

# Описание протокола REST API

## Общие настройки

Протокол предназначен для обмена данными с устройствами и сторонними системами сбора данных, подключаемых по шине REST API. Обмен данными осуществляется в формате JSON.

### Подключение к серверу системы мониторинга по протоколу REST API

Для подключения к серверу системы мониторинга по протоколу REST API используется *http*-протокол соединения. В *http*-адресе нужно указать домен или IP-адрес сервера. Например:

```
http://[Домен или IP адрес]/api/[Параметр запроса]
```

На сервере поддерживается два типа запроса:

- GET-запрос для получения данных из системы мониторинга;
- POST-запрос для записи данных в систему мониторинга.

### Регистрация устройств и сторонних систем сбора данных по протоколу REST API

Для авторизации необходимо указать API-токен. Токен указан в настройках Пользователя. Для того, чтобы токен работал, в настройках Пользователя должен быть включен чек-бокс «Доступ к API».

Для запроса данных необходимо использовать GET-запрос с параметром «Authorization» в заголовке запроса. В адресной строке в качестве GET-параметра указать API-токен. Например:

```
http://[SERVER]/api/[Параметр_запроса]?api_token=hBI408rClzAL0PcrJ4Db
ljhHS8cVthn)
```

Для отправки данных на сервер создать POST-запрос одним из трех способов:

1. В адресной строке в качестве GET-параметра указать API-токен. Например:

```
http://[SERVER]/api/[Параметр_запроса]?api_token=hBI408rClzAL0PcrJ4Db
ljhHS8cVthn)
```

2. В теле запроса указать параметр API-токен

3. В заголовке запроса («Bearer») указать параметр «Authorization», значение «Bearer» hBI408rClizAL0PcrJ4DbljhHS8cVthn.

## Синхронизация времени сервера и источника данных

Для синхронизации времени формируется запрос метки времени, который возвращает текущее время сервера в формате JSON. Параметр GET запроса – «timestamp». Время указано в формате GMT.

Пример запроса:

```
http://[SERVER]/api/timestamp?api_token=hBI408rClizAL0PcrJ4DbljhHS8cVthn
```

Пример ответа:

```
{  
  "code": 200,  
  "status": "success",  
  "data": {  
    "timestamp": 1660039669  
  }  
}
```

## Запросы

### Запрос списка шин

Запрос списка шин формируется для отображения параметров активных шин в системе. Параметр GET запроса – «/buses».

Таблица 1.

Параметр в JSON	Поле в системе мониторинга
<i>id</i>	Уникальный номер шины в системе мониторинга
<i>code</i>	Код шины
<i>comment</i>	Комментарий
<i>tcp_remote_address</i>	IP-адрес (Конвертор, OPC Gateway Service, XNB mqtt URL)
<i>tcp_port</i>	Порт сервера
<i>bus_type</i>	Тип шины
<i>enabled/disabled</i>	Шина включена/выключена

Пример запроса:

```
http://[SERVER]/api/buses?api_token=hBI408rClzAL0PcrJ4Db1jhHS8cVthn
```

Пример ответа:

```
{
  "code": 200,
  "status": "success",
  "data": [
    {
      "id": 1,
      "code": "rest1",
      "comment": null,
      "tcp_remote_address": null,
      "tcp_port": null,
      "bus_type": "restapijson",
      "enabled": true
    }
  ]
}
```

## Запрос информации по выбранной шине

Для получения параметров шины и списка устройств необходимо сформировать GET-запрос «/buses/{buse\_code}».

Для выбора шины необходимо указать «code» (код шины) вместо переменной «{buse\_code}». Входные данные на запрос содержат информацию по запрошенной шине (см. Таблица 1) и информацию по всем устройствам, подключенными к шине (см. Таблица 2).

Таблица 2.

Параметр в JSON (массив значений: device)	Поле в системе мониторинга
<i>id</i>	Уникальный номер устройства в системе мониторинга
<i>code</i>	Код устройства
<i>device_type</i>	Тип устройства
<i>comment</i>	Комментарий
<i>active_metering_period</i>	Периодичность опроса устройства, секунд
<i>auto_metering_period</i>	Периодичность сеансов измерений, секунд
<i>correction_angle</i>	Коррекция азимута, градус
<i>correction_reverse</i>	Система координат

Пример запроса:

```
http://[SERVER]/api/buses/rest1?api_token=hBI408rClzAL0PcrJ4Db1jhHS8cVthn
```

Пример ответа:

```
{  
  "code": 200,  
  "status": "success",  
  "message": "Information on devices and buses"  
  "data": {  
    "id": 1,  
    "code": "rest1",  
    "comment": "",  
    "tcp_remote_address": "192.168.0.1",  
    "tcp_port": 0,  
    "bus_type": "restapijson",  
    "device1": {  
      "id": 1,  
      "code": "test1"  
      "device_type": "Спектроанализатор",  
      "comment": "",  
      "active_metering_period": 600,  
      "auto_metering_period": 60,  
      "correction_angle": 0,  
      "correction_reverse": 0  
    }  
  }  
}
```

### Запрос информации по выбранному устройству

Для получения параметров устройства и списка каналов необходимо сформировать GET-запрос «/sensors/{device\_code}».

Для выбора устройства необходимо указать «code» (код устройства) вместо переменной «{device\_code}». Входные данные на запрос содержат информацию пошине (см. Таблица 1), запрошенному устройству (см. Таблица 2) и информацию по всем каналам выбранного устройства (см. Таблица 3).

Таблица 3.

Параметр в JSON (массив значений: <i>channels</i> )	Поле в системе мониторинга
<i>id</i>	Уникальный номер канала в системе мониторинга
<i>sensor_id</i>	Уникальный номер устройства в системе мониторинга
<i>addr</i>	Адрес
<i>code</i>	Код канала
<i>height</i>	Высотная отметка, м

Параметр в JSON (массив значений: <i>channels</i> )	Поле в системе мониторинга
<i>correction_ratio</i>	Коэффициент преобразования
<i>correction</i>	Смещение 0

Пример запроса (запрос возвращает данные за последние 24 часа):

```
http://[SERVER]/api/sensors/test1?api_token=hBI408rClzAL0PcrJ4DbljhHS8cVthn
```

Пример ответа:

```
{
  "code": 200,
  "status": "success",
  "data": {
    "id": 1,
    "bus_id": 1,
    "code": "test1",
    "device_type": "spectroanalyzer",
    "comment": null,
    "active_metering_period": "600.00",
    "auto_metering_period": "0.0000",
    "correction_angle": "0",
    "correction_reverse": false,
    "enabled": true,
    "channels": [
      {
        "id": 1,
        "sensor_id": 1,
        "addr": "t1",
        "code": "t1",
        "height": "0",
        "correction_ratio": null,
        "correction": "0"
      },
      {
        "id": 2,
        "sensor_id": 1,
        "addr": "t2a",
        "code": "t2c",
        "height": "0",
        "correction_ratio": null,
        "correction": "0"
      }
    ],
    "bus": {
      "id": 1,
      "code": "rest1",
      "name": "test1"
    }
  }
}
```

```
        "tcp_remote_address": "62.213.109.66",
        "tcp_port": 0,
        "bus_type": "restapijson",
        "enabled": true,
        "comment": null
    }
}
}
```

## Запрос данных измерения по выбранному Каналу

Для получения информации об измерениях за выбранный период времени необходимо сформировать GET-запрос «/channels\_data/{channel\_id}».

Для выбора канала необходимо указать «id» (уникальный номер канала в Системе Мониторинга) вместо переменной «{channel\_id}». Входные данные на запрос содержат информацию данных в выбранном канале (см. Таблица 4).

Таблица 4.

Параметр в JSON	Поле в системе мониторинга
dateFrom	Дата начала запрошенного периода
dateTo	Дата окончания запрошенного периода
values→time	Время измерения
values→val	Значения измерения

Пример запроса:

```
http://[SERVER]/api/channels_data/2?api_token=hBI408rClzAL0PcrJ4DbljhHS8cVthn
```

Пример ответа:

```
{
  "code": 200,
  "status": "success",
  "data": {
    "dateFrom": "2022-08-09 20:14:35",
    "dateTo": "2022-08-10 20:14:35",
    "values": [
      {
        "time": 1660058075,
        "val": null
      },
      {
        "time": 1660058675,
        "val": null
      }
    ]
  }
}
```

```

    "time": 1660115343,
    "val": null
    "time": 1660116283,
    "val": 160
    }, {
    "time": 1660144475,
    "val": null
    }]
}
}
}

```

### Запрос статусов по объекту мониторинга

Для получения статусов по каналам (сигнализации) на объекте мониторинга необходимо сформировать GET-запрос «/objects\_status/{object\_id}».

Для выбора объекта мониторинга укажите «id» (уникальный номер объекта мониторинга в Системе) вместо переменной «{object\_id}». Входные данные на запрос содержат информацию по объекту мониторинга (см. Таблица 5).

Таблица 5.

Параметр в JSON	Поле в системе мониторинга
<i>id</i>	Уникальный номер объекта мониторинга
<i>title</i>	Название объекта
<i>interval</i>	Интервал, секунды (для внешних систем)
<i>date</i>	Дата и время запроса данных
<i>enabled_buses</i> → <i>bus_id</i>	Уникальный номер шины
<i>enabled_buses</i> → <i>name</i>	Название шины
<i>enabled_buses</i> → <i>code</i>	Код шины
<i>enabled_sensors</i> → <i>sensor_id</i>	Уникальный номер устройства
<i>enabled_sensors</i> → <i>name</i>	Название устройства
<i>enabled_sensors</i> → <i>code</i>	Код устройства
<i>channels</i> → <i>channel_id</i>	Уникальный номер канала
<i>channels</i> → <i>name</i>	Название канала
<i>channels</i> → <i>code</i>	Код канала
<i>channels</i> → <i>state</i>	Статус

Пример запроса:

```
http://[SERVER]/api/objects_status/{object_id}?api_token=hBI408rClizAL0PcrJ4DbljhHS8cVthn
```

Пример ответа:

```
{  
"code": 200,  
"status": "success",  
"data": {  
    "id": 1,  
    "title": "Название объекта",  
    "interval": 15,  
    "date": "2022-10-03 22:24:34",  
    "enabled_buses": [{  
        "bus_id": 1,  
        "name": "REST API (JSON)",  
        "code": "rest1",  
        "enabled_sensors": [{  
            "sensor_id": 1,  
            "name": "Название устройства",  
            "code": "Код устройства",  
            "channels": [{  
                "channel_id": 1,  
                "name": "Название канала",  
                "code": "Код канала",  
                "state": "gray"  
            }, {  
                "channel_id": 2,  
                "name": "Название канала",  
                "code": "Код канала",  
                "state": "gray"  
            }]  
        }, {  
            "sensor_id": 2,  
            "name": "RestAPI 2",  
            "code": "test2",  
            "channels": [{  
                "channel_id": 3,  
                "name": "T1",  
                "code": "t1",  
                "state": "red"  
            }, {  
                "channel_id": 4,  
                "name": "T2",  
                "code": "t2c",  
                "state": "green"  
            }]  
        }, ....  
    }]
```

## Запрос элементов конструкции по объекту мониторинга

Для получения информации о параметрах конструктивных элементов и сигнализации по каналам на объекте мониторинга необходимо сформировать GET-запрос «`object_elements/{object_id}`». Для выбора объекта мониторинга укажите «`id`» (уникальный номер объекта мониторинга в Системе) вместо «`{object_id}`». Назначение параметров в JSON-формате – см. Таблица 6.

Таблица 6.

Параметр в JSON	Поле в системе мониторинга
<code>location_id</code>	Уникальный номер ОМ
<code>location_title</code>	Название объекта мониторинга
<code>location_code</code>	Код объекта мониторинга
<code>elements→id</code>	ID элемента
<code>elements→bus_id</code>	Уникальный номер шины
<code>elements→_lft</code>	Параметр для построения дерева элемента конструкции
<code>elements→_rgt</code>	Параметр для построения дерева элемента конструкции
<code>elements→parent_id</code>	Параметр для построения дерева элемента конструкции
<code>elements→title</code>	Название элемента
<code>elements→type</code>	Тип элемента
<code>elements→code</code>	Код элемента
<code>elements→description</code>	Описание элемента
<code>elements→state</code>	Категория технического состояния
<code>elements→sensors</code>	Параметры устройства, привязанного к элементу
<code>elements→sensor_channels</code>	Параметры Канала, привязанного к элементу

Пример запроса:

```
http://[HOSTNAME]/api/object_elements/{object_id}?api_token=1I18lUJW0
lIiBjE0zvSeum3TSs06vcKi
```

Пример ответа:

```
{
  "code": 200,
  "status": "success",
  "data": {
    "location_id": 1,
    "location_title": "1",
    "location_code": null,
    "elements": [
      {
        "id": 9,
        "_lft": 2,
        "_rgt": 5,
        "parent_id": 1,
        "bus_id": null
      }
    ]
  }
}
```

```
"title": "L1E1",
"type": "",
"code": "l1e1",
"description": null,
"state": "",
"sensors": [
  {
    "id": 8,
    "name": "A1.1 X",
    "code": "8",
    "enabled": true,
    "channels": [
      {
        "channel_id": 26,
        "name": "1 d1",
        "code": "d1",
        "state": "gray"
      },
      {
        "channel_id": 34,
        "name": "5 f5-2",
        "code": "f5-2",
        "state": "gray"
      }
    ]
  },
  "sensor_channels": [
    {
      "channel_id": 26,
      "sensor_id": 8,
      "name": "1 d1",
      "code": "d1",
      "enabled": true,
      "state": "gray"
    }
  ]
},
{
  "id": 15,
  "_lft": 9,
  "_rgt": 10,
  "parent_id": 14,
  "title": "L1E2222",
  "type": "",
  "code": "l1e2222",
  "description": null,
  "state": "",
  "sensors": [
    {
      "id": 9,
      "name": "A1.1 Z",
      "code": "9",
      "enabled": true,
      "channels": [
        {
          "channel_id": 41,
        }
      ]
    }
  ]
}
```

```
        "name": "1  d1",
        "code": "d1",
        "state": "gray"
    }, {
        "channel_id": 42,
        "name": "1 f1",
        "code": "f1",
        "state": "gray"
    }, {
        "channel_id": 48,
        "name": "5  f5",
        "code": "f5",
        "state": "gray"
    }, {
        "channel_id": 49,
        "name": "5  f5-2",
        "code": "f5-2",
        "state": "gray"
    }]
},
"sensor_channels": []
}
}
```

## Запись данных по протоколу

Запись данных в базу данных Систему Мониторинга осуществляется при условии, что для устройства выбран тип шины «REST API (JSON)».

Для записи данных необходимо сформировать POST-запрос «/sensors\_data/{device\_code}».

Для выбора устройства укажите «code» (код устройства) вместо переменной «{device\_code}». Данные запроса для записи в базу данных содержат информацию по каналу выбранного устройства (см. Таблица 7).

### Таблица 7.

Параметр в JSON	Поле в системе мониторинга
<i>api_token</i>	Токен для доступа к API
<i>timestamp</i>	Время измерения
<i>data→addr</i>	Адрес канала
<i>data→value</i>	Значения измерения

Пример адреса для POST запроса:

[http://\[SERVER\]/api/sensors\\_data/test1](http://[SERVER]/api/sensors_data/test1)

Пример JSON-формата, где параметр data – строковое значение:

```
{  
  "api_token": "hBI408rClizAL0PcrJ4DbljhHS8cVth",  
  "timestamp": 1659696538,  
  "data": "[{\\"addr\\": \"t1\", \"value\": 12}, {\\"addr\\": \"t2a\", \"value\": 22}]"
```

## Сообщения об ошибках в протоколе

А. Ответ при отсутствии ID объекта мониторинга при запросе списка элементов:

```
{  
  "code": 404,  
  "status": "error",  
  "message": "Object with this id was not found.",  
  "data": {  
    "object_id": 15  
  }  
}
```

Б. Ответ при отключении пользователя (чек-бокс «Доступ к API» выключен у Пользователя):

```
{  
  "code": 404,  
  "status": "error",  
  "message": "Device with this code was not found.",  
  "data": {  
    "device_code": "test3"  
  }  
}
```

В. Ответ при неверном API-Токене:

```
{  
  "code": 401,  
  "status": "error",  
  "message": "Unauthenticated."  
}
```

Г. Не найдена шина в системе мониторинга:

```
{  
"code": 404,  
"status": "error",  
"message": "Bus with this code was not found.",  
"data": {  
    "bus_code": "rest2"  
}  
}
```

Д. При записи данных протокол обмена указан неверно (не «REST API (JSON)»):

```
{  
"code": 403,  
"status": "error",  
"message": "Bus type must be REST API (JSON).",  
"data": {  
    "bus_code": "rest1"  
}  
}
```

Е. Шина выключена или не создана:

```
{  
"code": 403,  
"status": "error",  
"message": "The bus is off.",  
"data": {  
    "bus_code": "rest1"  
}  
}
```

Ж. Код устройства не найден:

```
{  
"code": 404,  
"status": "error",  
"message": "Device with this code was not found.",  
"data": {  
    "device_code": "test3"  
}  
}
```

З. Устройство выключено или отсутствует в системе мониторинга:

```
{  
"code": 403,
```

```
"status": "error",
"message": "The device is off.",
"data": {
    "bus_code": "rest1",
    "sensor_code": "test1"
}
}
```

И. Ответ при отсутствии записи отдельных каналов в базе данных (при этом запись в базе данных других каналов не останавливается):

```
{
"code": 201,
"status": "success",
"data": {
    "sensor_code": "test2",
    "answer": "Skipped addresses: ['t1v']"
}
}
```