

How to use UBUS commands for Bluetooth device scanning / pairing

[Main Page](#) > [General Information](#) > [Configuration Examples](#) > [Hardware application](#) > **How to use UBUS commands for Bluetooth device scanning / pairing**

Introduction

This page contains instructions on how to use **ubus** commands in [CLI](#) / SSH to scan and pair **Bluetooth** devices on **RUTX10/11**.

The **ubus** command line tool allows to interact with the **ubusd** server (with all currently registered services). It's useful for investigating/debugging registered namespaces as well as writing shell scripts. For calling procedures with parameters and returning responses it uses the user-friendly JSON format.

Before you start using ubus commands to control Bluetooth, make sure to enable it first. You can do that in **Network** → **Bluetooth** → **General** settings. Click **Enable Bluetooth** and press **Save & Apply**.



- Now login to [CLI](#) or SSH and initialize blesemd, run **blesemd -D** command:

```
...
root@Teltonika-RUTX11:~# blesemd -D
Initializing blesemd...
Warning: debug messages will be shown!
Preparing bluetooth interface...
Reading config...
Devices loaded: 0
Creating database...
modbus_data table already exists
SENT_ID_TABLE already exists
Create DB done
Initializing uloop...
Connecting to ubus...
Initializing BLE device...
Initializing BLE ubus interface...
Adding ubus object...
...
```

- Run **scan.start** command to start Bluetooth scan:

```
...
ubus call blesem scan.start
...
```

- To see scan results, use command **scan.result**:

```
...
ubus call blesem scan.result
...
```

The scan process takes about 30 seconds. If you can see "**scanning**": **1** in the output, it means, that scanning is still in progress. After it finishes you should see a similar output:

```
...
{
  "scanning": 0,
  "devices": [
    {
      "name": "RT_T",
      "rssi": -72,
      "address": "FF:CB:FF:6F:23:FB"
    }
  ]
}
...
```

The list of devices always contains "**rssi**" and "**address**", but "**name**" output might be missing if the device is not supported:

```
...
{
  "scanning": 0,
  "devices": [
    {
      "rssi": -42,
      "address": "2F:2A:0A:0A:7A:AA"
    },
    {
      "name": "RT_T",
      "rssi": -77,
      "address": "CF:0A:52:5E:35:D7"
    }
  ]
}
...
```

- In order to pair devices use this command:

```
...
ubus call blesem pair '{"address":"FF:CC:FF:6A:23:CB"}'
...
```

If the pairing process was successful you should see this output:

```
...
{
    "success": "device successfully paired"
}
...
```

- To get the statistics from paired devices, use **stat** command:

```
...
ubus call blesem stat '{"address":"FF:CB:FA:6A:23:CB"}'
...
```

You should see similar output:

```
...
{
    "success": "successfully requested status",
    "model": "3901",
    "battery": 98,
    "temperature": "20.34",
    "humidity": 20,
    "firmware": "23",
.
}
...
```