

```
python ./bled112_scanner_upgrade_v5.py --h
```

Help file for using Bluetooth BLE112 scanner tool

Usage: bled112_scanner_upgrade_v5.py [options]

Bluetooth Smart Scanner Updated script for Bluegiga BLED112 v2020-11-30

Options:

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

Serial Port Options:

-p PORT, --port=PORT Serial port device name (default /dev/ttyACM0)
-b BAUD, --baud=BAUD Serial port baud rate (default 115200)

Scan Options:

-i INTERVAL, --interval=INTERVAL Scan interval width in units of 0.625ms (default 200)
-w WINDOW, --window=WINDOW Scan window width in units of 0.625ms (default 200)
-a, --active Perform active scan (default passive)
NOTE: active scans result in a 'scan response' request being sent to the slave device, which should send a follow-up scan response packet. This will result in increased power consumption on the slave device.

Filter Options:

-u UUID, --uuid=UUID Service UUID(s) to match
-m ADDRESS, --mac=ADDRESS MAC address(es) to match format :: xxxxxxxxxx
-S payload_instr, --instr=payload_instr Search match 'Payload data' for match ::format :: xxxxxxxxxx
-y byte_position, --byte=byte_position Select single byte from 'Payload data' when there is a match from '--instr search' '--byte=byte_position' output out in byte column, !! first character position is Zero !!
-R RSSI, --rssi=RSSI RSSI minimum filter (-110 to -20), omit to disable
-I, --install Guide of how to install

Output Options:

-Z, --switch If options '--instr search' and '--byte=byte_position' selected. Put byte value in RSSI column

-q, --quiet Quiet mode (suppress initial scan parameter display)

-z, --time_in_ms time_in_ms (Display time in milliseconds)

-k, --csv CVS mode (If options -q and -f are set output in directly excel csv file friendly format)

-c, --comma Comma mode (If options -q and -f are set output in basic excel csv file not friendly format)

-x, --plot Plot mode , If options '-q -f -c -x --time_in_ms -d tr' are set use live plot graph of rssi verses time)

-X, --plotbyte Plot mode , If options '-q -f -c -x --time_in_ms -d tr' are set use live plot graph of payload selected byte verses time)

-f, --friendly Friendly mode (output in human-readable format)

-d FIELDS, --display=FIELDS Display fields and order (default '%default')

 t = Unix time, with milliseconds

 r = RSSI measurement (signed integer)

 p = Packet type (0 = normal, 4 = scan response)

 s = Sender MAC address (hexadecimal)

 a = Address type (0 = public, 1 = random)

 b = Bonding status (255 = no bond, else bond handle)

 d = Advertisement data payload (hexadecimal)

Examples:

bled112_scanner_upgrade_V5.py

Default options, passive scan, display all devices

bled112_scanner_upgrade_V5.py -p /dev/ttyUSB0 -d sd

Use ttyUSB0, display only sender MAC address and ad data payload

bled112_scanner_upgrade_V5.py -u 1809 -u 180D

Display only devices advertising Health Thermometer service (0x1809) or the Heart Rate service (0x180D)

bled112_scanner_upgrade_V5.py -m 00:07:80 -m 08:57:82:bb:27:37

Display only devices with a Bluetooth address (MAC) starting with the Bluegiga OUI (00:07:80), or exactly matching 08:57:82:bb:27:37

Sample Output Explanation:

1364699494.574 -57 0 000780814494 0 255 02010603030918

't' (Unix time): 1364699464.574, 1364699591.128, etc.

'r' (RSSI value): -57, -80, -92, etc.

'p' (Packet type): 0 (advertisement), 4 (scan response)

's' (Sender MAC): 000780535BB4, 000780814494, etc. (search in the actual MAC field

and not the payload status data

'a' (Address type): 0 (public), 1 (random)

'b' (Bond status): 255 (no bond), 0 to 15 if bonded

'd' (Data payload): 02010603030918, etc.

See BT4.0 Core Spec for details about ad packet format

```
=====
Examples graph plot functions running on Microsoft Window WINxx or Apple OSX

( !! NOTE : Must be using Activestate ActivePython and not regular Python from www.python.org !!!! )
+++++
```

Note : Command line instruction examples are in blue

Use for plotting rssi to graph plot function in MS WinXX

```
python ./bled112_scanner_upgrade_v5.py -q -f -c -x --time_in_ms -d tr -p com27
```

Use for plotting rssi to dos CMD function in MS WinXX

```
python ./bled112_scanner_upgrade_v5.py -q -f -c --time_in_ms -d tr -p com27
```

Use for plotting rssi and other data to dos CMD function in MS WinXX onwards to MS EXCEL .CSV file

```
python ./bled112_scanner_upgrade_v5.py -q -f -k --time_in_ms -p com27 1>>./dump.csv
```

Use live RSSI plot function to screen live graph

(!! NOTE : Must be using Activestate ActivePython and not regular Python from www.python.org !!!!)

```
python ./bled112_scanner_upgrade_v5.py -q -f -c -x --time_in_ms -d tr -p com27
```

Use live RSSI plot function to screen live graph and dump to excel CSV file also
(!! See Warning NOTE above !!)

For MS WinXx : `python ./bled112_scanner_upgrade_v5.py -q -f -k -x --time_in_ms -p com27 1>>./dump.csv`

For Apple OSX : `python ./bled112_scanner_upgrade_v5.py -q -f -k -x --time_in_ms -p /dev/cu.usbmodem11 1>>./dump.csv`

Using live search in the Payload_status message with live RSSI plot function to screen
live graph and dump to excel CSV file also (!! See Warning NOTE above !!)

For MS WinXx : `python3 ./bled112_scanner_upgrade_v5.py -q -f -x -k --time_in_ms -p com27 --instr=FFFE96B6E511 1>>./dump.csv`

For Apple OSX : `python3 ./bled112_scanner_upgrade_v5.py -q -f -x -k --time_in_ms -p /dev/cu.usbmodem11 --instr=FFFE96B6E511 1>>./dump.csv`

Using live search in the Payload_status message with live RSSI plot function to screen
live graph and dump to excel CSV file also (!! See Warning NOTE above !!)
plus search and find the value in byte position 09 and swap the found byte value
convert to decimal with the RSSI value.

For MS WinXx : `python3 ./bled112_scanner_upgrade_v5.py -q -f -k -x --time_in_ms -p com27 --instr=FFFE96B6E511 --byte=09 --switch`

Apple OSX :

```
python3 ./bled112_scanner_upgrade_v5.py -q -f -k -x --time_in_ms -p /dev/cu.usbmodem11 --instr=FFFE96B6E511 --byte=09 --switch
```

Using live search in the Payload_status message with live RSSI plot function to screen live graph (graph 1)
and Selected Byte from Payload data live to plot to graph (graph 2) and dump to excel CSV file
Selected Byte from Payload data = search payload data and find the value in byte position 09 .
Also (!! See Warning NOTE above !!)

For MS WinXx : `python3 ./bled112_scanner_upgrade_v5.py -q -f -k -x --time_in_ms -p com27 --instr=FFFE96B6E511 --byte=09 -X`

For Apple OSX : `python3 ./bled112_scanner_upgrade_v5.py -q -f -k -x --time_in_ms -p /dev/cu.usbmodem11 --instr=FFFE96B6E511 --byte=09 -X`

History

Changelog:

- 2013-04-07 - Fixed 128-bit UUID filters
 - Added more verbose output on startup
 - Added "friendly mode" output argument
 - Added "quiet mode" output argument
 - Improved comments in code
- 2013-03-30 - Initial release
- 2020-11-xx - Massive upgrade
 - Fixed to work with python >=3.6
 - Now works on both MS WinXX and Apple OSX x86 using
ActiveState Python (Not regular Python from Python.org)
 -
 - dump to excel .csv file function added so that actually works properly
 - real time plotting of RSSI to live graph function added
 - real time plotting of user selected packet payload against time in milliseconds
 - bglib.py code merged into this file so the whole thing is in one file

author "Jeff Rowberg & others"
license "MIT"
version "2020-11-xx"
email "jeff@rowberg.net"

Installation notes

Things that need to be installed for this script to run

- 1) Remove all other python installation from the install machine
- 2) Install activestate ActivePython from

<https://platform.activestate.com/ActiveState/ActivePython-3.8>

This is the only Python version which works straight out of the box with all the features of this script.

- 3) pip3 install pyserial future pandas matplotlib
- 4) If on Microsoft Windows WinX machine install the SI-LABS BLE112 driver
- 5) Know the serial port that your BLE112 device is mount on .

Note : The BLE112 device when installed correctly shows up as a virtual serial port . This is the case for Apple Mac OSX and Microsoft windows

Links the BLUETOOTH BLE112 USB hardware dongle can be found here

REF : <https://www.silabs.com/wireless/bluetooth/bluegiga-low-energy-legacy-modules/device.bled112>

REF : <https://www.mouser.dk/ProductDetail/Silicon-Labs/BLED112-V1/?qs=2bnkQPT%252BPaf%252BBQ6Sw8mJqQ%3D%3D>