Template Documents

Release 1.0

Beni Valotker

Jul 20, 2020

Table of Contents

1 Main Function 1

Python Module Index 3

Index 5

Main Function

class project_templates.main_function.**SimpleBleDevice** (*client*, *addr=None*, *addrType=None*, *iface=0*, *data=None*, *rssi=0*, *connectable=False*, *updateCount=0*)

This is a conceptual class representation of a simple BLE device (GATT Server). It is essentially an extended combination of the `bluepy.btle.Peripheral` and `bluepy.btle.ScanEntry` classes

Parameters

- **client** (class:*simpleble.SimpleBleClient*) – A handle to the `simpleble.SimpleBleClient` client object that detected the device
- **addr** (*str, optional*) – Device MAC address, defaults to None
- **addrType** (*str, optional*) – Device address type - one of ADDR_TYPE_PUBLIC or ADDR_TYPE_RANDOM, defaults to ADDR_TYPE_PUBLIC
- **iface** (*int, optional*) – Bluetooth interface number (0 = /dev/hci0) used for the connection, defaults to 0
- **data** (*list, optional*) – A list of tuples (adtype, description, value) containing the AD type code, human-readable description and value for all available advertising data items, defaults to None
- **rssi** (*int, optional*) – Received Signal Strength Indication for the last received broadcast from the device. This is an integer value measured in dB, where 0 dB is the maximum (theoretical) signal strength, and more negative numbers indicate a weaker signal, defaults to 0
- **connectable** (*bool, optional*) – *True* if the device supports connections, and *False* otherwise (typically used for advertising 'beacons')., defaults to *False*
- **updateCount** (*int, optional*) – Integer count of the number of advertising packets received from the device so far, defaults to 0

connect ( )

Attempts to initiate a connection with the device.

Returns *True* if connection was successful, *False* otherwise

Return type bool

disconnect ( )

Drops existing connection to device

getCharacteristics (*startHnd=1*, *endHnd=65535*, *uuids=None*)

Returns a list containing `bluepy.btle.Characteristic` objects for the peripheral. If no arguments are given, will return all characteristics. If startHnd and/or endHnd are given, the list

is restricted to characteristics whose handles are within the given range.

Parameters
- **startHnd** (*int, optional*) – Start index, defaults to 1
- **endHnd** (*int, optional*) – End index, defaults to 0xFFFF
- **uuids** (*list, optional*) – a list of UUID strings, defaults to None

Returns List of returned `bluepy.btle.Characteristic` objects

Return type list

getServices (*uuids=None*)

Returns a list of `bluepy.blte.Service` objects representing the services offered by the device. This will perform Bluetooth service discovery if this has not already been done; otherwise it will return a cached list of services immediately..

Parameters **uuids** (*list, optional*) – A list of string service UUIDs to be discovered, defaults to None

Returns A list of the discovered `bluepy.blte.Service` objects, which match the provided `uuids`

Return type list On Python 3.x, this returns a dictionary view object, not a list

isConnected ( )

Checks to see if device is connected

Returns *True* if connected, *False* otherwise

Return type bool

printInfo ( )

Print info about device

setNotificationCallback (*callback*)

Set the callback function to be executed when the device sends a notification to the client.

Parameters **callback** (*function, optional*) – A function handle of the form `callback(client, characteristic, data)`, where `client` is a handle to the `simpleble.SimpleBleClient` that invoked the callback, `characteristic` is the notified `bluepy.blte.Characteristic` object and data is a *bytearray* containing the updated value. Defaults to None

p

project_templates
    project_templates.main_function, 1

C

connect() (project_templates.main_function.SimpleBleDevice method), 1

D

disconnect() (project_templates.main_function.SimpleBleDevice method), 1

G

getCharacteristics() (project_templates.main_function.SimpleBleDevice method), 1
getServices() (project_templates.main_function.SimpleBleDevice method), 2

I

isConnected() (project_templates.main_function.SimpleBleDevice method), 2

M

module
    project_templates.main_function, 1

P

printInfo() (project_templates.main_function.SimpleBleDevice method), 2
project_templates.main_function
    module, 1

S

setNotificationCallback() (project_templates.main_function.SimpleBleDevice method), 2
SimpleBleDevice (class in project_templates.main_function), 1