# Codec JSON

□

## Contents



### Codec JSON Description

JSON (JavaScript Object Notation) is a light data exchange format, that due to its simplicity to be written and fast interpretation and generation for machines.

JSON is made up of two structures:

- A collection of name/value pairs. In various languages, this is known as an object, record, structure, dictionary, hash table, key list, or associative array.
- An ordered list of values. In most languages, this is implemented as arrays, vectors, lists, or sequences.

As this Codec 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 widely known 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 language for data exchange.
JSON is an open standard file format and data exchange format used between FMBXXX devices and AWS IoT Shadow Service:

**Its basic principles are:**

- AVL data will be uploaded to Amazon Web Services IOT Shadow service.
- AVL data is JSON key-value based

Data sending procedure and logic are the same as sending to regular servers, except data is being packed to JSON document as required in Amazon Shadow service.

This codec is required to use in the Amazon AWS console, it is important to note that AWS uses TLS encryption protocol, and it must be set up the FMBXXX device properly into the AWS IoT service.

Since the firmware version **03.28.00. Rev.00** AWS JSON codec is supported.

## AWS Shadow

AWS shadow allows to retain device state, this service reports only the latest Shadow state data. Using this mode does not allow sending commands to device or receiving responses.
When using this mode JSON format must be enabled, otherwise, data sending will not begin. Using this service device uses x509 certificates to authenticate. No other method is possible.
https://docs.aws.amazon.com/iot/latest/developerguide/x509-client-certs.html
Doc: https://docs.aws.amazon.com/iot/latest/developerguide/iot-device-shadows.html

## AWS Custom

This mode allows configuring custom topic names, where data will be published and commands received. When using this mode all Codec protocols are supported (Codec8, Codec8E (E-Extended), Codec JSON).

Full Codec description can be found **here**.

Authentication is the same as AWS Shadow.


Codec JSON is available as a transmission Codec, *this parameter can be selected in:*

***Teltonika Configurator >> System >> System Settings >> Data Protocol >> and selecting Codec JSON***

Additionally, to develop the full configuration in the Teltonika device, make sure to use the **MQTT protocol**.

This can be done in ***Teltonika configurator >> GPRS Server Settings >> and in protocol selecting MQTT.***

**Note.** communication to the AWS IoT Service uses a TLS certificate to authenticate, please follow this link where is explained how to obtain it:

**MQTT Start with our devices**

## Codec JSON Structure

```
JSON format:
{
"reported": { - object as per AWS shadow documentation
   "tcxn": {
     "connection_status": 2
   },                    - Current state of device
   "ts": 1510641143000,  - timestamp then record was generated
   "pr": 0, - record priority
```

```
   "lnglat": "0.000000,0.000000", - longitude latitude
   "alt": 0, - altitude
   "ang": 0, - angle
   "sat": 0, - num of visible satellites
   "sp": 0, - speed
   "evt": 0, - IO which generated event
   "21": 3, - „AVL_ID“:value
   "24": 0,
   "66": 13909,
   "67": 0,
   "68": 0,
   "69": 2,
   "80": 0,
   "181": 0,
   "182": 0,
   "200": 0,
   "239": 1,
   "240": 0
 }
}
```

## Sending commands from AWS to the device

Any SMS command, which is interpretable by the device‘s firmware, can be used to control the device that was set up in AWS. SMS/GPRS commands:

**SMS commands List**

The structure for the JSON:

**{ "CMD": "<SMS_Command> <Input>" }**

An example of using the **setdigout** command:

**{ "CMD": "setdigout 111" }**