YOSENSI.IO

Reference Manual

yosensi-cli-tool ver 3.3.0

Release notes

Date	Version	Changes
April 2021	1.0.0	Initial version
April 2021	1.1.0	1. New functionality was added (subcommand 'sensor_calibration') to calibrate sensors in YO PurePro devices. 2. Fixed optional argument '-m' and 'devParam' in list subcommand. 3. Changed the display of the parameter list of YOPurePro devices (sensor calibration settings are hidden). 4. Changed name of subcommands and help message: a. 'list_ble_char' to 'list_dev_params'; b. 'change_dev_param' to 'change_dev_parmas';
April 2021	1.1.1	1. Added support for device version LNAA-1.0.4. 2. The parameter list display for YOPurePro devices has been changed. 3. Fixed names for Seal Tracker devices (LNNA -> LNAA). 4. Fixed encoding of data (string/ASCII) to be sent via Bluetooth. 5. Fixed bug the display of eg '700002E5' as 'exp' and large value as 'inf' in subcommand 'list_dev_params'.
May 2021	1.2.0	1. Added support for device version: LNPP-1.0.2, LNPP-1.0.7. 2. Added column 'valid value' to the 'list_dev_params' subcommand. 3. Changed the way the file is checked for compatibility with the device. 4. Changed the argument name 'version_to_upload' to 'actual_version' and the help description of the 'firmware_upload' subcommand.
June 2021	1.3.0	1. Added support for: LNAN-1.0.7, LNPW-1.0.7, LNAG-1.0.8, LNFD-1.0.8, HWCM-1.0.7 and LNMP-1.0.8. 2. Added response when writing data to Bluetooth characteristics. 3. Changed message when connecting to the device. 4. Changed description of 'ntry' parameter in column 'Valid value'. 5. Changed discover and connect function to run faster. 6. Fixed display Bluetooth MAC address for bootloader. 7. Fixed subcommand 'factory_reset' - reboot device. 8. Fixed unnecessary first '0' in the parameter decimals values given with the 'change_dev_params' subcommand.
July 2021	1.3.1	1. Added support for: LNPT-1.0.8. 2. Fixed the display of parameters for LNPT devices.

August 2021	1.4.0	1. The tool has been adapted for macOS 10.15 systems. 2. Changed message to display help if no subcommand is given.
January 2022	2.0.1	1. Added new commands: a. download_dev_config; b. upload_dev_config; 2. Changed the method of configuring the devices.
April 2022	3.0.0	1. Remove deprecated functions: a. 'list_dev_params', b. 'change_dev_params', c. 'sensor_calibration'.
June 2022	3.1.0	1. Fixed bug with device discovery in macOS Monterey (12.3).
August 2022	3.2.0	1. Added new devices: a. YO Pure Lite (LNPL) b. YO CoMod test board 2. Added new parameters in YO Pure Pro: a. Vref b. tempOffset
October 2022	3.3.0	1. Added functionality to use multiple configuration files per device. 2. Added new configuration files to: a. YO Modbus b. YO Pure Pro 3.0

Contents

Reference Manual

The following reference manual describes how to use 'yosensi-cli-tool' from the command line to change settings in Yosensi devices via bluetooth communication. The 'yosensi-cli-tool' is compatible with Windows 10 and macOS 10.15 and requires bluetooth enabled (tool also works with external bluetooth modems).

The 'yosensi-cli-tool' is a console tool and should be run from the command line (it has no graphical interface). To use the tools, once you have downloaded them from our website, you must:
- update the operating system and the bluetooth modem drivers;
- enable bluetooth in your system settings;
- run the system console;
- use the system command to change the location to the folder with the downloaded tool;
- turn on the device you want to communicate with.

After changing the location to the folder with the CLI, you can start working with the tool by entering the appropriate commands. The following part of the reference manual describes supported subcommands, parameters and examples of their use.

NOTE **yosensi-cli-tool is a console tool and should be run from the command line, therefore it should not be run by double-clicking on its icon.**

Example of changing location in Windows 10 using PowerShell console (tool was downloaded and saved in the default location - 'Download').

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\user.name> cd .\Downloads\
PS C:\Users\user.name\Downloads> ls


    Directory: C:\Users\user.name\Downloads


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----          9/1/2021   7:48 AM       10177387 yosensi-cli-tool_v1.4.0_macos.tgz
-a----        12/20/2021   9:50 AM       14462127 yosensi-cli-tool_v1.4.0_win.exe
```

NOTE **PowerShell and cmd have Quick Edit mode which is a Microsoft Windows feature which allows the user to select text (click and drag the mouse across the desired text) using the mouse in the Command Prompt window. This feature freezes the console and the programs running in it until it is unfrozen, e.g. by clicking again.**
This is not a bug in the tool but a property of system terminals that can be disabled in system options.

Communication using Bluetooth technology is for sending data over short distances. Any obstacle influences the range and quality of transmission (data loss, transmission interruption). If a device is at the border of its range or broadcast frames are not coming from the device, the connection is not established, then you should change the location and try connecting to the device again.

NOTE **The 'Could not get GATT services: Unreachable' message means that CLI started to connect to the device and read GATT data (described by Bluetooth standard), but unfortunately the connection was broken. Then you should reset the device and try to connect again.**

CLI command descriptions

When working with the tool, it is important to follow the proper syntax for the commands given at the console. To use the functionality implemented in the tool, first give the name of the tool (depending on the version and the operating system) and then a subcommand specifying the action to be performed. For some subcommands, the user must define required (positional) and/or optional arguments. If the required arguments are missing, an appropriate message will be displayed.
Description of the command syntax used when working with the tool:

.\yosensi_cli_tool.exe subcommand_name positional_arg -optional_arg value ….

subcommand_name	name of the subcommand used
positional_arg	argument required, the order of the arguments is important do not change it. Following arguments are given after the 'space' without any prefixes or quotation marks
-optional_arg value	optional argument, may or may not be provided. It is preceded by a prefix, e.g. '-r' means to pass a new RSSI level value.

Display of all supported subcommands by the tool and the tool version:

```
.\yosensi-cli-tool_v3.0.0_WIN.exe --help
usage: yosensi-cli-tool [-h] [-v]

{list,list_dev_params,change_dev_params,factory_reset,firmware_upload,sensor_calibration}
                        ...

optional arguments:
  -h, --help            show this help message and exit
  -v, --version         show program's version number and exit

Subcommands list:
  {list,list_dev_params,change_dev_params,factory_reset,firmware_upload,sensor_calibration}
    list                list node(s)/gateway(s) by name or bluetooth MAC
    factory_reset       restore device to factory settings
    firmware_upload     uploading device firmware from a file
```

```
    download_dev_config download and save node configuration data to file
    upload_dev_config   load and upload node configuration data from file


.\yosensi-cli-tool_v3.0.0_WIN.exe --version
3.0.0
```

list subcommand

This subcommand allows you to look for all Yosensi devices which are in Bluetooth range. After the search is complete, a list of found devices with their name, RSSI level and public MAC address will be displayed. The public MAC address from the table will be needed when using next subcommands and it's the same as on the 'macBLE' nameplate.

```
.\yosensi-cli-tool_v1.4.0_WIN.exe list --help
usage: yosensi-cli-tool list [-h] [-m [MAC]] [-n [NAME]]

optional arguments:
  -h, --help  show this help message and exit
  -m [MAC]    Bluetooth public MAC address
  -n [NAME]   node/gateway name
```

OPTIONAL ARGUMENTS

-h	display help
-m	bluetooth public MAC address of the device you are looking for. The MAC address can be written with a colon (:) and without e.g. 80:e1:26:1d:2a:33 or 80e1261d2a33
-n	the name of the device to be found. This subcommand allows you to look for a specific group of devices by their name, e.g. LNPP-1.0.5

Examples of use:

```
.\yosensi-cli-tool_v1.4.0_WIN.exe list
  Discovery Bluetooth devices takes 30s, please wait ...
  +------------+------+--------------------+
  |    Name    | RSSI | Public MAC Address |
  +============+======+====================+
  | LNPC013200 | -55  | 80:e1:26:07:b5:bd  |
  +------------+------+--------------------+
  | LNTP013200 | -78  | 80:e1:26:1a:39:93  |
  +------------+------+--------------------+
  | LNPW-1.0.5 | -68  | 80:e1:26:1d:2a:33  |
  +------------+------+--------------------+
  | LNAP-1.0.6 | -58  | 80:e1:26:1d:2b:07  |
  +------------+------+--------------------+
  | LNPW-1.0.6 | -63  | 80:e1:26:1d:2a:df  |
  +------------+------+--------------------+
```

```
.\yosensi-cli-tool_v1.4.0_WIN.exe list -n LNPW
  Discovery Bluetooth devices takes 30s, please wait ...
  +------------+------+--------------------+
  |    Name    | RSSI | Public MAC Address |
  +============+======+====================+
  | LNPW-1.0.6 | -73  | 80:e1:26:1d:2a:df  |
  +------------+------+--------------------+
  | LNPW-1.0.5 | -57  | 80:e1:26:1d:2a:33  |
  +------------+------+--------------------+

.\yosensi-cli-tool_v1.4.0_WIN.exe list -m 80:e1:26:1d:2a:33
  Discovery Bluetooth devices takes 30s, please wait ...
  +------------+------+--------------------+
  |    Name    | RSSI | Public MAC Address |
  +============+======+====================+
  | LNPW-1.0.5 | -80  | 80:e1:26:1d:2a:33  |
  +------------+------+--------------------+
```

factory_reset subcommand

The subcommand is used to restore the factory settings. All individual user settings will be permanently deleted and overwritten with default values.

```
.\yosensi-cli-tool_v1.4.0_WIN.exe factory_reset --help
usage: yosensi-cli-tool factory_reset [-h] mac

positional arguments:
  mac         Bluetooth public MAC address

optional arguments:
  -h, --help  show this help message and exit
```

REQUIRED ARGUMENTS

mac	bluetooth public MAC address of the device you want to restore to factory settings; the MAC address can be written with a colon (:) and without e.g. 80:e1:26:1d:2a:33 or 80e1261d2a33

Example of use:

```
.\yosensi-cli-tool_v1.4.0_WIN.exe factory_reset 80:e1:26:1d:2a:33
It will take up to 120s to find and restore to factory settings the device, please wait ...
Searching for a device ...
Trying to connect to the device LNPP-1.0.5...
Factory reset: SUCCESS
```

firmware_upload subcommand

Command is used to upload firmware to the device. This process is done in two steps. In the first step, CLI makes three attempts to connect to the device and launch the bootloader (which removes the current firmware to free up memory). If the three attempts fail, uploading firmware on that device will not begin. In the second step, the program is uploaded via bluetooth. This process takes some time. During the upload process progress is displayed in the console. CLI makes three attempts to upload the firmware, if all fails the device will stay in bootloader mode. Then you have to start the process from the beginning.

NOTE **The device should not be disconnected from supply, reset or moved.**

```
.\yosensi-cli-tool_v1.4.0_WIN.exe firmware_upload --help
usage: yosensi-cli-tool firmware_upload [-h] [-r [RSSI]]
                                        version_to_upload file mac_list
                                        [mac_list ...]

positional arguments:
  actual_version     device(s) with this name and version will have new firmware
                     loaded
  file               firmware file name
  mac_list           Bluetooth public MAC address list

optional arguments:
  -h, --help         show this help message and exit
  -r [RSSI]          acceptable RSSI level for upload, default value -75 [dBm]
```

REQUIRED ARGUMENTS

actual_version	device(s) with this name and version will have new firmware loaded; the name of the device, e.g. LNPP-1.0.5, contains the model name and firmware version
NOTE	**It is necessary to check the compatibility of the updated device with the uploaded file.**
file	the name of the firmware file to be uploaded to the device. The latest firmware is provided on Yosensi.io in the support section.
mac	in order to initialize the upload process, you need to specify a list of public MAC addresses of the devices to be updated or enter the keyword 'all'. If 'all' is entered, all devices whose names match the 'version_to_upload' argument will take in the update process. Accepted MAC address format: '80:e1:26:1d:2a:33' or '80e1261d2a33'

OPTIONAL ARGUMENTS

-r	acceptable RSSI level required to start the firmware upload. The default value is set to -75 dBm. **It is not recommended to set a low RSSI value because it may have a negative impact on the firmware upload process**
-h	display help

Example of use:

```
.\yosensi-cli-tool_v1.4.0_WIN.exe firmware_upload LNPW-1.0.5 .\LNPW-1.0.6_EU868.bin 80e1261d2a33
Searching for a device ...
Found 1 reachable LoRa devices:
+------------+------+--------------------+
|    Name    | RSSI | Public MAC Address |
+============+======+====================+
| LNPW-1.0.5 | -58  | 80:e1:26:1d:2a:33  |
+------------+------+--------------------+

Applying new firmware to reachable devices (RSSI > -75 [dBm])...
        1. macBLE: 80:e1:26:1d:2a:33
                Erase memory: attempt 1/3
                        SUCCESS
                Uploading new firmware: attempt 1/3
                        Firmware size: 233692
                        Numbers of packages to send via ble: 11685
                        Package 11685 of 11685
                        Send all packages
                        SUCCESS
                SUCCESS applying firmware '.\LNPW-1.0.6_EU868.bin' on device
                        macBLe: 80:e1:26:1d:2a:da
                        devEUI: 544e500d00400018

.\yosensi-cli-tool_v1.4.0_WIN.exe firmware_upload LNPW-1.0.5 .\LNPW-1.0.6_EU868.bin all
Searching for a device ...
Found 2 reachable LoRa devices:
+------------+------+--------------------+
|    Name    | RSSI | Public MAC Address |
+============+======+====================+
| LNPW-1.0.5 | -37  | 80:e1:26:1d:2a:da  |
+------------+------+--------------------+
| LNPW-1.0.5 | -47  | 80:e1:26:1d:2a:33  |
+------------+------+--------------------+

Applying new firmware to reachable devices (RSSI > -75 [dBm])...
        1. macBLE: 80:e1:26:1d:2a:da
                Erase memory: attempt 1/3
                        SUCCESS
                Uploading new firmware: attempt 1/3
                        Firmware size: 233692
                        Numbers of packages to send via ble: 11685
                        Package 11685 of 11685
                        Send all packages
                        SUCCESS
                SUCCESS applying firmware '.\LNPW-1.0.6_EU868.bin' on device
                        macBLe: 80:e1:26:1d:2a:da
                        devEUI: 0123456789abcdef
```

```
        2. macBLE: 80:e1:26:1d:2a:33
                Erase memory: attempt 1/3
                        SUCCESS
                Uploading new firmware: attempt 1/3
                        Firmware size: 233692
                        Numbers of packages to send via ble: 11685
                        Package 11685 of 11685
                        Send all packages
                        SUCCESS
                SUCCESS applying firmware '.\LNPW-1.0.6_EU868.bin' on device
                        macBLe: 80:e1:26:1d:2a:33
                        devEUI: 0123456789abcdef

.\yosensi-cli-tool_v1.4.0_WIN.exe firmware_upload LNPW-1.0.5 .\LNPW-1.0.6_EU868.bin
80:e1:26:1d:2a:da 80:e1:26:1d:2a:33
Searching for a device ...
Found 2 reachable LoRa devices:
+------------+------+--------------------+
|    Name    | RSSI | Public MAC Address |
+============+======+====================+
| LNPW-1.0.5 | -39  | 80:e1:26:1d:2a:da  |
+------------+------+--------------------+
| LNPW-1.0.5 | -51  | 80:e1:26:1d:2a:33  |
+------------+------+--------------------+

Applying new firmware to reachable devices (RSSI > -75 [dBm])...
        1. macBLE: 80:e1:26:1d:2a:da
(...)
```

Firmware update process

To better explain the firmware update process, let's assume that we have a 'YOPower' device that currently has firmware version 'LNPW-1.0.5' and we want to update it to version 'LNPW-1.0.6'.

1. Download 'yosensi-cli-tool' for Windows 10
2. Download the latest firmware version for your device model 'LNPW-1.0.6_EU868'.
3. Put the downloaded firmware into the CLI tool folder.

```
PS C:\yo-CLI\yosensi-cli-tool\dist> ls


    Directory: C:\yo-CLI\yosensi-cli-tool\dist


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         3/29/2021   1:12 PM         233692 LNPW-1.0.6_EU868.bin
-a----         3/30/2021   3:38 PM       11598807 yosensi-cli-tool_v3.0.0_WIN.exe
```

4. Open a system console such as cmd or PowerShell and then go to the folder with the saved CLI tool.
5. Update devices must be turned on and be within Bluetooth range.
6. Download actual configuration to file (backup of configurations) and rename the file so that it will not be overwritten in the next steps:

```
.\yosensi-cli-tool_v3.0.0_WIN.exe download_dev_config 80e1261d2a33
```

7. Enter the subcommand to upload the firmware and specify its arguments:

```
.\yosensi-cli-tool_v3.0.0_WIN.exe firmware_upload LNPW-1.0.5 .\LNPW-1.0.6_EU868.bin 80e1261d2a33
```

- actual_version: **LNPW-1.0.5**
- file: **.\LNPW-1.0.6_EU868.bin**
- mac: **80e1261d2a33**

8. After the subcommand is run, the CLI tool will start to look for devices with the specified parameters and then start the upload process. The process steps are displayed in the console.
9. Verify the status of the software transfer process.
   9.1. If everything was successful, the device will automatically run with the new firmware.

```
.\yosensi-cli-tool_v3.0.0_WIN.exe firmware_upload LNPW-1.0.5 .\LNPW-1.0.6_EU868.bin 80e1261d2a33
Searching for a device ...
Found 1 reachable LoRa devices:
+------------+------+--------------------+
|    Name    | RSSI | Public MAC Address |
+============+======+====================+
| LNPW-1.0.5 | -47  | 80:e1:26:1d:2a:33  |
+------------+------+--------------------+

Applying new firmware to reachable devices (RSSI > -75 [dBm])...
        1. macBLE: 80:e1:26:1d:2a:33
                Erase memory: attempt 1/3
                        SUCCESS
                Uploading new firmware: attempt 1/3
                        Firmware size: 233692
                        Numbers of packages to send via ble: 11685
                        Package 11685 of 11685
                        Send all packages
                        SUCCESS
                SUCCESS applying firmware '.\LNPW-1.0.6_EU868.bin' on device
                        macBLe: 80:e1:26:1d:2a:33
                        devEUI: 544e500d003d0026
```

In our case CLI in the first attempt managed to bootloader and clear the memory and then in the first attempt started the upload process, which was successful. In addition, the data of the device being updated is displayed.

   9.2. If the bootloader is successfully started and the memory is erased, but all attempts to upload the firmware fail, the device will stay running in bootloader mode.

```
.\yosensi-cli-tool_v1.4.0_WIN.exe firmware_upload LNPW-1.0.5 .\LNPW-1.0.6_EU868.bin 80e1261d2a33
Searching for a device ...
Found 1 reachable LoRa devices:
+------------+------+--------------------+
|    Name    | RSSI | Public MAC Address |
+============+======+====================+
| LNPW-1.0.5 | -58  | 80:e1:26:1d:2a:33  |
+------------+------+--------------------+

Applying new firmware to reachable devices (RSSI > -75 [dBm])...
        1. macBLE: 80:e1:26:1d:2a:33
                Erase memory: attempt 1/3
                        SUCCESS
                Uploading new firmware: attempt 1/3
                        Firmware size: 233692
                        Numbers of packages to send via ble: 11685
                        Package 11685 of 11685
                        Send all packages
                        FAIL
                Uploading new firmware: attempt 2/3
                Uploading new firmware: attempt 3/3
                FAILED applying firmware '.\LNPW-1.0.6_EU868.bin' on device
                        macBLe: 80:e1:26:1d:2a:33
                        devEUI: 544e500d003d0026
```

Then you have to change the subcommand arguments and make another attempt to upload the firmware.

```
.\yosensi-cli-tool_v1.4.0_WIN.exe firmware_upload STM_OTA .\LNPW-1.0.6_EU868.bin 80e1261d2a33
```

- actual_ version: **STM_OTA**
- file: **.\LNPW-1.0.6_EU868.bin**
- mac: **80e1261d2a33**

In this case, we have specified that we want to upload the program 'LNPW-1.0.6' to the device with MAC address '80e1261d2a33', which is in bootloader mode.

NOTE **In this case, there is no way to verify the compatibility of the device with the uploaded firmware.**

```
.\yosensi-cli-tool_v1.3.0_WIN.exe firmware_upload STM_OTA .\LNPW-1.0.6_EU868.bin 80e1261d2a33
*** WARNING ***
You are trying to upload the firmware 'LNPW-1.0.6_EU868.bin' to a device with a bootloader.
In this case, checking compatibility is not possible. Please be careful what you do.
Are you sure? (y/n)y
Searching for a device ...
Found 1 reachable LoRa devices:
+---------+------+--------------------+
|  Name   | RSSI | Public MAC Address |
+=========+======+====================+
| STM_OTA | -52  | 80:e1:26:1d:2a:33  |
+---------+------+--------------------+
```

```
Applying new firmware to reachable devices (RSSI > -75 [dBm])...
        1. macBLE: 80:e1:26:1d:2a:34
                Uploading new firmware: attempt 1/3
                        Firmware size: 233692
                        Numbers of packages to send via ble: 11685
                        Package 11685 of 11685
                        Send all packages
                        SUCCESS
                SUCCESS applying firmware '.\LNPW-1.0.6_EU868.bin' on device
                        macBLe: 80:e1:26:1d:2a:34
                        devEUI: UNKNOWN
```

The upload process was successful. In this case, the summary does not include the 'devEUI' of the device.

10. After performing a firmware update, try to download the configuration. The firmware upload process in some cases may cause some settings to be lost/overwritten. **Remember not to overwrite the configuration file you downloaded in step 6 - you should rename it.**

```
.\yosensi-cli-tool_v3.0.0_WIN.exe download_dev_config 80e1261d2a33
```

10.1. If the configuration file downloads without a problem, the CLI does not display any error, the file is successfully downloaded and saved. Verify the correctness of the settings with the backup - overwrite the settings if necessary. If it was necessary to overwrite settings after the firmware upload process, upload restored settings to the device - subcommand "*upload_dev_config*".

10.2. If, after the firmware update process, the CLI is unable to download the configuration file, it displays the following message (example of a message):

```
.\yosensi-cli-tool_v3.0.0_win.exe download_dev_config 80e1261d2a33
It will take up to 120s to find and start downloading configuration data from the device to file,
please wait ...
Searching for a device ...
Trying to connect to the device LNPW-3.0.0 ...
Downloading the configuration file ...
100%|████████████████████████████████████████████████████████████████████████████| 3/3
[00:00<00:00, 13.24PKG/s]
Downloading the configuration file is complete.
Error parsing message with type 'PBConfig'
```

Please restore the factory settings with the command "*factory_reset*" and try to download the configuration file again. The downloaded configuration file will then contain the default settings. Using the backup of the settings made in step 6, restore your settings. When you have finished restoring your settings, save the file and use the subcommand "*upload_dev_config*" to upload the new configuration to the device.

download_dev_config subcommand

NOTE **Subcommand to download the configuration has been added to devices with firmware version 2.0.0 or newer. For devices with older firmware versions use CLI version 1.4.0.**

This subcommand is used to download the device configuration and save it to a file. The device configuration is saved in JSON format in a folder from the CLI. The file name consists of 'device_config', the model name e.g. LNFN and bluetooth public MAC address of the device (without a colon in address) to which the configuration relates e.g. device_config_LNFM_80e1261cf9e9.json.

```
.\yosensi-cli-tool_v2.0.1_WIN.exe download_dev_config --help
usage: yosensi-cli-tool download_dev_config [-h] mac

positional arguments:
  mac         Bluetooth public MAC address

optional arguments:
  -h, --help  show this help message and exit
```

REQUIRED ARGUMENTS

mac	bluetooth public MAC address of the device you want to restore to factory settings; the MAC address can be written with a colon (:) and without e.g. 80:e1:26:1d:2a:33 or 80e1261d2a33

Example of use:

```
PS C:\Users\user.name\Desktop\CLI> ls


    Directory: C:\Users\user.name\Desktop\CLI


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         1/14/2022  12:43 PM       12860818 yosensi-cli-tool_v2.0.1_WIN.exe


PS C:\Users\user.name\Desktop\CLI> .\yosensi-cli-tool_v2.0.1_WIN.exe download_dev_config
80:e1:26:1c:f9:e9
It will take up to 120s to find and start downloading configuration data from the device to file,
please wait ...
Searching for a device ...
Trying to connect to the device LNFM-2.1.0 ...
Downloading the configuration file ...
100%|████████████████████████████████████████████████████████████████| 14/14
[00:01<00:00,  8.44PKG/s]
Downloading the configuration file is complete.
Checksum of the downloaded configuration data is correct.
```

```
The configuration data has been written to the file.
C:\Users\user.name\Desktop\CLI\device_config_LNFM_80e1261cf9e9.json

PS C:\Users\user.name\Desktop\CLI> ls


    Directory: C:\Users\user.name\Desktop\CLI


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         1/20/2022   2:04 PM            653 device_config_LNFM_80e1261cf9e9.json
-a----         1/14/2022  12:43 PM       12860818 yosensi-cli-tool_v2.0.1_WIN.exe
```

Example preview of downloaded configuration file - device_config_LNFM_80e1261cf9e9.json:


```
{
	"info": {
		"devmodel": "LNRM",
		"fwver": "2.1.0",
		"loraradio": "SX1261",
		"lorawanver": "1.0.2",
		"loraregion": "EU868",
		"blemacaddr": "80e1261cf9e9"
	},
	"lora": {
		"nwktype": "private",
		"acttype": "abp",
		"otaa": {
			"deveui": "0123456789abcdef",
			"appeui": "fedcba9876543210",
			"appkey": "000102030405060708090a0b0c0d0e0f",
			"trials": 1
		},
		"abp": {
			"devaddr": "01234567",
			"nwkskey": "0123456789abcdef0123456789abcdef",
			"appskey": "00112233445566778899aabbccddeeff"
		}
	},
	"ble": {
		"power": 0,
		"interval": 1600
	},
	"device": {
		"measinterval": 7200,
		"pressurerange": 50
	}
}
```


The device configuration is split into sections:

- **info** (generic section): information about the device (**read only**), i.e. device version, model, etc.
- **lora** (generic section): configuration data required to connect to the LoRaWAN, i.e. activation type, keys, etc.
- **ble** (generic section): bluetooth settings
- **device** (dynamic section): individual configuration for a specific device (the structure of this section is different for each device).

NOTE **Sections "info", "lora" and "ble" are common for all devices and have the same structure, parameters and names.**
Section "device" is dynamic - its structure, parameters and names depend on the device.

NOTE **The structure of the json file or the names of the parameters may change as the firmware version changes.**

upload_dev_config subcommand

NOTE **Subcommand to download the configuration has been added to devices with firmware version 2.0.0 or newer. For devices with older firmware versions use CLI version 1.4.0.**

This subcommand is used to upload the configuration data saved in the JSON file to the device. When running this subcommand you only need to enter the bluetooth public MAC address of the device (without a colon in address) you want to configure, then after finding the device the tool reads its model and then attempts to load the file dedicated for this device. If the tool doesn't find the file (the name is incompatible with the standard) or if the file format is incompatible with the JSON standard, the tool will stop working by displaying an error message. The configuration file will be located in the CLI tool folder and its name must be in accordance with the standard and consist of: 'device_config', the name of the model to which the configuration relates, bluetooth public MAC address of the device (without a colon in address) to which the configuration relates e.g. device_config_LNFM_80e1261cf9e9.json.
To create the configuration file with the specified JSON file structure and name it is recommended to first download the configuration file with the subcommand "download_dev_config". The tool will download the current configuration and save it into a JSON file with proper name. Then we are sure that the structure and file name are correct and we can start changing the parameters before sending them to the device. After the configuration file has been uploaded successfully, the device will be rebooted and it will start with the new settings that have been entered.

```
.\yosensi-cli-tool_v2.0.1_WIN.exe upload_dev_config --help
usage: yosensi-cli-tool upload_dev_config [-h] mac

positional arguments:
  mac         Bluetooth public MAC address

optional arguments:
```

```
  -h, --help  show this help message and exit
```

REQUIRED ARGUMENTS

mac	bluetooth public MAC address of the device you want to restore to factory settings; the MAC address can be written with a colon (:) and without e.g. 80:e1:26:1d:2a:33 or 80e1261d2a33

Example of use:

```
PS C:\Users\user.name\Desktop\CLI> ls


    Directory: C:\Users\user.name\Desktop\CLI


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         1/21/2022   9:41 AM            653 device_config_LNFM_80e1261cf9e9.json
-a----         1/14/2022  12:43 PM       12860818 yosensi-cli-tool_v2.0.1_WIN.exe


PS C:\Users\user.name\Desktop\CLI> .\yosensi-cli-tool_v2.0.1_WIN.exe upload_dev_config
80:e1:26:1c:f9:e9
It will take up to 120s to find and start uploading configuration data to the device from file,
please wait ...
Searching for a device ...
Trying to connect to the device LNFM-2.1.0 ...
Uploading the configuration file ...
100%|████████████████████████████████████████████████████████████████████████████| 14/14
[00:01<00:00,  8.44PKG/s]
Uploading the configuration file is complete.
```

Example of the tool's response if the JSON file structure is incorrect - the name of the "devmodel" parameter has been written without double quotes marks:

```
{
        "info": {
                devmodel: "LNFM",
                "fwver": "2.1.0",
                "loraradio": "SX1261",
                "lorawanver": "1.0.2",
                "loraregion": "EU868",
                "blemacaddr": "80e1261cf9e9"
        },
        ...
}
```

```
PS C:\Users\user.name\Desktop\CLI> .\yosensi-cli-tool_v2.0.1_WIN.exe upload_dev_config
80:e1:26:1c:f9:e9
It will take up to 120s to find and start uploading configuration data to the device from file,
please wait ...
```

```
Searching for a device ...
Syntax error in the configuration file. Expecting property name enclosed in double quotes: line 3
column 3 (char 15)
```

The error message description specifies the location of the error in the data structure of the file. You can use any JSON file editor or use online editors to validate the JSON file structure.

Example of the tool's response if the configuration file does not exist or its name is not compliant with the standard (in this case the MAC address in the filename is wrong):

```
PS C:\Users\user.name\Desktop\CLI> ls


    Directory: C:\Users\user.name\Desktop\CLI


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         1/21/2022   9:41 AM            653 device_config_LNFM_80e1261cf9.json
-a----         1/14/2022  12:43 PM       12860818 yosensi-cli-tool_v2.0.1_WIN.exe

PS C:\Users\user.name\Desktop\CLI> .\yosensi-cli-tool_v2.0.1_WIN.exe upload_dev_config
80:e1:26:1c:f9:e9
It will take up to 120s to find and start uploading configuration data to the device from file,
please wait ...
Searching for a device ...
The configuration file does not exist:
C:\Users\user.name\Desktop\CLI\device_config_LNFM_80e1261cf9e9.json
```

Extending the configuration file by adding a new section or parameter will not cause an upload error. However, this data will not be uploaded to the device, it will be omitted during compression to optimize the data upload size. For example, the parameter "networkname" was added in the "lora" section, it will be omitted when compressing data - unknown parameter. The JSON file structure is preserved.

```
{
        "info": {
                ...
        },
        "lora": {
                "networkname": "1st floor",
                "nwktype": "private",
                "acttype": "abp",
                ...
        },
        ...
}
```

It is allowed to upload only the parameters that you want to change without uploading the whole configuration file. let's assume that you only want to change the value of the "devaddr" parameter to the new one, then the configuration file will look like this:

```
{
        "lora":{
                "abp":{
                        "devaddr": "01234AFF"
                }
        }
}
```

```
PS C:\Users\user.name\Desktop\CLI> .\yosensi-cli-tool_v2.0.1_WIN.exe upload_dev_config
80:e1:26:1c:f9:e9
It will take up to 120s to find and start uploading configuration data to the device from file,
please wait ...
Searching for a device ...
Trying to connect to the device LNFM-2.1.0 ...
Uploading the configuration file ...
100%|████████████████████████████████████████████████████████████| 3/3
[00:03<00:00,  1.01s/PKG]
Uploading the configuration file is complete.
```

As you can see, the amount of data uploaded is much less. **Please note that in this case the correct data structure must be preserved - placing the parameters in sections.**

Change of device configuration process

NOTE **For devices with firmware version 2.0.0 and newer this is the recommended method for changing device settings.**

To better explain the process of uploading a new configuration, let's assume that we want to change the Refrigerant Monitor 'LNFM-2.1.0' version 2.1.0 in the device:
- device address "devaddr": *012345*FF
- network type "nwktype": public
- measurement interval "measinterval": 3600 (measurement every 3600 seconds)

1. Download the CLI tool 'yosensi-cli-tool' on your operating system.
2. Download the current configuration from the device.

```
PS C:\Users\user.name\Desktop\CLI> .\yosensi-cli-tool_v2.0.1_WIN.exe download_dev_config
80:e1:26:1c:f9:e9
It will take up to 120s to find and start downloading configuration data from the device to file,
please wait ...
Searching for a device ...
Trying to connect to the device LNFM-2.1.0 ...
Downloading the configuration file ...
100%|████████████████████████████████████████████████████████████| 14/14
[00:01<00:00,  8.44PKG/s]
Downloading the configuration file is complete.
Checksum of the downloaded configuration data is correct.
The configuration data has been written to the file.
C:\Users\user.name\Desktop\CLI\device_config_LNFM_80e1261cf9e9.json
```

3. Open the downloaded configuration file in a JSON file editor.


```
{
        "info": {
                "devmodel": "LNFM",
                "fwver": "2.1.0",
                "loraradio": "SX1261",
                "lorawanver": "1.0.2",
                "loraregion": "EU868",
                "blemacaddr": "80e1261cf9e9"
        },
        "lora": {
                "nwktype": "private",
                "acttype": "abp",
                "otaa": {
                        "deveui": "0123456789abcdef",
                        "appeui": "fedcba9876543210",
                        "appkey": "000102030405060708090a0b0c0d0e0f",
                        "trials": 1
                },
                "abp": {
                        "devaddr": "01234567",
                        "nwkskey": "0123456789abcdef0123456789abcdef",
                        "appskey": "00112233445566778899aabbccddeeff"
                }
        },
        "ble": {
                "power": 0,
                "interval": 1600
        },
        "device": {
                "measinterval": 7200,
                "pressurerange": 50
        }
}
```


4. Make changes to the required parameters in the editor.


```
{
        "info": {
                "devmodel": "LNFM",
                "fwver": "2.1.0",
                "loraradio": "SX1261",
                "lorawanver": "1.0.2",
                "loraregion": "EU868",
                "blemacaddr": "80e1261cf9e9"
        },
```

```
        "lora": {
                "nwktype": "public",
                "acttype": "abp",
                "otaa": {
                        "deveui": "0123456789abcdef",
                        "appeui": "fedcba9876543210",
                        "appkey": "000102030405060708090a0b0c0d0e0f",
                        "trials": 1
                },
                "abp": {
                        "devaddr": "012345FF",
                        "nwkskey": "0123456789abcdef0123456789abcdef",
                        "appskey": "00112233445566778899aabbccddeeff"
                }
        },
        "ble": {
                "power": 0,
                "interval": 1600
        },
        "device": {
                "measinterval": 3600,
                "pressurerange": 50
        }
}
```

5. Save the changes to the file, remember not to change the file name.
6. Upload the configuration file to the device.

```
PS C:\Users\user.name\Desktop\CLI> .\yosensi-cli-tool_v2.0.1_WIN.exe upload_dev_config
80:e1:26:1c:f9:e9
It will take up to 120s to find and start uploading configuration data to the device from file,
please wait ...
Searching for a device ...
Trying to connect to the device LNFM-2.1.0 ...
Uploading the configuration file ...
100%|████████████████████████████████████████████████████████████| 14/14
[00:01<00:00,  8.44PKG/s]
Uploading the configuration file is complete.
```

The device will reset itself after uploading the configuration.

7. You can check if the configuration has been correctly entered into the memory by downloading and analyzing the current configuration.

Revision history

Date	Version	Changes
01.04.2021	1.0.0	Initial version.
08.04.2021	1.1.0	Added 'sensor_calibration' section. Changed section name and subcommand name from 'list_ble_char' to 'list_dev_params' and from 'change_dev_param' to 'change_dev_parmas'.
27.05.2021	1.2.0	Changed description of 'list_dev_params' section. Changed parameter name from 'version_to_upload' to 'actual_version' in section 'firmware_upload subcommand'.
30.06.2021	1.3.0	Changed version 1.2.0 to 1.3.0. Changed message when connecting to a device (added device names). Changed description of parameter 'ntry' in column 'valid value' of subcommand 'list_dev_params' to 'Reserved'. Changed in description of 'firmware_upload' subcommand: listing of an example of how to use the subcommand, the word throughout the listing 'block' to 'package', and the MAC address of the STM_OTA.
07.07.2021	1.3.1	Changed version 1.3.0 to 1.3.1.
06.08.2021	1.4.0	Changed version 1.3.1 to 1.4.0. Added "macOS 10.15" to the “Reference Manual” description.
12.21.2021	1.4.1	Changed the description of the "Reference Manual". Changed the description of the “CLI command descriptions” Changed the name of the parameter from *“value_of_positional_arg”* to *“positional_arg”* in section CLI command descriptions.
21.01.2022	2.0.1	Added 'download_dev_config' section. Added the 'upload_dev_config' section. Added a new general parameter - LoraWAN network type. Added new devices - LNFM. Changed “CLI command descriptions” section - list of supported subcommands.

15.04.2022	3.0.0	Removed sections: - list_dev_params subcommand; - change_dev_params subcommand; - sensor_calibration subcommand. Changed the description of the “firmware update process”.