

# FlexAPI Reference

---

*For LAN application (REST API Version)*

# Revision History

Revision	Date	Author	Item(s) changed	Note
1.0.0	29/7/2020	dengzt, ganjx, wucl	Create document.	
1.0.1	25/8/2020	dengzt, ganjx, wucl	Added cellular1 and system info group GNSS group added <code>gnss.num_sv</code>	
1.0.2	23/10/2020	wangzy	Added Userdata group.	
1.0.3	26/10/2020	dengzt	sysinfo group add <code>sysinfo.lan_mac</code> , <code>sysinfo.wlan_mac</code> and <code>sysinfo.wlan_5g_mac</code>	
1.0.4	9/11/2020	dengzt	Add APP group	
1.0.5	27/1/2021	wangzy	Added 1-wire group	

# 1. Introduction

We introduced FlexAPI for the fast evolving IoT applications, which highly value easy integration, openness, flexibility, extensibility and programmability.

FlexAPI is designed to be efficient, clean and ready to use. It's network oriented and programming language independent, and is ideal for application integration inside the vehicle.

FlexAPI provides unified data and control service via REST API for LAN access.

For data service, we have ready to use reserved groups such as: GNSS, OBD, Motion, IO and Summary.

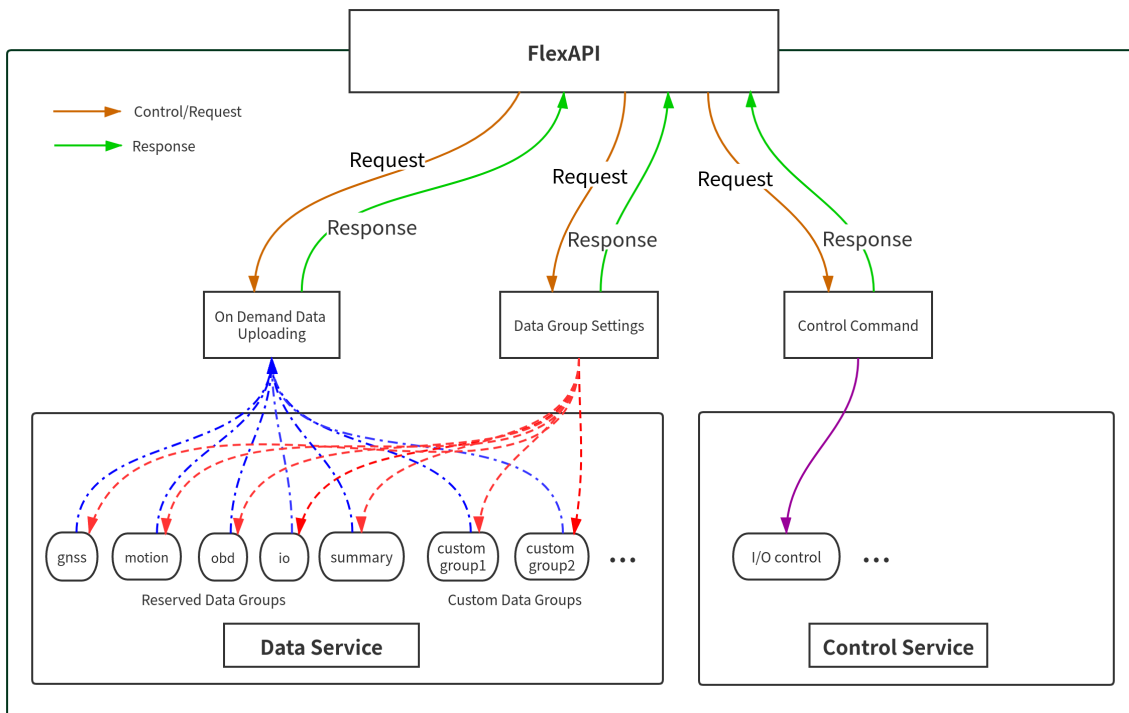
Note that the Summary group is the all in one data group which includes all the data from our reserved OBD, GNSS, Motion and IO groups.

In general, reserved groups are enough for user's need.

FlexAPI also provides REST API for users to apply control, such as turn on/off the digital output.

For advanced users, they can even define their interested groups.

## 1.1 Architecture



## 1.2 REST API Introduction

**RE**presentational **S**tate **T**ransfer (REST) is an architectural style that defines a set of constraints to be used for creating web services. **REST API** is a way of accessing the web services in a simple and flexible way without having any processing.

## 1.3 REST API settings

REST API is disabled by default, you need to enable it first on VG710 for your subsequent service access.

### APP >> REST API

#### REST API Management

Enable	Localhost & LAN ▾
Localhost Listen Address	http://127.0.0.1:5432
LAN Listen Address	https://10.5.16.33:60000
LAN Access Token	l7B2U0Nm7Oz5E1E4AWCzgudn7q4l0C32 <a href="#">Refresh Token</a>
Include Invalid Data	<input type="checkbox"/>
FlexAPI Config File	<a href="#">Import</a> <a href="#">Export</a> <a href="#">Restore default configuration</a>

- **Enable:** Options include **None**, **localhost** and **localhost & LAN**.
- **Localhost Listen Address:** Server settings for APPs run on VG710
- **LAN Listen Address:** Server settings for APPs within VG710 LAN
- **LAN Access Token:** Required **ONLY** for **Bearer** authentication in HTTP header for LAN Access.
- **Include Invalid Data:** if enabled, FlexAPI will also return invalid data items with `null` value besides valid data items.
- **FlexAPI Config File:** Mange FlexAPI configuration file of REST API

## 2 FlexAPI Overview

FlexAPI organizes data as groups and provides ready to use reserved groups for users to develop their applications.

FlexAPI allow users to change reserved and custom group settings.

This overview part gives summary on: FlexAPI general information, error codes and supported REST APIs.

For Basic Usage, see [3. Basic usage](#).

For Advanced Usage, see [4. Advanced usage](#).

For FlexAPI supported Parameters, see [Appendix A. FlexAPI supported Parameters](#).

### FlexAPI Limits :

Resource	Limit
Minimum retry interval of <code>settings</code> , <code>refresh</code> , <code>get</code> requests	2 s
Minimum retry interval of <code>io control</code> request	5 s
Available custom groups	up to 16
Maximum data items per group	256

## 2.1 FlexAPI Return information and Errors

### 2.1.1 General information

Parameter Name	Description	Type	Note
result	result	object	When the request succeeds, there will be result field in response message body.
error	error code	string	When the request fails, it is added to the response message body. For more information, see <a href="#">General Error Codes</a>
error_desc	error description	string	When the request fails, it is added to the response message body. For more information, see <a href="#">General Error Codes</a>
ts	time stamp	number	UNIX timestamp since Epoch. Indicates when the message was transmitted by device.

### 2.1.2 General Error Codes

Status code	Error Code	Description	Error Handling
401	auth_failed	authentication failed	check username and password
400	invalid_parameter	invalid parameter	check request parameter
404	not_found	resource not exist	make sure related service is enabled and running
503	device_busy	device busy	retry request
500	device_error	device internal error	retry request
500	data_invalid	resource invalid	retry request

## 2.2 FlexAPI supported APIs

### 2.2.1 Data service

#### 2.2.1.1 Reserved group settings

Reserved groups provide ready to use data service.

Users can use the following REST APIs to define their interested data group.

URL	HTTP Method	Description
/v1/summary/set	POST	Set Summary group setting. See <a href="#">Summary settings</a> .
/v1/obd/set	POST	Set OBD group setting. See <a href="#">OBD settings</a> .
/v1/gnss/set	POST	Set GNSS group setting. See <a href="#">GNSS settings</a> .
/v1/motion/set	POST	Set Motion group setting. See <a href="#">Motion settings</a> .
/v1/io/set	POST	Set IO group setting. See <a href="#">IO settings</a> .
v1/cellular1/set	POST	Set Cellular1 group setting. see <a href="#">Cellular1 settings</a> .
v1/userdata/set	POST	Set User data group setting. see <a href="#">User data settings</a> .
v1/1-wire/set	POST	Set User data group setting. see <a href="#">User data settings</a> .

### 2.2.1.2 Get reserved group data

Users can use the following REST APIs to get data.

URL	HTTP Method	Description
/v1/summary/refresh	GET	Get Summary data. See <a href="#">Summary Data</a> .
/v1/obd/refresh	GET	Get OBD data. See <a href="#">OBD data</a> .
/v1/gnss/refresh	GET	Get GNSS data. See <a href="#">GNSS Data</a> .
/v1/motion/refresh	GET	Get Motion data. See <a href="#">Motion Data</a> .
/v1/io/refresh	GET	Get IO data. See <a href="#">IO Data</a> .
v1/cellular1/refresh	GET	Get Cellular1 data. see <a href="#">Cellular1 Data</a> .
v1/sysinfo/refresh	GET	Get system data. see <a href="#">System Info</a> .
v1/userdata/refresh	GET	Get User data. see <a href="#">User data</a> .
v1/app/refresh	GET	Get APP data. see <a href="#">APP data</a> .
v1/1-wire/refresh	GET	Get 1-wire data. see <a href="#">1-wire data</a> .

## 2.2.2 Control Service

### 2.2.2.1 IO control

Users can use the following REST APIs to turn on/off the digital output.

URL	HTTP Method	Description
/v1/io/control	POST	IO control. See <a href="#">IO Control</a> .

### 2.2.2.2 APP control

Users can use the following REST APIs to notify APP to do something.

URL	HTTP Method	Description
/v1/app/control	POST	APP control. See <a href="#">APP Control</a> .

## 2.2.3 Advanced usage

Advanced users can use the following REST APIs to define their interested groups.

### 2.2.3.1 Custom group settings



### 2.2.3.1.1 Create/Update custom group

URL	HTTP Method	Description
/v1/group/set	POST	Create/Update group. See <a href="#">Create/Update custom group</a> .

### 2.2.3.1.2 Get custom group settings

URL	HTTP Method	Description
/v1/group/get	GET	Get group settings. See <a href="#">Get custom group settings</a> .

### 2.2.3.1.3 Remove custom group

URL	HTTP Method	Description
/v1/group/set	POST	Remove group. See <a href="#">Remove custom group</a> .

### 2.2.3.2 Get custom group data

URL	HTTP Method	Description
/v1/{group_name}/refresh	GET	Get group data. See <a href="#">Get custom group data</a> .

## **3. Basic usage**

---

### **3.1 Reserved group settings**

#### **3.1.1 General settings**

Parameter Name	Description	Type	Range	Units	Optional	Note
interest	<p>interest parameter</p> <p>List of interested item, each item is represented as key: alias. alias is used in reported messages to rewrite key, a value of "" means no alias.</p> <p>For example, set interest with alias:</p> <pre>{"obd.mil": "MIL", "obd.dtcs": "dtcNum"}</pre> <p>reported data: {"MIL": "1", "dtcNum": "3"}</p> <p>set interest without alias:</p> <pre>{"obd.mil": "", "obd.dtcs": ""}</pre> <p>reported data: {"obd.mil": "1", "obd.dtcs": "3"}</p>	object			optional	<p>'key': FlexAPI Supported parameters</p> <p>'alias': parameter alias</p> <p>OBD group, see <a href="#">OBD Parameters</a></p> <p>GNSS group, see <a href="#">GNSS Parameters</a></p> <p>Motion group, see <a href="#">Motion Parameters</a></p> <p>IO group, see <a href="#">IO Parameters</a></p>

For both reserved and custom groups, you can redefine your interested data, simply use the following API to specify your interested data items using `interest` parameter.

#### Request Syntax :

**Note :** `{deviceIP}` and `{port}` differ per your access mode( `localhost` or `localhost & LAN` ), please refer to listen address of [REST API settings](#).

**Note:** Authorization field in HTTP request header is only required for LAN access mode. So if your APPs run on VG710, you don't need to specify this field. Also note that all the examples in this document are for LAN access mode.

```
1 POST /v1/{group_name}/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-type: application/json
5
6 {
7     "interest": {
8         "gnss.latitude": "lat",
9         "gnss.longitude": "lon",
10        "obd.speed": "speed",
11        "obd.odo": ""
12    }
13 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "interest": {
7             "gnss.latitude": "lat",
8             "gnss.longitude": "lon",
9             "obd.speed": "speed",
10            "obd.odo": ""
11        }
12    }
13 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#).

### 3.1.2 Summary settings

Use this REST API to set your interested summary data.

Note that the Summary group is the all in one data group which includes all the data from our reserved OBD, GNSS, Motion and IO groups.

Default interest is available parameters from the [FlexAPI supported Parameters](#).

#### Request Syntax :

```
1  POST /v1/summary/set HTTP/1.1
2  Host: {deviceIP}:{port}
3  Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4  Content-Type: application/json
5
6  {
7      "interest": {
8          "gnss.latitude": "lat",
9          "gnss.longitude": "lon",
10         "obd.speed": "speed",
11         "obd.odo": ""
12     }
13 }
```

#### Response Syntax:

Success:

```
1  HTTP/1.1 200
2  Content-type: application/json
3
4  {
5      "result": {
6          "interest": {
7              "gnss.latitude": "lat",
8              "gnss.longitude": "lon",
9              "obd.speed": "speed",
10             "obd.odo": ""
11          }
12      }
13 }
```

Failure:

```
1  HTTP/1.1 400
2  Content-type: application/json
3
4  {
5      "error": "invalid_parameter",
6      "error_desc": "Invalid request parameter"
7  }
```

Parameter description, see [General Information](#).

### 3.1.3 OBD settings

Use this REST API to set your interested OBD data.

Default interest is available parameters from the [OBD Parameters](#).

#### Request Syntax :

```
1 POST /v1/obd/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-Type: application/json
5
6 {
7   "interest": {
8     "obd.mil": "MIL",
9     "obd.dtcs": "dtcNum",
10    "obd.rpm": "engineSpeed"
11  }
12 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5   "result": {
6     "interest": {
7       "obd.mil": "MIL",
8       "obd.dtcs": "dtcNum",
9       "obd.rpm": "engineSpeed"
10    }
11  }
12 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5   "error": "invalid_parameter",
6   "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#).

### 3.1.4 GNSS settings

Use this REST API to set your interested GNSS data.

Default interest is available parameters from the [GNSS Parameters](#).

#### Request Syntax :

```
1 POST /v1/gnss/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-Type: application/json
5
6 {
7     "interest": {
8         "gnss.latitude": "lat",
9         "gnss.longitude": "lon",
10        "gnss.altitude": "alt"
11    }
12 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "interest": {
7             "gnss.latitude": "lat",
8             "gnss.longitude": "lon",
9             "gnss.altitude": "alt"
10        }
11    }
12 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#).

### 3.1.5 Motion settings

Use this REST API to set your interested motion data.

Default interest is available parameters from the [Motion Parameters](#).

#### Request Syntax :

```
1 POST /v1/motion/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-Type: application/json
5
6 {
7     "interest": {
8         "motion.ax": "acceleration_x",
9         "motion.ay": "acceleration_y",
10        "motion.az": "acceleration_z"
11    }
12 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "interest": {
7             "motion.ax": "acceleration_x",
8             "motion.ay": "acceleration_y",
9             "motion.az": "acceleration_z"
10        }
11    }
12 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#).



### 3.1.6 IO settings

Use this REST API to set your interested IO data.

Default interest is available parameters from the [IO Parameters](#).

#### Request Syntax :

```
1 POST /v1/io/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-Type: application/json
5
6 {
7     "interest": {
8         "io.AI1": "ai1",
9         "io.AI2": "ai2",
10        "io.AI3": "ai3"
11    }
12 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "interest": {
7             "io.AI1": "ai1",
8             "io.AI2": "ai2",
9             "io.AI3": "ai3"
10        }
11    }
12 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#).

### 3.1.7 Cellular1 settings

Use this REST API to set your interested cellular1 data.

Default interest is available parameters from the [Cellular Parameters](#).

#### Request Syntax :

```
1 POST /v1/cellular1/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer 0A25eJ643f9J7ZI59tc96X7NA6p9Md3g
4 Content-Type: application/json
5
6 {
7     "interest": {
8         "modem1.active_sim": "active_sim",
9         "modem1.signal_lvl": "signal_lvl",
10        "cellular1.status": "status"
11    }
12 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "interest": {
7             "modem1.active_sim": "active_sim",
8             "modem1.signal_lvl": "signal_lvl",
9             "cellular1.status": "status"
10        }
11    }
12 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#).

## 3.1.8 User data settings

### 3.1.8.1 Insert user data

Use this REST API to insert your user data.

#### Request Syntax :

```
1 POST /v1/userdata/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer 0A25eJ643f9J7ZI59tc96X7NA6p9Md3g
4 Content-Type: application/json
5
6 {
7     "insert": {
8         "userdata.custom_key": "custom_value",
9         "userdata.serial_number": "SN0125"
10    }
11 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "inserted": {
7             "userdata.custom_key": "custom_value",
8             "userdata.serial_number": "SN0125"
9         }
10    }
11 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#).

### 3.1.8.2 Update user data

Use this REST API to update your user data.

#### Request Syntax :

```
1 | POST /v1/userdata/set HTTP/1.1
2 | Host: {deviceIP}:{port}
3 | Authorization: Bearer 0A25eJ643f9J7ZI59tc96X7NA6p9Md3g
4 | Content-Type: application/json
5 |
6 | {
7 |     "update": {
8 |         "userdata.serial_number": "SN0232"
9 |     }
10 | }
```

### Response Syntax :

Success :

```
1 | HTTP/1.1 200
2 | Content-type: application/json
3 |
4 | {
5 |     "result": {
6 |         "updated": {
7 |             "userdata.serial_number": "SN0232"
8 |         }
9 |     }
10 | }
```

Failure :

```
1 | HTTP/1.1 400
2 | Content-type: application/json
3 |
4 | {
5 |     "error": "invalid_parameter",
6 |     "error_desc": "Invalid request parameter"
7 | }
```

Parameter description, see [General Information](#).

### 3.1.8.3 Set User Data interest

Use this REST API to set your interested user data.

### Request Syntax :

```
1 | POST /v1/userdata/set HTTP/1.1
2 | Host: {deviceIP}:{port}
3 | Authorization: Bearer 0A25eJ643f9J7ZI59tc96X7NA6p9Md3g
4 | Content-Type: application/json
5 |
6 | {
7 |     "interest": {
8 |         "userdata.custom_key": "custom_key",
9 |         "userdata.serial_number": "serial_number"
10 |     }
11 | }
```

## Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "interest": {
7             "userdata.custom_key": "custom_key",
8             "userdata.serial_number": "serial_number"
9         }
10    }
11 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#).

### 3.1.8.4 Delete user data

Use this REST API to delete your user data.

## Request Syntax :

```
1 POST /v1/userdata/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer 0A25eJ643f9J7ZI59tc96X7NA6p9Md3g
4 Content-Type: application/json
5
6 {
7     "delete": {
8         "userdata.serial_number": "serial_number"
9     }
10 }
```

## Response Syntax :

Success :

```
1 | HTTP/1.1 200
2 | Content-type: application/json
3 |
4 | {
5 |     "result": {
6 |         "deleted": {
7 |             "userdata.serial_number": "serial_number"
8 |         }
9 |     }
10 | }
```

Failure :

```
1 | HTTP/1.1 400
2 | Content-type: application/json
3 |
4 | {
5 |     "error": "invalid_parameter",
6 |     "error_desc": "Invalid request parameter"
7 | }
```

Parameter description, see [General Information](#).

### 3.1.9 1-Wire settings

Use this REST API to set your interested 1-wire data.

Default interest is available parameters from the [1-wire Parameters](#).

#### Request Syntax :

```
1 | POST /v1/1-wire/set HTTP/1.1
2 | Host: {deviceIP}:{port}
3 | Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 | Content-Type: application/json
5 |
6 | {
7 |     "interest": {
8 |         "1-wire.temp1_data" : "data1",
9 |         "1-wire.temp1_id" : "ID1",
10 |        "1-wire.temp1_name" : "name1"
11 |     }
12 | }
```

#### Response Syntax :

Success :

```
1 | HTTP/1.1 200
2 | Content-type: application/json
3 |
4 | {
5 |     "result": {
6 |         "interest": {
7 |             "1-wire.temp1_data": "data1",
8 |             "1-wire.temp1_id": "ID1",
9 |             "1-wire.temp1_name": "name1"
10 |        }
11 |     }
12 | }
```

Failure :

```
1 | HTTP/1.1 400
2 | Content-type: application/json
3 |
4 | {
5 |     "error": "invalid_parameter",
6 |     "error_desc": "Invalid request parameter"
7 | }
```

Parameter description, see [General Information](#).

## 3.2 Get reserved group data

### 3.2.1 Summary data

Use this REST API to get summary data.

**REST API :** `/v1/summary/refresh`

**Request Syntax :**

```
1 GET /v1/summary/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

**Response Syntax :**

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5   "result": {
6     "gnss.latitude": 40.232213,
7     "gnss.longitude": 116.34366,
8     "gnss.altitude": 346.0,
9     "gnss.speed": 87.6,
10    "gnss.heading": 234.0,
11    "gnss.hdop": 1.2,
12    "gnss.pdop": 2.1,
13    "gnss.hacc": 1.0,
14    "gnss.fix": 3,
15    "gnss.num_sv": 7,
16    "gnss.date": "2020-4-17",
17    "gnss.time": "10:16:21",
18    "obd.rpm" : 1234,
19    "obd.speed" : 20,
20    "obd.odo": 1400,
21    "obd.up_time": 3600,
22    "io.AI1": 0.0,
23    "io.AI2": 0.0,
24    "io.AI3": 0.0,
25    "io.AI4": 0.0,
26    "io.AI5": 0.0,
27    "io.AI6": 0.0,
28    "io.DI1": 0,
29    "io.DI1_pullup": 0,
30    "io.DI2": 0,
31    "io.DI2_pullup": 0,
32    "io.DI3": 0,
33    "io.DI3_pullup": 0,
34    "io.DI4": 0,
35    "io.DI4_pullup": 0,
36    "io.DI5": 0,
37    "io.DI5_pullup": 0,
38    "io.DI6": 0,
```



```
39     "io.DI6_pullup": 0,  
40     "io.D01": 0,  
41     "io.D01_pullup": 0,  
42     "io.D02": 0,  
43     "io.D02_pullup": 0,  
44     "io.D03": 0,  
45     "io.D03_pullup": 0,  
46     "io.D04": 0,  
47     "io.D04_pullup": 0  
48   }  
49 }
```

Failure :

```
1  HTTP/1.1 400  
2  Content-type: application/json  
3  
4  {  
5    "error": "invalid_parameter",  
6    "error_desc": "Invalid request parameter"  
7  }
```

Parameter description, see [General Information](#) & [FlexAPI supported Parameters](#).

### 3.2.2 OBD data

Use this REST API to get OBD data.

#### Request Syntax :

```
1 GET /v1/obd/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "obd.rpm": 34245,
7         "obd.speed": 53255
8     }
9 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) & [OBD Parameters](#).

### 3.2.3 GNSS data

Use this REST API to get GNSS data.

#### Request Syntax :

```
1 GET /v1/gnss/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5   "result": {
6     "gnss.latitude": 40.232213,
7     "gnss.longitude": 116.34366,
8     "gnss.altitude": 346.0,
9     "gnss.speed": 87.6,
10    "gnss.heading": 234.0,
11    "gnss.hdop": 1.2,
12    "gnss.pdop": 2.1,
13    "gnss.hacc": 1.0,
14    "gnss.fix": 3,
15    "gnss.num_sv": 7,
16    "gnss.date": "2020-4-17",
17    "gnss.time": "10:16:21"
18  }
19 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5   "error": "invalid_parameter",
6   "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) & [GNSS Parameters](#).

### 3.2.4 Motion data

Use this REST API to get motion data.

#### Request Syntax :

```
1 GET /v1/motion/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5   "result": {
6     "motion.ax": 0.08,
7     "motion.ay": 0.0,
8     "motion.az": 0.0,
9     "motion.gx": 0.15,
10    "motion.gy": 0.03,
11    "motion.gz": -0.47,
12    "motion.roll": -0.65,
13    "motion.pitch": 1.03,
14    "motion.yaw": 302.49
15  }
16 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5   "error": "invalid_parameter",
6   "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) & [Motion Parameters](#).

### 3.2.5 IO data

Use this REST API to get IO data.

#### Request Syntax :

```
1 GET /v1/io/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5   "result": {
6     "io.AI1": 0.0,
7     "io.AI2": 0.0,
8     "io.AI3": 0.0,
9     "io.AI4": 0.0,
10    "io.AI5": 0.0,
11    "io.AI6": 0.0,
12    "io.DI1": 0,
13    "io.DI1_pullup": 0,
14    "io.DI2": 0,
15    "io.DI2_pullup": 0,
16    "io.DI3": 0,
17    "io.DI3_pullup": 0,
18    "io.DI4": 0,
19    "io.DI4_pullup": 0,
20    "io.DI5": 0,
21    "io.DI5_pullup": 0,
22    "io.DI6": 0,
23    "io.DI6_pullup": 0,
24    "io.D01": 0,
25    "io.D01_pullup": 0,
26    "io.D02": 0,
27    "io.D02_pullup": 0,
28    "io.D03": 0,
29    "io.D03_pullup": 0,
30    "io.D04": 0,
31    "io.D04_pullup": 0
32  }
33 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) & [IO Parameters](#).

## 3.2.6 Cellular1 data

Use this REST API to get cellular1 data.

### Request Syntax :

```
1 GET /v1/cellular1/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer 0A25eJ643f9J7ZI59tc96X7NA6p9Md3g
```

### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "modem1.ts": 1598425245,
7         "modem1.active_sim": 1,
8         "modem1.imei": "862104021247207",
9         "modem1.imsi": "460013231603009",
10        "modem1.iccid": "89860118802836799717",
11        "modem1.signal_lvl": 29,
12        "modem1.reg_status": 1,
13        "modem1.operator": "46001",
14        "modem1.network": 3,
15        "modem1.lac": "EA00",
16        "modem1.cell_id": "71CF520",
17        "cellular1.ts": 1598425316,
18        "cellular1.status": 3,
19        "cellular1.ip": "10.210.255.168",
20        "cellular1.netmask": "255.255.255.255",
21        "cellular1.gateway": "1.1.1.3",
22        "cellular1.dns1": "119.7.7.7",
23        "cellular1.dns2": "119.6.6.6",
24        "cellular1.up_at": 1598424985,
25        "cellular1.down_at": 0,
26        "cellular1.traffic_ts": 1598425316,
27        "cellular1.tx_bytes": 83777,
28        "cellular1.rx_bytes": 30258
29    }
30 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) & [Cellular Parameters](#).



### 3.2.7 System Info

Use this REST API to get system info data.

#### Request Syntax :

```
1 GET /v1/sysinfo/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer 0A25eJ643f9J7ZI59tc96X7NA6p9Md3g
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5   "result": {
6     "sysinfo.ts": 1598424935,
7     "sysinfo.language": "Chinese",
8     "sysinfo.hostname": "VG710",
9     "sysinfo.timezone": "UTC-8",
10    "sysinfo.model_name": "VG710",
11    "sysinfo.oem_name": "inhand",
12    "sysinfo.serial_number": "VG7102019052101",
13    "sysinfo.firmware_version": "1.0.0.r13083",
14    "sysinfo.bootloader_version": "2012.07.r235",
15    "sysinfo.product_number": "TL01",
16    "sysinfo.description": "www.inhand.com.cn",
17    "sysinfo.lan_mac": "00:18:05:10:99:66",
18    "sysinfo.wlan_mac": "00:18:05:10:99:03",
19    "sysinfo.wlan_5g_mac": "00:18:05:10:99:04"
20   }
21 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5   "error": "invalid_parameter",
6   "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) & [System Parameters](#).

### 3.2.8 User data

Use this REST API to get your user data.

#### Request Syntax :

```
1 GET /v1/sysinfo/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer 0A25eJ643f9J7ZI59tc96X7NA6p9Md3g
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "userdata.custom_key": "custom_value",
7         "userdata.serial_number": "SN0125"
8     }
9 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) .

### 3.2.9 APP data

Use this REST API to get APP data.

#### Request Syntax :

```
1 GET /v1/app/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "app.wifi_mode_2g": 0,
7         "app.wifi_mode_5g": 0
8     }
9 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) & [APP Parameters](#).

### 3.2.10 1-wire data

Use this REST API to get 1-wire data.

#### Request Syntax :

```
1 GET /v1/1-wire/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5   "result": {
6     "1-wire.ts": 1644560984",
7     "1-wire.status" : "Connected",
8     "1-wire.type" : "Temperature & ROM Code",
9     "1-wire.temp_num" : 2,
10    "1-wire.rom_num" : 1,
11    "1-wire.temp1_data" : 24.06,
12    "1-wire.temp1_id" : "aa012029901e7928",
13    "1-wire.temp1_name" : "Inside",
14    "1-wire.temp2_data" : 23.69,
15    "1-wire.temp2_id" : "27012029cf6a8328",
16    "1-wire.temp2_name" : "Outside",
17    "1-wire.rom_code1" : "cc00001b559ae001"
18  }
19 }
20
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5   "error": "invalid_parameter",
6   "error_desc": "Invalid request parameter"
7 }
```

Parameter description, reference [General Information](#) & [1-wire Parameters](#).

## 3.3 Control Service

### 3.3.1 IO Control

Use this REST API to turn on/off the digital output.

#### Request Syntax :

```
1 POST /v1/io/control HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-Type: application/json
5
6 {
7     "io.D01": 0,
8     "io.D01_pullup": 0,
9     "io.D02": 0,
10    "io.D02_pullup": 0,
11    "io.D03": 0,
12    "io.D03_pullup": 0,
13    "io.D04": 0,
14    "io.D04_pullup": 0
15 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "io.D01": 0,
7         "io.D01_pullup": 0,
8         "io.D02": 0,
9         "io.D02_pullup": 0,
10        "io.D03": 0,
11        "io.D03_pullup": 0,
12        "io.D04": 0,
13        "io.D04_pullup": 0
14    }
15 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#) & [IO Parameters](#) digital output part.

### 3.3.2 APP Control

Use this REST API to notify APP to do something.

#### Request Syntax :

```
1 POST /v1/app/control HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-Type: application/json
5
6 {
7     "app.wifi_mode_2g": 0,
8     "app.wifi_mode_5g": 0
9 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "app.wifi_mode_2g": 0,
7         "app.wifi_mode_5g": 0
8     }
9 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#) & [APP Parameters](#) digital output part.

## 4. Advanced usage

### 4.1 Custom group settings

#### 4.1.1 Create/Update custom group

Use this REST API to define your interested groups.

For `interest` parameters, please refer to [General settings](#) for details.

#### Request Syntax :

```
1 POST /v1/group/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-Type: application/json
5
6 {
7     "settings": [
8         {
9             "group_name": "group1",
10            "interest": {
11                "gnss.latitude": "lat",
12                "gnss.longitude": "lon",
13                "gnss.altitude": "alt",
14                "obd.speed": "speed",
15                "obd.odo": "odo",
16                "userdata.custom_key": "custom_value"
17            }
18        },
19        {
20            "group_name": "group2",
21            "interest": {
22                "io.DI1": "DI1",
23                "io.DI2": "DI2",
24                "io.DI3": "DI3",
25                "io.DI4": "DI4",
26                "io.D01": "D01",
27                "io.D02": "D02",
28                "io.D03": "D03"
29            }
30        }
31    ]
32 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": [
6         {
7             "group_name": "group1",
```

```

 8         "interest": {
 9             "gnss.latitude": "lat",
10             "gnss.longitude": "lon",
11             "gnss.altitude": "alt",
12             "obd.speed": "speed",
13             "obd.odo": "odo",
14             "userdata.custom_key": "custom_value"
15         }
16     },
17     {
18         "group_name": "group2",
19         "interest": {
20             "io.DI1": "DI1",
21             "io.DI2": "DI2",
22             "io.DI3": "DI3",
23             "io.DI4": "DI4",
24             "io.D01": "D01",
25             "io.D02": "D02",
26             "io.D03": "D03"
27         }
28     }
29 ]
30 }

```

Failure :

```

 1 HTTP/1.1 400
 2 Content-type: application/json
 3
 4 {
 5     "error": "invalid_parameter",
 6     "error_desc": "Invalid request parameter"
 7 }

```

Parameter description, see [General Information](#) & [General settings](#).



## 4.1.2 Get custom group settings

Use this REST API to get custom group settings.

### Request Syntax :

```
1 GET /v1/group/get HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5   "result": [
6     {
7       "group_name": "group1",
8       "interest": {
9         "gnss.latitude": "lat",
10        "gnss.longitude": "lon",
11        "gnss.altitude": "alt",
12        "obd.speed": "speed",
13        "obd.odo": "odo",
14        "userdata.custom_key": "custom_value"
15      }
16    },
17    {
18      "group_name": "group2",
19      "interest": {
20        "io.DI1": "DI1",
21        "io.DI2": "DI2",
22        "io.DI3": "DI3",
23        "io.DI4": "DI4",
24        "io.D01": "D01",
25        "io.D02": "D02",
26        "io.D03": "D03"
27      }
28    }
29  ]
30 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5   "error": "invalid_parameter",
6   "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#) & [General settings](#).

### 4.1.3 Remove custom group

Use this REST API to remove group.

#### Request Syntax :

```
1 POST /v1/group/set HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
4 Content-Type: application/json
5
6 {
7     "settings": [{
8         "group_name": "group1",
9         "interest": null
10    }, {
11        "group_name": "group2",
12        "interest": null
13    }
14 ]
15 }
```

#### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "settings": [{
6         "group_name": "group1",
7         "interest": null
8     }, {
9         "group_name": "group2",
10        "interest": null
11    }
12 ]
13 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#) & [General settings](#).

## 4.2 Get custom group data

Use this REST API to get `group_name` data.

### Request Syntax :

```
1 GET /v1/{group_name}/refresh HTTP/1.1
2 Host: {deviceIP}:{port}
3 Authorization: Bearer iWUFB4y7720f841yLcR10dLTuo2T04JR
```

### Response Syntax :

Success :

```
1 HTTP/1.1 200
2 Content-type: application/json
3
4 {
5     "result": {
6         "lat": 40.232213,
7         "ai1": 1.0,
8         "obd.speed": 50,
9         "userdata.custom_key": "custom_value"
10    }
11 }
```

Failure :

```
1 HTTP/1.1 400
2 Content-type: application/json
3
4 {
5     "error": "invalid_parameter",
6     "error_desc": "Invalid request parameter"
7 }
```

Parameter description, see [General Information](#) & [FlexAPI supported Parameters](#).

# Appendix A. FlexAPI supported Parameters

## A.1 GNSS Parameters

Parameter Name	Description	Type	Range	Units	Optional	Note
gnss.latitude	latitude	float		deg	mandatory	
gnss.longitude	longitude	float		deg	mandatory	
gnss.altitude	altitude	float		deg	mandatory	
gnss.speed	speed	float		km/h	mandatory	from GNSS modem
gnss.heading	heading	float	[0.0,360.0]	°		
gnss.hdop	Horizontal DOP	float				
gnss.pdop	Position DOP	float				
gnss.hacc	Horizontal accuracy estimate	float		m		
gnss.fix	GNSS fix status	int	0: NoFix; 1: DR Only 2: 2D; 3: 3D 4: GNSS+DR; 5: Time Only			
gnss.num_sv	number of satellites used	int	[0,12]			
gnss.date	date	string	format: yy-mm-dd			
gnss.time	time	string	format: hh:mm:ss			

## A.2 Motion Parameters

Parameter Name	Description	Type	Range	Units	Optional	Note
motion.ax	x-axis accelerometer	float		g	mandatory	accelerometer
motion.ay	y-axis accelerometer	float		g	mandatory	accelerometer
motion.az	z-axis accelerometer	float		g	mandatory	accelerometer
motion.gx	x-axis gyroscope	float		deg/s	mandatory	gyroscope
motion.gy	y-axis gyroscope	float		deg/s	mandatory	gyroscope
motion.gz	z-axis gyroscope	float		deg/s	mandatory	gyroscope
motion.roll	roll angle	float		deg	mandatory	
motion.pitch	pitch angle	float		deg	mandatory	
motion.yaw	yaw angle	float		deg	mandatory	

### A.3 IO Parameters

Parameter Name	Description	Type	Range	Units	Optional	Note
io.AI{n}	Analog Input n	float	[0,36.0] null : invalid	V	mandatory	n: [1,6]
io.DI{n}	Digital Input n	int	0: low 1: high null : invalid		mandatory	n: [1,6]
io.DI{n}_pullup	Digital Input pullup n	int	0: down 1: up null : invalid		mandatory	n: [1,6]
io.DO{n}	Digital Output n	int	0: low 1: high null : invalid		mandatory	n: [1,4]
io.DO{n}_pullup	Digital Output pullup n	int	0: down 1: up null : invalid		mandatory	n: [1,4]

# A.4 OBD Parameters



Parameter Name	Description	Type	Range	Units	Optional	Note
obd.vin	Vehicle Identification Number	string				
obd.e_load	Engine Load	double	[0,250] 0: stopped >0: started	%		
obd.c_temp	Engine Coolant Temp	int	[-40,215]	°C		
obd.rpm	Engine Speed	double	[0,16383.75]	RPM		
obd.speed	Vehicle Speed	int	[0,255]	km/h		
obd.f_lvl	Fuel Level	double	[0,100]	%		
obd.f_rate	Fuel Rate	double	[0,3276.75]	l/h		
obd.dtcs	DTC Count	int	[0,250]			
obd.mil	MIL Status	boolean	0:off 1:on			
obd.b_volt	Battery Voltage	double	[0,3212.75]	V		
obd.a_temp	Ambient Air Temp	int	[-273,1734]	°C		
obd.o_temp	Engine Oil Temp	int	[-273,1734]	°C		
obd.up_time	Engine Start Time	int	[0,65535]	sec		
obd.m_dist	Distance traveled while MIL is Activated	int	[0,65535]	km		
obd.d_dist	Distance traveled since DTCs cleared	int	[0,65535]	km		
obd.m_time	Engine run time while MIL activated	int	[0,65535]	min		
obd.d_time	Engine run time since DTCs cleared	int	[0,65535]	min		
obd.f_press	Fuel Pressure	int	[0,6425]	kPa		
obd.t_pos	Throttle Position	double	[0,100]	%		
obd.brake	Brake Switch Status	boolean	0:brake pedal released 1:brake pedal depressed			
obd.parking	Parking Brake Switch Status	boolean	0:parking brake not set 1:parking brake set			
obd.s_w_angle	Steering Wheel Angle	double	[-31.374,31.374]	rad		
obd.f_econ	Fuel Economy	double	[0,125.50]	km/L		
obd.odo	Odometer	double	[0,526385151.875]	km		
obd.a_pos	Accelerator Pedal Position	double	[0,100]	%		
obd.t_dist	trip distance	double	[0,526385151.875]	km		

Parameter Name	Description	Type	Range	Units	Optional	Note
obd.i_temp	Intake Manifold Temp	int	[-40,215]	°C		
obd.i_press	Intake Manifold Pressure	int	[0,255]	kPa		
obd.b_press	Barometirc Pressure	int	[0,255]	kPa		
obd.f_r_press	Fuel Rail Pressure	int	[0,65530]	kPa		
obd.r_torque	Engine reference Torque	int	[0,64255]	Nm		
obd.f_torque	Engine friction Torque	float	[-125,125]	%		
obd.max_avl_torque	Engine Maximum Available Torque	float	[0,100]	%		
obd.a_torque	Engine actual Torque	float	[-125,125]	%		
obd.d_e_f_vol	Diesel Exhaust Fluid Volume	float	[0,100]	%		
obd.mf_mon	Misfire Monitor Status	int	0:not completed 1:completed			
obd.f_s_mon	Fuel System Monitor Status	int	0:not completed 1:completed			
obd.c_c_mon	Comprehensive Component Monitor Status	int	0:not completed 1:completed			
obd.c_mon	Catalyst Monitor Status	int	0:not completed 1:completed			
obd.h_c_mon	Heated Catalyst Monitor Status	int	0:not completed 1:completed			
obd.e_s_mon	Evaporative System Monitor Status	int	0:not completed 1:completed			
obd.s_a_s_mon	Secondary Air System Monitor Status	int	0:not completed 1:completed			
obd.a_s_r_mon	A/C System Refrigerant Monitor Status	int	0:not completed 1:completed			
obd.e_g_s_mon	Exhaust Gas Sensor Monitor Status	int	0:not completed 1:completed			
obd.e_g_s_h_mon	Exhaust Gas Sensor heater Monitor Status	int	0:not completed 1:completed			
obd.e_v_s_mon	EGR/VVT System Monitor Status	int	0:not completed 1:completed			

Parameter Name	Description	Type	Range	Units	Optional	Note
obd.c_s_a_s_mon	Cold Start Aid System Monitor Status	int	0:not completed 1:completed			
obd.b_p_c_s_mon	Boost Pressure Control System Monitor Status	int	0:not completed 1:completed			
obd.dpf_mon	DPF Monitor Status	int	0:not completed 1:completed			
obd.n_c_mon	NOx Catalyst Monitor Status	int	0:not completed 1:completed			
obd.nmhc_mon	NMHC Catalyst Monitor Status	int	0:not completed 1:completed			
obd.o_s_mon	Oxygen Sensor Monitor Status	int	0:not completed 1:completed			
obd.o_s_h_mon	Oxygen Sensor heater Monitor Status	int	0:not completed 1:completed			
obd.pf_mon	PF Monitor Status	int	0:not completed 1:completed			
obd.brake_prim_press	Brake Primary Pressure	float		kPa		unavailable
obd.brake_sec_press	Brake Secondary Pressure	float		kPa		unavailable

## A.5 Cellular Parameters

Parameter Name	Description	Type	Range	Units	Optional	Note
modem1.ts	The last time the modem1 info was updated	int		s		UNIX timestamp, in seconds since the epoch
modem1.active_sim	active SIM card	number	[1,2]			1: SIM1, 2: SIM2
modem1.imei	IMEI code	string				
modem1.imsi	IMSI code	string				
modem1.iccid	ICCID code	string				
modem1.phone_num	phone number	string				
modem1.signal_lvl	signal level	number		asu		
modem1.reg_status	register status	number	[0,6]			0: Not registered, ME is not currently searching an operator to register to. 1: Registered, home network. 2: Not registered, but ME is currently trying to attach or searching an operator to register to. 3: Registration denied. 4: Unknown, e.g. out of LTE coverage. 5: Registered, roaming.
modem1.operator	operator	string				
modem1.network	network type	number	[0,3]			0: NA, 1: 2G, 2: 3G, 3: 4G
modem1.lac	LAC	string				hexadecimal
modem1.cell_id	Cell ID	string				hexadecimal

Parameter Name	Description	Type	Range	Units	Optional	Note
modem1.rssi	RSSI(Received Signal Strength Indication)	number		dBm		
modem1.rsrp	RSRP(Reference Signal Receiving Power)	number		dBm		
modem1.rsrq	RSRQ(Reference Signal Receiving Quality)	number		dB		
modem1.sinr	SINR(Signal to Interference plus Noise Ratio)	number		dB		
cellular1.ts	The last time the cellular1 network info was updated	int		s		UNIX timestamp, in seconds since the epoch
cellular1.status	cellular1 network status	number	[0,3]			0: destroy 1: create 2: down 3: up
cellular1.ip	cellular1 ip address	string				
cellular1.netmask	cellular1 netmask	string				
cellular1.gateway	cellular1 gateway	string				
cellular1.dns1	cellular1 dns1	string				
cellular1.dns2	cellular1 dns2	string				
cellular1.up_at	cellular1 connected timestamp	number		s		UNIX timestamp, in seconds since the epoch
cellular1.down_at	cellular1 disconnected timestamp	number		s		UNIX timestamp, in seconds since the epoch
cellular1.traffic_ts	The last time the cellular1 traffic info was updated	int		s		
cellular1.tx_bytes	TX bytes	int		byte		
cellular1.rx_bytes	RX bytes	int		byte		

## A.6 System Parameters

Parameter Name	Description	Type	Range	Units	Optional	Note
sysinfo.ts	The last time the modem1 info was updated	int		s		UNIX timestamp, in seconds since the epoch
sysinfo.language	language	string				Chinese   English
sysinfo.hostname	hostname	string				
sysinfo.timezone	timezone	string				
sysinfo.model_name	model name	string				
sysinfo.oem_name	OEM name	string				
sysinfo.serial_number	serial number	string				
sysinfo.firmware_version	firmware version	string				
sysinfo.bootloader_version	bootloader version	string				
sysinfo.product_number	product number	string				
sysinfo.description	description	string				
sysinfo.lan_mac	MAC address of bridge1, is the same with device label	string				
sysinfo.wlan_mac	MAC address of 2G WiFi	string				
sysinfo.wlan_5g_mac	MAC address of 5G WiFi	string				

## A.7 APP Parameters

Parameter Name	Description	Type	Range	Units	Optional	Note
app.wifi_mode_2g	Notify <code>WIFIControl</code> APP to change 2.4G Wi-Fi mode	int	0:AP 1:STA -1:N/A(status only)			
app.wifi_mode_5g	Notify <code>WIFIControl</code> APP to change 5G Wi-Fi mode	int	0:AP 1:STA -1:N/A(status only)			

## A.8 1-Wire Parameters

Parameter Name	Description	Type	Range	Units	Optional	Note
1-wire.status	The connection state of 1-wire bus	string	"Connected" "Disconnected"			
1-wire.type	The type of device on 1-wire bus	string	"Temperature" "ROM Code" "Temperature & ROM Code"			
1-wire.temp_num	The number of temperature sensor devices on 1-wire bus	int	[1, 4]			
1-wire.rom_num	The number of electronic registration code devices on 1-wire bus	int	[1, 4]			
1-wire.tempN_data	The temperature value of one of the temperature sensors on 1-wire bus	float		°C		
1-wire.tempN_id	The ID of one of the temperature sensors on 1-wire bus	string				
1-wire.tempN_name	The custom name of one of the temperature sensors on 1-wire bus	string			Optional	
1-wire.rom_codeN	The value of one of the electronic registration codes on 1-wire bus	string				



Note: The letter N in a parameter like "1-wire.tempN\_data" and "1-wire.rom\_codeN" represents the number of sensors on 1-wire, on a scale of 1 to 4.