



Application Note – Python programming tool usage for software upgrade

Table of Contents

1. Introduction.....	2
2. Package	2
3. Requirements	3
4. Programming configuration	3
4.1. Configuration	3
4.2. CAN communication.....	3
4.3. Nodes configuration	4
5. Programming procedure	5

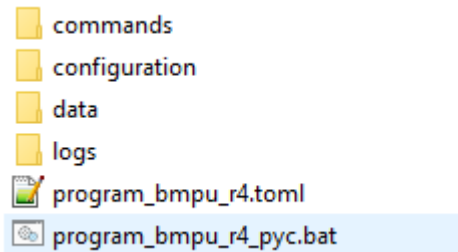


1. Introduction

This document aims to describe how to use the programming python script to update the software of Watt and Well products on a CAN bus.

2. Package

The programming package consists of 4 folders and two files:



- **commands folder** contains the compiled python scripts (.pyc)
- **configuration folder** contains configuration files that must not be modified. For instance, it contains the "log_conf.toml" that configure the logs formatting.
- **data folder** contains the data of each available software version. This folder is divided into subfolders. Each containing a specific software version
 - **eds files** describe the CANopen dictionary of a specific software version and are used by python scripts.
 - **wtcfw files** are the software binary to upload into the devices.
- **log folder** is used to store the command execution logs.
- **program_bmpu-rX.toml** is a editable file which describes the software configuration that the python script must apply
- **program_bmpu-rX _pyc.bat** contains a batch script that calls the programming script in a windows shell with program_mpu.toml as argument.

3. Requirements

1. Windows OS
2. Python 3.7.9 must be installed and be your main python interpreter.
 - a. See <https://www.python.org/downloads/>
3. CAN Transceiver kvaser with Canlib drivers installed.
 - a. See <https://www.kvaser.com/download/>

4. Programming configuration

The programming configuration to apply is defined by “**program_bmpu.toml**” file

```
configuration = "single_node"

[can]
bustype = "kvaser"
channel = 0
baudrate = 500_000

[[nodes_config]]
node_name = "bmpu1"
node_type = "bmpu"
node_id = 80
sw_version = "2.4.4r"
sw_build = 17760
program = true
from_asw = true
```

This file is constituted of 3 sections:

- configuration
- can
- nodes_config

4.1. Configuration

This parameter describes the CAN configuration. It is set to ‘single_node’ to program a BMPU.

```
configuration = "single_node"
```

4.2. CAN communication

This structure defines how to access to the can bus. The following configuration is mandatory to program a MPU with a kvaser transceiver:

Parameter	Value	description
bustype	kvaser	Type of connection or transceiver used to access to the CAN bus
channel	0	Channel used to access to the CAN bus
Baudrate	500000	Baudrate of the CAN bus

```
[can]
bustype = "kvaser"
channel = 0
baudrate = 500_000
```

4.3. Nodes configuration

The nodes_config structure describes the software configuration to apply for each node.

The example given below describes how to configure the script to program the version 2.4.4r build 17760 to B MPU-R2 with CANopen node ID of 80.

```
configuration = "single_node"

[can]
bustype = "kvaser"
channel = 0
baudrate = 500_000

[[nodes_config]]
node_name = "bmpu1"
node_type = "bmpu"
node_id = 80
sw_version = "2.4.4r"
sw_build = 17760
program = true
from_asw = true
```

Parameter	Value	description
node_name	BMPUX	Name of the node to program
node_type	B MPU	Type of the node to program
node_id	80	CAN node id of the target B MPU node
sw_version	2.x.y	Software version to program
sw_build	xxxxx	Build number of the software version
program	true or false	Boolean defining if the node is to be programmed

from_asw	true	Boolean defining if the programming is to be done from application software (asw) or bootloader
-----------------	------	---

Only 'sw_version' and 'sw_build' parameters must be modified by the user to select a new software version. The subfolder relative to these two parameters must be present on data folder. This subfolder must contain the software binary and eds file needed by the script.

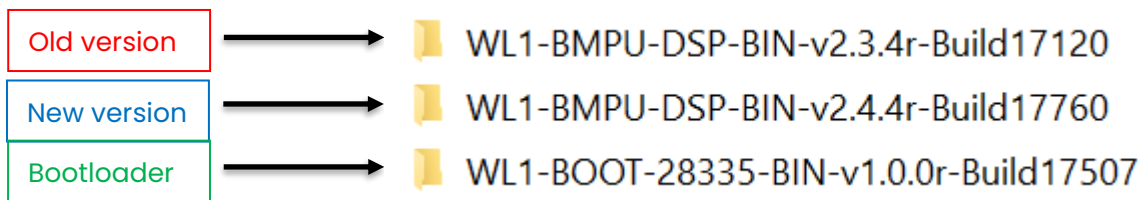


Figure 1: data folder example layout

5. Programming procedure

To execute software upgrade script, the steps to follow are

- Connect BMPU to auxiliary power supply (12/24V)
- Disconnect BMPU from main or battery
- Connect CAN Transceiver kvaser to PC with Windows OS.
- Launch by double-click the batch script **program_bmpu_rx_pyc**
- Command window as shown in the figure below (Figure 2) will be started
- Procedure takes few minutes
- Make sure that at the end of the operation the following message appears

```

: *****
:
:           Programming script ended at 2022-05-05 10:37:20.184948
:           Recap of the programming sequence
:
: *****
:
: node bmpu1 is programmed with software v2.4.4r build: 17760
  
```

It indicated the success of the reprogramming.

- If operation fails, restart the auxiliary power supply and the repeat the process (double-click the batch script **program_bmpu_rx_pyc**)

```

C:\WINDOWS\system32\cmd.exe
INFO : Scan Network
+K[Scan Network Delay | 5.00 s / 5.00 s

INFO : Node ids detected on Network
INFO : => CAN node id 0x7E (126) detected on CAN Bus
INFO : => CAN node id 0x50 (80) detected on CAN Bus
INFO :
INFO : Access sw information about node Bootloader id 0x7E (126)
INFO : Load eds file C:\Users\meghnor1\Desktop\BMPU-R2 reprogramming package\SinglePU_prog_scripts\data\WL1-BOOT-28335-BIN-v1.0.0r-Build17507\WL1-BOOT-28335-BIN-v1.0.0r-Build17507.eds for Bootloader id 0x7E (126)
INFO :
INFO : Check bootloader state before Erase
INFO : => Bootloader is waiting for FW upgrade
INFO :
INFO : Unlock bootloader
+K[Unlock bootloader | 1.00 s / 1.00 s

INFO : Erase Flash
+K[Erase flash | 30.00 s / 30.00 s

INFO : Check Flash erasing before programming
INFO : => Flash is successfully erased
INFO :
INFO : Program Target
INFO : Initiating block download for 0x1F50:1
+K[Program Target | 132972 bytes / 26252 bytes

```

Figure 2 Command window

```

C:\WINDOWS\system32\cmd.exe
INFO : => Calibration subindex 53: name = calib_dc_dc_i_sensor_g2v_offset, value = 1.600000023841858
INFO : => Calibration subindex 54: name = calib_dc_dc_i_sensor_v2g_offset, value = 1.600000023841858
INFO : => Calibration subindex 55: name = calib_q_offset_g2v, value = 0.0
INFO : => Calibration subindex 56: name = calib_q_offset_v2g, value = 0.0
INFO : => Calibration subindex 0: name = number_of_distant_pu, value = 0
INFO :
INFO : Calibration parameters were correctly loaded
INFO : Save control objects from node bmpu1 PFC id 0x50 (80)
INFO : control object :
INFO : => control subindex 1: name = hardware_version, value = 210.0
INFO : => control subindex 0: name = number_of_distant_pu, value = 0
INFO :
INFO : Control parameters were correctly loaded
INFO : Node bmpu1 PFC id 0x50 (80) has been correctly programmed: v2.4.4r build 17760
INFO : Changing NMT state from OPERATIONAL to INITIALISING
INFO : Sending NMT command 0x82 to node 80
INFO : Programming finished!!
INFO :
INFO : *****
INFO :
INFO : Programming script ended at 2022-05-05 10:37:20.184948
INFO : Recap of the programming sequence
INFO : *****
INFO :
INFO : node bmpu1 is programmed with software v2.4.4r build: 17760
+[]?25h
C:\Users\meghnor1\Desktop\BMPU-R2 reprogramming package\SinglePU_prog_scripts>pause
Press any key to continue . . .

```

Figure 3 End of operation message