BiPOM Electronics, Inc. Wireless Solutions for the Real World

WiPORTAL Integration Protocol

Date: 30 September 2025
Document Revision: 1.01B

BiPOM Electronics

Telephone : 1-713-283-9970
E-mail : info@bipom.com
Web : www.bipom.com

Introduction

This document describes the protocol for integrating WiPortal to other systems and applications.

WiPortal has option to push collected data, alarms, and notifications to external services over MQTT or HTTP channels. This can be configured under the **Tools → Integration** section.

When WiPortal receives new data (tag value, alarm, time waveform data) from remote devices or WiPortal itself generates new alarm and/or notification messages WiPortal can send this data to an external system.

To send data to an external system (HTTP server or MQTT broker), WiPortal uses JSON format of payload with the information about each data object.

Why JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language, Standard ECMA-262 3rd Edition - December 1999. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language.

Many JSON formatters/parser for a wide list of languages can be found at www.json.org

Tag and Timewave Data

The basic entity on WiPortal is Tag. A tag is a named variable, much like in a programming language or on a Programmable Logic Controller (PLC tag). Tags may contain real-world inputs such as sensor values, switches, Modbus registers or output values such as actuators, relays, control devices. Tags may also contain interim values and results of calculations.

A tag describes metadata related to its value, such as its name, how to read data, its units, etc. Each Tag has a unique Tag ID.

WiPortal receives a Tag's Value from remote a device in the JSON structure format below. If WiPortal is configured to send this information to an external system, WiPortal uses the same format to send the tag value:

```
{
        "Id"                    : string,
        "Version"               : number,
        "Timestamp"             : string,
        "TagValues"             : [object array],
        "Datasnapshots"         : [object array],
}
```

Id

Unique random number assigned to this message

Version

Version number of the message.
Supported versions: 1

Timestamp

Time in UTC when the message was sent. Time format is following: YYYY-MM-DDTHH:mm:SS.zzzZ

YYYY	– year
MM	– month (1 ... 12)
DD	– day (1 ... 31)
HH	– hours (0 ... 23)
mm	– minutes (0 ... 59)
SS	– seconds (0 ... 59)
zzz	– milliseconds (0...999)

Example: 2025-09-11T12:12:49.334Z

The **Z** at the end of an ISO-8601 timestamp (like 2025-09-30T00:00:00.000Z) means:

- The time is expressed in **UTC** (Coordinated Universal Time).
- Z stands for **Zulu time**, which is the military/aviation designation for UTC.

TagValues
Array of objects where each object represents a Tag Value object.

Datasnapshots
Array of objects where each object represents Time waveform data.

TagValue object

This object represents single tag value and has following structure:

```
{
        "Id"                    : number,
        "DeviceId"              : number,
        "DeviceName"            : string,
        "TagId"                 : number,
        "TagName"               : string,
        "Timestamp"             : string,
        "Value"                         : string,
        "Units"                 : string,
}
```

Id
Unique random number assigned to this message

DeviceId
Unique Device ID from WiPortal Database which owns this tag

DeviceName
Device Name assigned in WiPortal Database

TagId
Unique Tag ID which was assigned on Device configuration. Tags from different Devices may have the same TagId. It is unique only across tags of the same device.

TagName
Tag Name assigned on Device configuration

Timestamp
Time when value was read in UTC. Time format is following: YYYY-MM-DDTHH:mm:SS.zzzZ

YYYY	– year
MM	– month (1 ... 12)
DD	– day (1 ... 31)
HH	– hours (0 ... 23)
mm	– minutes (0 ... 59)
SS	– seconds (0 ... 59)
zzz	– milliseconds (0...999)

Example: 2025-09-11T12:12:49.334Z

Value
Text presentation of tag's value

Units
Units applied to this tag's value. Units set on Device configuration

Timewave data object

This object represents a single tag value and has following structure:

```
{
        "Name"                  : string,
        "DeviceId"              : number,
        "DeviceName"            : string,
        "TagId"                 : number,
        "Timestamp"             : string,
        "XData"                 : array,
        "XUnits"                : string,
        "YData"                 : array,
        "YUnits"                : string
}
```

Name
Name of timewave data tag

DeviceId
Unique Device ID from WiPortal Database which owns this tag

DeviceName
Device Name assigned in WiPortal Database

TagId
Unique Tag ID which was assigned on Device configuration. Tags from different Devices may have the same TagId. It is unique only across tags of the same device.

TagName
Tag Name assigned on Device configuration

Timestamp
Time when value was read in UTC. Time format is following: YYYY-MM-DDTHH:mm:SS.zzzZ

YYYY	– year
MM	– month (1 ... 12)
DD	– day (1 ... 31)
HH	– hours (0 ... 23)
mm	– minutes (0 ... 59)
SS	– seconds (0 ... 59)
zzz	– milliseconds (0...999)

Example: 2025-09-11T12:12:49.334Z

XUnits
Units applied to Axis X values

YUnits
Units applied to Axis Y values

XData
Values of Axis X (usually time ticks)

XData
Values of Axis Y (tag values)

Every index in XData is related to an index in YData. For example, if XData is time ticks and YData is temperature, then the timestamp at index N in XData defines timestamp when value at index N in YData was read. Using these two arrays, the external system can draw an XY chart.