



pySimpleBGC

PySimpleBGC Documentation

Release 0.1dev

Lionel Darras & Franck Perret

Nov 29, 2017

1 About

This project was conducted by the laboratories [UMR5133-Archéorient \(FR\)](#) and [UMR56000-Environnement Ville Société \(FR\)](#).

2 Description

PySimpleBGC is a python project which aims to allow the communication with Basecam Simple-BGC controller board.

The main features include automatic collecting of data and settings (read only) as a list of dictionaries.

The tool can be used in your python scripts for data post-processing, or in command line mode to collect data in CSV files.

Note: PySimpleBGC uses the [PyLink](#) lib, offers a universal communication interface with File-Like API.

2.1 Examples

We init communication by giving the datalogger URL.

```
>>> from pysimplebgc import SimpleBGC32
>>> device = SimpleBGC32.from_url('tcp:host-ip:port')
>>> # or with Serial connection
>>> device = SimpleBGC32.from_url('serial:/dev/ttyUSB0:38400:8N1')
```

To get list of commands, use:

```
>>> device.getcmdlist()
['CMD_BOARD_INFO_3', 'CMD_BOARD_INFO', 'CMD_REALTIME_DATA_4',
 'CMD_REALTIME_DATA_3', ...]
```

To execute one of these commands, you can use “device.setcmd(cmdtype, cmdparams)” with cmdtype = ‘CMD_BOARD_INFO’ ... and cmdparams, the parameters needed by the command.

```
>>> device.setcmd('CMD_BOARD_INFO', "")
[{'name': 'BOARD_VER', 'valuefmt': '%d', 'framefmt': 'B', 'value': 31},
 ...]
```

This operation returns a list of dictionnaires with each information for the parameter returned.

2.2 Features

- Collecting real-time data in a CSV file
- Reading settings
- Various types of connections are supported (TCP, UDP, Serial, GSM)
- Comes with a command-line script
- Compatible with Python 3.x

2.3 Installation

You can install, upgrade, uninstall PySimpleBGC with these commands

```
$ pip install pysimplebgc
$ pip install --upgrade pysimplebgc
$ pip uninstall pysimplebgc
```

Or if you don't have pip

```
$ easy_install pysimplebgc
```

Or you can get the source code from [github](#).

```
$ git clone https://github.com/LionelDarras/PySimpleBGC.git
$ cd PySimpleBGC
$ python setup.py install
```

2.4 About Basecam SimpleBGC 3-Axis Type Controller Board

PySimpleBGC implement part of [SimpleBGC 2.5 serial protocol](#) and can thus communicate with this type of controller board.

2.5 Command-line usage

PySimpleBGC has a command-line script that interacts with the controller board.

```
$ pysimplebgc -h
usage: pysimplebgc32 [-h] [--version] {getboardinfo,getboardinfo3,
getrealtimedata3,collectdata3,collectdata4} ...

Communication tools for Basecam SimpleBGC 3-Axis Controller Board

optional arguments:
  -h, --help            Show this help message and exit
  --version             Print PySimpleBGC's version number and exit.

The PySimpleBGC commands:
  getboardinfo          Get board and firware information.
  getboardinfo3         Get additionnal board information.
  getrealtimedata3     Get current real-time data.
  collectdata3         Collect real-time data and save in a file.
  collectdata4         Collect extended real-time data and save in
a file.
```

3 Getboardinfo

The *getboardinfo* command gives board and firmware information.

```
$ pysimplebgc32 getboardinfo -h
usage: pysimplebgc32 getboardinfo [-h] [--timeout TIMEOUT] [--debug]
url
```

Print board and firmware information.

positional arguments:

url Specify URL for connection link.
 E.g. tcp:iphost:port or
 serial:/dev/ttyUSB0:19200:8N1

optional arguments:

-h, --help Show this help message and exit
--timeout TIMEOUT Connection link timeout (default: 10.0)
--debug Display log (default: False)

Example

```
$ pysimplebgc32 getboardinfo serial:COM1:115200:8N1
BOARD_VER : 31
FIRMWARE_VER : 2439
DEBUG_MODE : 0
BOARD_FEATURES : 2
CONNECTION_FLAGS : 1
FRW_EXTRA_ID : 0
```

4 Getboardinfo3

The *getboardinfo3* command gives additional board information.

```
$ pysimplebgc32 getboardinfo3 -h
usage: pysimplebgc32 getboardinfo3 [-h] [--timeout TIMEOUT] [--debug]
url

Print additional board information.

positional arguments:
  url                      Specify URL for connection link. E.g.
                           tcp:iphost:port or
                           serial:/dev/ttyUSB0:19200:8N1

optional arguments:
  -h, --help                Show this help message and exit
  --timeout TIMEOUT        Connection link timeout (default: 10.0)
  --debug                    Display log (default: False)
```

Example

```
$ pysimplebgc32 getboardinfo3 serial:COM1:115200:8N1
deviceID : b'\x01#/C\xb8\xc4w\x0f\xee'
mcuID : b'299 \x04W2H<\x00#\x00'
EEPROM_SIZE : 32768
SCRIPT_SLOT1_SIZE : 0
SCRIPT_SLOT2_SIZE : 0
SCRIPT_SLOT3_SIZE : 0
SCRIPT_SLOT4_SIZE : 0
SCRIPT_SLOT5_SIZE : 0
```

5 Getrealtimedata

The *getrealtimedata* command gives current real-time data.

```
$ pysimplebgc32 getrealtimedata -h
usage: pysimplebgc32 getrealtimedata [-h] [--timeout TIMEOUT] [--debug]
url

Get current real-time data.

positional arguments:
  url                    Specifiy URL for connection link.
                        E.g. tcp:iphost:port or
                        serial:/dev/ttyUSB0:19200:8N1

optional arguments:
  -h, --help            Show this help message and exit
  --timeout TIMEOUT     Connection link timeout (default: 10.0)
  --debug               Display log (default: False)
```

Example

```
$ pysimplebgc32 getrealtimedata serial:COM1:115200:8N1
ACC_ROLL : 64
GYRO_ROLL : -3
ACC_PITCH : -10
GYRO_PITCH : -18
ACC_YAW : -472
GYRO_YAW : -1
...
MOTOR_POWER_ROLL : 0
MOTOR_POWER_PITCH : 0
MOTOR_POWER_YAW : 0
```

6 Collectdata3

The *collectdata3* command collect real-time data and save in a file.

```
$ pysimplebgc32 collectdata3 -h
usage: pysimplebgc32 getboardinfo3 [-h] [--timeout TIMEOUT] [--debug]
                                     [--output OUTPUT] [--delim DELIM]
                                     [--stdout] [--measuresnb MEASURESNB]
                                     [--samplingperiod SAMPLINGPERIOD]
                                     [--storingperiod STORINGPERIOD]
                                     url

Collect real-time data and save in a file.

positional arguments:
  url                    Specifiy URL for connection link. E.g.
                        tcp:iphost:port or
                        serial:/dev/ttyUSB0:19200:8N1

optional arguments:
  -h, --help            Show this help message and exit
  --timeout TIMEOUT    Connection link timeout (default: 10.0)
  --debug              Display log (default: False)
  --output OUTPUT      Filename where output is written
                        (default: standard out)
  --delim DELIM        CSV char delimiter (default: ';')
  --stdout             Display on the standard out if
                        defined output is a file
  --measuresnb MEASURESNB
                        Number of measures to realize, 0
                        if continue until break (Ctrl-C)
  --samplingperiod SAMPLINGPERIOD
                        Period of sampling, 10ms
                        (default: 10 (10*10ms = 100ms))
  --storingperiod STORINGPERIOD
                        Period of storing, 10ms
                        (default: 10 (10*10ms = 100ms))
```

Example

```
$ pysimplebgc32 collectdata3 serial:COM1:115200:8N1 --output save.csv
--stdout
DATETIME; ACC_ROLL;GYRO_ROLL;ACC_PITCH;GYRO_PITCH; ...;MOTOR_POWER_YAW
2016-01-04 10:33:39.465;62;-2;-9;0;-471;-5; ...;0
2016-01-04 10:33:40.465;62;0;-11;0;-471;-4; ...;0
2016-01-04 10:33:41.474;64;-3;-9;-2;-473;-6; ...;0
2016-01-04 10:33:42.474;66;-11;-11;-18;-472;-4; ...;0
...
```

7 Collectdata4

The *collectdata4* command collect extended real-time data and save in a file.

```
$ pysimplebgc32 collectdata4 -h
usage: pysimplebgc32 getboardinfo4 [-h] [--timeout TIMEOUT] [--debug]
                                     [--output OUTPUT] [--delim DELIM]
                                     [--stdout] [--measuresnb MEASURESNB]
                                     [--samplingperiod SAMPLINGPERIOD]
                                     [--storingperiod STORINGPERIOD]
                                     url
```

Collect extended real-time data and save in a file.

positional arguments:

url Specify URL for connection link.
 E.g. tcp:iphost:port or
 serial:/dev/ttyUSB0:19200:8N1

optional arguments:

-h, --help Show this help message and exit
--timeout TIMEOUT Connection link timeout (default: ↪10.0)
--debug Display log (default: False)
--output OUTPUT Filename where output is written
 (default: standard out)
--delim DELIM CSV char delimiter (default: ';')
--stdout Display on the standard out
 if defined output is a file
--measuresnb MEASURESNB Number of measures to realize, 0
 if continue until break (Ctrl-C)
--samplingperiod SAMPLINGPERIOD Period of sampling, 10ms
 (default: 10 (10*10ms = 100ms))
--storingperiod STORINGPERIOD Period of storing, 10ms
 (default: 10 (10*10ms = 100ms))

Example

```
$ pysimplebgc32 collectdata4 serial:COM1:115200:8N1 --output save.csv
--stdout
DATETIME; ACC_ROLL;GYRO_ROLL;ACC_PITCH;GYRO_PITCH; ...
;FRAME_IMU_TEMPERATURE
2016-01-04 10:54:26.261;58;1;-11;1;-472;-6; ...;0
2016-01-04 10:54:27.261;58;-4;-13;-6;-475;-6; ...;0
2016-01-04 10:54:28.261;58;9;-12;6;-469;-6; ...;0
2016-01-04 10:54:29.261;59;2;-11;1;-471;-7; ...;0
2016-01-04 10:54:30.261;59;-1;-11;-1;-471;-5; ...;0
```

```
2016-01-04 10:54:31.265;58;-2;-10;-6;-471;-4; ...;0
...
```

8 Debug mode

You can use debug option if you want to print log and see the flowing data.

```
$ pysimplegcc32 getboardinfo serial:COM1:115200 --debug
2016-01-04 12:00:37,455 INFO: new <SerialLink serial:COM1:115200:8N1>
    was initialized
2016-01-04 12:00:37,455 INFO: check validity of command type:
    OK CMD_BOARD_INFO
2016-01-04 12:00:37,455 INFO: try pack command : CMD_BOARD_INFO
2016-01-04 12:00:37,455 INFO: check CMDID: OK (CMD_BOARD_INFO,86,0)
2016-01-04 12:00:37,455 INFO: check CMDBODY: OK(0)
2016-01-04 12:00:37,455 INFO: try send : >V V
2016-01-04 12:00:37,455 INFO: write : <'>V\x00V\x00'>
2016-01-04 12:00:37,485 INFO: read : <3E 56 12 68 1F 87 09 00 02 00 01
    00 00 00 00 00 00 00 00 00 00 00 B2>
2016-01-04 12:00:37,485 INFO: try unpack response : b'>V\x12h\x1f\x87
    \t\x00\x02\x00\x01\x00\x00\x00\x00\x00
    \x00\x00\x00\x00\x00\x00'
2016-01-04 12:00:37,485 INFO: check MINRESPSIZE: OK (23)
2016-01-04 12:00:37,485 INFO: check ACK: OK (b'>V')
2016-01-04 12:00:37,485 INFO: check HEADERCRC: OK (104)
2016-01-04 12:00:37,485 INFO: check DATASIZE: OK (18)
2016-01-04 12:00:37,485 INFO: check DATACRC: OK (b2)
2016-01-04 12:00:37,485 INFO: unpacked data: b'\x1f\x87\t\x00\x02\x00
    \x01\x00\x00\x00\x00\x00\x00\x00\x00\x00
    \x00\x00'

BOARD_VER : 31
FIRMWARE_VER : 2439
DEBUG_MODE : 0
BOARD_FEATURES : 2
CONNECTION_FLAGS : 1
FRW_EXTRA_ID : 0
2016-01-04 12:00:37,505 INFO: connection <SerialLink
    serial:COM1:115200:8N1> was closed
```


8.1 API reference

8.2 Feedback & Contribute

Your feedback is more than welcome. Write email to the [PySimpleBGC32 mailing list](#).

There are several ways to contribute to the project:

1. Post bugs and feature [requests on github](#).
2. Fork [the repository](#) on Github to start making your changes.
3. Write a test which shows that the bug was fixed or that the feature works as expected.
4. Send a pull request and bug the maintainer until it gets merged and published. :) Make sure to add yourself to [AUTHORS](#).

9 Changelog

Released on 2016-01-15.

- First properly tagged release.

Index

P

pysimplebgc (module), 1