, "maximum": 4.0, "multipleOf": 0.01 }, { "type": "null" }] }, "alo": { "description": "Alert Low", "oneOf": [{ "type": "number", "minimum": 2.4, "maximum": 3.0, "multipleOf": 0.01 }, { "type": "null" }] } } }</pre>
Notes	<p>Sensor specific "mi" and "pi" are reserved for future use. Do not implement. Use the "mi" and "pi" sensor platform system configuration topic.</p> <p>The "ahi" & "alo" values only support 2 decimal places.</p> <p>The "ahi" setting must be set to a value greater than the "alo" setting for proper operation.</p>

QUALITY CODES

ID	Major Quality	Quality Sub-Status	Quality Code (Decimal)	Quality Code (Hexadecimal)	Notes on OPC quality and substatus codes.
1	No Quality	No Value	255	0xFF	No quality value is provided.
2	Bad	Non Specific	0	0x00	The value is bad, but no specific reason is known.
3	Bad	Configuration Error	4	0x04	The value is bad, but no specific reason is known.
4	Bad	Not Connected	8	0x08	The input is required to be logically connected to something, but is not. This quality may reflect that no value is available at this time, for reasons like the value may have not been provided by the data source.
5	Bad	Device Failure	12	0x0C	A device failure has been detected.
6	Bad	Sensor Failure	16	0x10	A sensor failure had been detected (the 'Limits' field can provide additional diagnostic information in some situations.)
7	Bad	Last Known Value	20	0x14	Communications have failed; however, the last known value is available. Note that the 'age' of the value may be determined from the timestamp.
8	Bad	Communication Failure	24	0x18	Communications have failed. There is no last known value is available.
9	Bad	Out of Service	28	0x1C	The sensor is off scan or otherwise locked.
10	Uncertain	Non Specific	64	0x40	There is no specific reason why the value is uncertain.
11	Uncertain	Last Usable Value	68	0x44	Whatever was writing this value has stopped doing so. The returned value should be regarded as 'stale'. Note that this differs from a BAD value with Substatus 5 (Last Known Value). That status is associated specifically with a detectable communications error on a 'fetched' value. This error is associated with the failure of some

					external source to 'put' something into the value within an acceptable period of time. Note that the 'age' of the value can be determined from the timestamp.
12	Uncertain	Sensor Not Accurate	80	0x50	Either the value has 'pegged' at one of the sensor limits (in which case the limit field should be set to 1 or 2) or the sensor is otherwise known to be out of calibration via some form of internal diagnostics (in which case the limit field should be 0).
13	Uncertain	EU Units Exceeded	84	0x54	The returned value is outside the limits defined for this parameter. Note that in this case the 'Limits' field indicates which limit has been exceeded, but does NOT necessarily imply that the value cannot move farther out of range.
14	Uncertain	Sub Normal	88	0x58	The value is derived from multiple sources and has less than the required number of Good sources.
15	Good	Non Specific	192	0xC0	The value is good. There are no special conditions.
16	Good	Local Override	216	0xD8	The value has been overridden. Typically this means the input has been disconnected and a manually entered value has been 'forced'.

ID	Quality Limit	Quality Limit (Decimal)	Quality Limit (Hexadecimal)	Notes on OPC quality limits.
1	Not Limited	0	0x00	The value is free to move up or down.
2	Low Limited	1	0x01	The value has 'pegged' at some lower limit.
3	High Limited	2	0x02	The value has 'pegged' at some high limit.
4	Constant	3	0x03	The value is a constant and cannot move.