Robotlibrary

Release 1.1.1

Marcus Jacobs, Nils Jacobs

Nov 05, 2024

CONTENTS:

1 API Reference 3

1.1 robotlibrary . . . 3

2 Indices and tables 25

Python Module Index 27

Index 29

Hello

CHAPTER

ONE

API REFERENCE

This page contains auto-generated API reference documentation[1].

1.1 robotlibrary

1.1.1 Subpackages

robotlibrary.bluetooth

Submodules

robotlibrary.bluetooth.advertising

Attributes

_ADV_MAX_PAYLOAD	

Functions

advertising_payload([limited_disc, br_edr, name, ...])	
decode_field(payload, adv_type)	
decode_services(payload)	

[1] Created with sphinx-autoapi

Module Contents

robotlibrary.bluetooth.advertising.**_ADV_MAX_PAYLOAD**

robotlibrary.bluetooth.advertising.**advertising_payload**(*limited_disc=False*, *br_edr=False*, *name=None*, *services=None*, *appearance=0*)

robotlibrary.bluetooth.advertising.**decode_field**(*payload*, *adv_type*)

robotlibrary.bluetooth.advertising.**decode_services**(*payload*)

robotlibrary.bluetooth.ble_flags

Attributes

ADV_TYPE_FLAGS
ADV_TYPE_NAME
ADV_TYPE_UUID16_COMPLETE
ADV_TYPE_UUID32_COMPLETE
ADV_TYPE_UUID128_COMPLETE
ADV_TYPE_UUID16_MORE
ADV_TYPE_UUID32_MORE
ADV_TYPE_UUID128_MORE
ADV_TYPE_APPEARANCE
ADV_TYPE_MANUFACTURER_DATA
IRQ_CENTRAL_CONNECT
IRQ_CENTRAL_DISCONNECT
IRQ_GATTS_WRITE
IRQ_GATTS_READ_REQUEST
IRQ_SCAN_RESULT
IRQ_SCAN_DONE
IRQ_PERIPHERAL_CONNECT

continues on next page

Table 1 – continued from previous page

IRQ_PERIPHERAL_DISCONNECT
IRQ_GATTC_SERVICE_RESULT
IRQ_GATTC_SERVICE_DONE
IRQ_GATTC_CHARACTERISTIC_RESULT
IRQ_GATTC_CHARACTERISTIC_DONE
IRQ_GATTC_DESCRIPTOR_RESULT
IRQ_GATTC_DESCRIPTOR_DONE
IRQ_GATTC_READ_RESULT
IRQ_GATTC_READ_DONE
IRQ_GATTC_WRITE_DONE
IRQ_GATTC_NOTIFY
IRQ_GATTC_INDICATE
IRQ_GATTS_INDICATE_DONE
IRQ_MTU_EXCHANGED
IRQ_L2CAP_ACCEPT
IRQ_L2CAP_CONNECT
IRQ_L2CAP_DISCONNECT
IRQ_L2CAP_RECV
IRQ_L2CAP_SEND_READY
IRQ_CONNECTION_UPDATE
IRQ_ENCRYPTION_UPDATE
IRQ_GET_SECRET
IRQ_SET_SECRET
GATTS_NO_ERROR
GATTS_ERROR_READ_NOT_PERMITTED
GATTS_ERROR_WRITE_NOT_PERMITTED

continues on next page

Table 1 – continued from previous page

GATTS_ERROR_INSUFFICIENT_AUTHENTICATION
GATTS_ERROR_INSUFFICIENT_AUTHORIZATION
GATTS_ERROR_INSUFFICIENT_ENCRYPTION
PASSKEY_ACTION_NONE
PASSKEY_ACTION_INPUT
PASSKEY_ACTION_DISPLAY
PASSKEY_ACTION_NUMERIC_COMPARISON
ADV_IND
ADV_DIRECT_IND
ADV_SCAN_IND
ADV_NONCONN_IND
SCAN_RSP
FLAG_READ
FLAG_WRITE_NO_RESPONSE
FLAG_WRITE
FLAG_NOTIFY

Module Contents

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_FLAGS**

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_NAME**

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_UUID16_COMPLETE**

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_UUID32_COMPLETE**

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_UUID128_COMPLETE**

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_UUID16_MORE**

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_UUID32_MORE**

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_UUID128_MORE**

robotlibrary.bluetooth.ble_flags.**ADV_TYPE_APPEARANCE** robotlibrary.bluetooth.ble_flags.**ADV_TYPE_MANUFACTURER_DATA**

robotlibrary.bluetooth.ble_flags.**IRQ_CENTRAL_CONNECT**

robotlibrary.bluetooth.ble_flags.**IRQ_CENTRAL_DISCONNECT**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTS_WRITE**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTS_READ_REQUEST**

robotlibrary.bluetooth.ble_flags.**IRQ_SCAN_RESULT**

robotlibrary.bluetooth.ble_flags.**IRQ_SCAN_DONE**

robotlibrary.bluetooth.ble_flags.**IRQ_PERIPHERAL_CONNECT**

robotlibrary.bluetooth.ble_flags.**IRQ_PERIPHERAL_DISCONNECT**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_SERVICE_RESULT**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_SERVICE_DONE**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_CHARACTERISTIC_RESULT**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_CHARACTERISTIC_DONE**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_DESCRIPTOR_RESULT**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_DESCRIPTOR_DONE**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_READ_RESULT**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_READ_DONE**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_WRITE_DONE**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_NOTIFY**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTC_INDICATE**

robotlibrary.bluetooth.ble_flags.**IRQ_GATTS_INDICATE_DONE**

robotlibrary.bluetooth.ble_flags.**IRQ_MTU_EXCHANGED**

robotlibrary.bluetooth.ble_flags.**IRQ_L2CAP_ACCEPT**

robotlibrary.bluetooth.ble_flags.**IRQ_L2CAP_CONNECT**

robotlibrary.bluetooth.ble_flags.**IRQ_L2CAP_DISCONNECT**

robotlibrary.bluetooth.ble_flags.**IRQ_L2CAP_RECV**

robotlibrary.bluetooth.ble_flags.**IRQ_L2CAP_SEND_READY**

robotlibrary.bluetooth.ble_flags.**IRQ_CONNECTION_UPDATE**

robotlibrary.bluetooth.ble_flags.**IRQ_ENCRYPTION_UPDATE**

robotlibrary.bluetooth.ble_flags.**IRQ_GET_SECRET**

robotlibrary.bluetooth.ble_flags.**IRQ_SET_SECRET** robotlibrary.bluetooth.ble_flags.**GATTS_NO_ERROR**

robotlibrary.bluetooth.ble_flags.**GATTS_ERROR_READ_NOT_PERMITTED**

robotlibrary.bluetooth.ble_flags.**GATTS_ERROR_WRITE_NOT_PERMITTED**

robotlibrary.bluetooth.ble_flags.**GATTS_ERROR_INSUFFICIENT_AUTHENTICATION**

robotlibrary.bluetooth.ble_flags.**GATTS_ERROR_INSUFFICIENT_AUTHORIZATION**

robotlibrary.bluetooth.ble_flags.**GATTS_ERROR_INSUFFICIENT_ENCRYPTION**

robotlibrary.bluetooth.ble_flags.**PASSKEY_ACTION_NONE**

robotlibrary.bluetooth.ble_flags.**PASSKEY_ACTION_INPUT**

robotlibrary.bluetooth.ble_flags.**PASSKEY_ACTION_DISPLAY**

robotlibrary.bluetooth.ble_flags.**PASSKEY_ACTION_NUMERIC_COMPARISON**

robotlibrary.bluetooth.ble_flags.**ADV_IND**

robotlibrary.bluetooth.ble_flags.**ADV_DIRECT_IND**

robotlibrary.bluetooth.ble_flags.**ADV_SCAN_IND**

robotlibrary.bluetooth.ble_flags.**ADV_NONCONN_IND**

robotlibrary.bluetooth.ble_flags.**SCAN_RSP**

robotlibrary.bluetooth.ble_flags.**FLAG_READ**

robotlibrary.bluetooth.ble_flags.**FLAG_WRITE_NO_RESPONSE**

robotlibrary.bluetooth.ble_flags.**FLAG_WRITE**

robotlibrary.bluetooth.ble_flags.**FLAG_NOTIFY**

robotlibrary.bluetooth.ble_services_definitions

Attributes

MOTOR_TX_UUID
MOTOR_RX_UUID
ULTRASONIC_UUID
INFRARED_UUID
ROBOT_UUID
MOTOR_RX
MOTOR_TX
ROBOT_SERVICE

Module Contents

robotlibrary.bluetooth.ble_services_definitions.**MOTOR_TX_UUID**

robotlibrary.bluetooth.ble_services_definitions.**MOTOR_RX_UUID**

robotlibrary.bluetooth.ble_services_definitions.**ULTRASONIC_UUID**

robotlibrary.bluetooth.ble_services_definitions.**INFRARED_UUID**

robotlibrary.bluetooth.ble_services_definitions.**ROBOT_UUID**

robotlibrary.bluetooth.ble_services_definitions.**MOTOR_RX**

robotlibrary.bluetooth.ble_services_definitions.**MOTOR_TX**

robotlibrary.bluetooth.ble_services_definitions.**ROBOT_SERVICE**

robotlibrary.bluetooth.central

Classes

BLECentral

Module Contents

class robotlibrary.bluetooth.central.**BLECentral**(*to_connect_name: str*, *add_robot_stuff=False*)

  _to_connect_name

  ble

  _irq_dict

  _handles

  _connections

  _read_callbacks

  _read_cb = None

  _service_to_scan = []

  _handle_scan(*data*)

  _handle_connect(*data*)

  _handle_disconnect(*data*)

  _handle_services(*data*)

  _on_service_discovery_complete(*data*)

  _handle_characteristics(*data*)

  _handle_read(*data*)

  _irq(*event: int*, *data*)

  register_irq(*event: int*, *func*)

  scan()

  register_read_callback(*uuid*, *callback*)

  send(*service_uuid*, *char_uuid*, *data*)

  is_connected()

robotlibrary.bluetooth.main_central

Functions

read(buffer)	
main()	

Module Contents

`robotlibrary.bluetooth.main_central.`**`read`**`(`*buffer*`)`

`robotlibrary.bluetooth.main_central.`**`main`**`()`

robotlibrary.bluetooth.main_peripheral

Functions

main()

Module Contents

`robotlibrary.bluetooth.main_peripheral.`**`main`**`()`

robotlibrary.bluetooth.message_parser

Functions

decode_motor(msg)
encode_motor(left_forward, right_forward, left_speed, ...)

Module Contents

`robotlibrary.bluetooth.message_parser.`**`decode_motor`**`(`*msg: bytes*`)`

`robotlibrary.bluetooth.message_parser.`**`encode_motor`**`(`*left_forward*, *right_forward*, *left_speed*, *right_speed*`)`

robotlibrary.bluetooth.motor

Classes

Motor

Module Contents

class robotlibrary.bluetooth.motor.**Motor**(*gpio_pin_forwards: int*, *gpio_pin_backwards: int*, *on_change_callback=None*)

MAX_SPEED = 58000

_callback

_speed = 0

_moving_forwards = True

get_speed() → int

set_speed(*speed: int*)

set_direction(*forwards: bool*)

moving_forwards() → bool

robotlibrary.bluetooth.parser

Functions

decode_motor(data)
encode_motor(→ bytes)

Module Contents

robotlibrary.bluetooth.parser.**decode_motor**(*data: bytes*)

robotlibrary.bluetooth.parser.**encode_motor**(*speed: int*, *turn: int*, *forward: bool*) → bytes

robotlibrary.bluetooth.peripheral

Classes

BLEPeripheral

Module Contents

class robotlibrary.bluetooth.peripheral.**BLEPeripheral**(*name: str*, *add_robot_stuff=False*)

ble

_ble_irq_dict

_handles

_connections

_read_callbacks

_payload

register_irq(*event*, *func*)

_irq(*event*, *data*)

_handle_connect(*data*)

_handle_disconnect(*data*)

_handle_read(*data*)

send(*service_uuid*, *char_uuid*, *data*)

is_connected()

advertise(*interval_us=500000*)

register_read_callback(*uuid*, *callback*)

robotlibrary.bluetooth.pin_map

Attributes

MOTOR_LEFT_FORWARD
MOTOR_LEFT_BACKWARD
MOTOR_RIGHT_FORWARD
MOTOR_RIGHT_BACKWARD
pin_map

Module Contents

robotlibrary.bluetooth.pin_map.**MOTOR_LEFT_FORWARD**

robotlibrary.bluetooth.pin_map.**MOTOR_LEFT_BACKWARD**

robotlibrary.bluetooth.pin_map.**MOTOR_RIGHT_FORWARD**

robotlibrary.bluetooth.pin_map.**MOTOR_RIGHT_BACKWARD**

robotlibrary.bluetooth.pin_map.**pin_map**

1.1.2 Submodules

robotlibrary.conf

Attributes

project
copyright
author
release
extensions
templates_path
exclude_patterns
html_theme
html_static_path
autoapi_dirs

Module Contents

robotlibrary.conf.**project** = **'Robotlibrary'**

robotlibrary.conf.**copyright** = **'2024, Marcus Jacobs, Nils Jacobs'**

robotlibrary.conf.**author** = **'Marcus Jacobs, Nils Jacobs'**

robotlibrary.conf.**release** = **'1.1.1'**

robotlibrary.conf.**extensions** = **[]**

robotlibrary.conf.**templates_path = ['_templates']**

robotlibrary.conf.**exclude_patterns = ['_build', 'Thumbs.db', '.DS_Store']**

robotlibrary.conf.**html_theme = 'alabaster'**

robotlibrary.conf.**html_static_path = ['_static']**

robotlibrary.conf.**autoapi_dirs = ['/home/marcus/Schule/Informatik/Robotik/SMARS/robotlibrary/',...**

robotlibrary.config

This defines the parameters for the joystick. Don’t change if you don’t know what you are doing.

Attributes

JS_X_MEDIAN	
JS_Y_MEDIAN	
JS_MAX_DUTY	
JS_MIN_DUTY	
ROBOT_NAME	This defines the parameters for the motors.
MAX_DUTY	
MIN_DUTY	
MIN_SPEED	
MAX_SPEED	This defines the waiting time for the debouncing of the buttons. Leave as it is if
DEBOUNCE_WAIT	Use these constants to check for white or black with the IR-sensor.
WHITE_DETECTED	
BLACK_DETECTED	Motors and ultrasonic sensor must use consecutive pins. Use >None< if you don't use the sensor.
ML	
MR	
US	
IR	
SERVO	

Module Contents

`robotlibrary.config.JS_X_MEDIAN = 29940`

`robotlibrary.config.JS_Y_MEDIAN = 30510`

`robotlibrary.config.JS_MAX_DUTY = 65535`

`robotlibrary.config.JS_MIN_DUTY = 260`

`robotlibrary.config.ROBOT_NAME = 'HAL9000'`

This defines the parameters for the motors.

MAX_DUTY: Set to lower than the maximum not to overload the motors.

MIN_DUTY: You can leave this at 0. Set MIN_SPEED instead.

MIN_SPEED: Set this to a value slightly below the speed that sets the robot in motion.

MAX_SPEED: If you want another scale than 0-100, set the maximum here.

`robotlibrary.config.MAX_DUTY = 60000`

`robotlibrary.config.MIN_DUTY = 0`

`robotlibrary.config.MIN_SPEED = 45`

`robotlibrary.config.MAX_SPEED = 100`

This defines the waiting time for the debouncing of the buttons. Leave as it is if you don’t know what it means.

`robotlibrary.config.DEBOUNCE_WAIT = 30`

Use these constants to check for white or black with the IR-sensor.

`robotlibrary.config.WHITE_DETECTED = 0`

`robotlibrary.config.BLACK_DETECTED = 1`

Motors and ultrasonic sensor must use consecutive pins. Use >None< if you don’t use the sensor.

`robotlibrary.config.ML = 12`

`robotlibrary.config.MR = 14`

`robotlibrary.config.US = 16`

`robotlibrary.config.IR = None`

`robotlibrary.config.SERVO = None`

robotlibrary.infrared

Classes

IR	This class manages the IR-sensor. Write your code in Robot.ir_detected()

Module Contents

class `robotlibrary.infrared.`**`IR`**(*pinNo*, *robot*)

This class manages the IR-sensor. Write your code in Robot.ir_detected()

`out`

`robot`

`ir`

`detected = False`

`timer`

`reset_detected`(*t*)

`obstacle`(*pin*)

This is called on any change in the IR-sensor.

robotlibrary.joystick

Attributes

joystick

Classes

Joystick

Module Contents

class `robotlibrary.joystick.`**`Joystick`**(*x*, *y*, *b*)

`x`

`y`

`b`

`pressed = False`

`last_pressed = 0`

`timer`

`reset`(*t*)

button_handler(*pin*)

get_speed(*s*)

get_direction(*d*)

robotlibrary.joystick.**joystick**

robotlibrary.motor

Classes

Motor	This class manages the motor. Don't edit!

Module Contents

class robotlibrary.motor.**Motor**(*pinNo*)

This class manages the motor. Don't edit!

gpio

speed = 0

forward = True

pwm1

pwm2

speed_offset = 0

set_speed(*s*)

Sets the speed of the motor. Checks for sensible input.

change_speed(*sc*)

This defines an offset to the speed in motor. It is used with the remote control to turn the robot.

reset_offset()

off()

set_forward(*forward*)

Sets the motor to forward or backward without changing the speed.

robotlibrary.rc

Classes

RC	This class represents the remote control with two rotary encoders and a slider to set the speed. Don't edit unless you know what you are doing.

Functions

main()	

Module Contents

`class robotlibrary.rc.RC`

This class represents the remote control with two rotary encoders and a slider to set the speed. Don't edit unless you know what you are doing.

`forward = True`

`speed = 0`

`turn_val = 0`

`change = True`

`rotary_top`

`rotary_bottom`

`timer`

`send_timer`

`duty_cycle = 0`

`p`

`server`

read(*a*)

send(*t*)

rotary_changed(*change*)

This is called when the direction knob is turned to determine the turn or spin.

button()

This is the button click.

set_speed(*t*)

This calculates the speed between MIN_SPEED and MAX_SPEED that is sent to the robot.

`robotlibrary.rc.main()`

robotlibrary.robot

Classes

Robot	This is the central class which manages and uses all the other components of the robot. The parameters are defined in config.py

Functions

main()

Module Contents

`class robotlibrary.robot.`**`Robot`**(*rc*)

This is the central class which manages and uses all the other components of the robot. The parameters are defined in config.py

`speed = 0`

`new_speed = 0`

`last_turn_right`

`_drive`(*dir_l*, *dir_r*)

This abstracted driving function is only called locally by the other functions with better names. It accelerates and decelerates to make driving more natural. Do not call directly!!

`_drive_instantly`(*dir_l*, *dir_r*)

This abstracted driving function is only called locally by the other functions with better names. It sets the speed immediatly. Do not call directly!!

`set_speed_instantly`(*s*)

Sets the new speed immediately. Doesn't change the driving mode of the robot.

`set_speed`(*s*)

Sets the new speed and accelerates and decelerates. Doesn't change the driving mode of the robot.

`set_forward`(*f*)

Sets the direction of the robot. True means forward.

`spin_right`()

Spin right indefinitely.

`spin_left`()

Spin left indefinitely.

`turn_right`()

This turns the robot to the right without it spinning on the spot. Each call makes the turn steeper.

turn_left()

This turns the robot to the right without it spinning on the spot. Each call makes the turn steeper.

go_straight()

Lets the robot go straight on. Usually called when a turn shall end.

spin_before_obstacle(*distance*)

This spins until the distance to an obstacle is greater than the given parameter __distance__.

toggle_spin(*d*)

Toggle turn for the given duration. With each call the oppsoite direction(clockwise / anti-clockwise) is used.

random_spin(*d*)

Randomly turn for the given duration.

stop()

Stop the robot slowly by deceleration.

emergency_stop()

Stop the robot immediately.

ir_detected(*pin*, *pin_num*)

If implemented this method is called when the IR-sensor has detected a change. Fill in your code accordingly

get_dist()

Get the distance from the ultrasonic sensor.

set_angle(*a*)

If implemented, turn the servo motor with the ultrasonic sensor to the given angle.

get_smallest_distance()

This returns the angle of the ultrasonic sensor where it measured the smallest distance

get_longest_distance()

This returns the angle of the ultrasonic sensor where it measured the longest distance

robotlibrary.robot.**main**()

robotlibrary.rotary

Classes

Rotary	This class deals with the rotary encoders for the remote control. Don't use directly or edit.

Module Contents

class robotlibrary.rotary.**Rotary**(*dt*, *clk*, *sw*, *rc*)

This class deals with the rotary encoders for the remote control. Don't use directly or edit.

ROT_CW = 1

ROT_CCW = 2

SW_PRESS = 4

SW_RELEASE = 8

dt_pin

clk_pin

sw_pin

rc

last_status

last_button_status

last_change

rotary_change(*pin*)

switch_detect(*pin*)

robotlibrary.servo

Classes

Servo	This class manages the servo motor that turns the ultrasonic sensor. You need a servo motor installed to get use out of this.

Module Contents

class robotlibrary.servo.**Servo**(*pin*)

This class manages the servo motor that turns the ultrasonic sensor. You need a servo motor installed to get use out of this. Don't use directly or edit.

pin

min = 1350

max = 8100

angle = 0

set_angle(*a*)

If installed, the servor motor will set the angle of the ultrasonic sensor. 90° ist straight ahead.

_get_duty(*angle*)

Internal function. Calculates the PWM duty for the given angle.

robotlibrary.ultrasonic

Classes

Ultra	This class manages the ultrasonic sensor. It returns the distance to an obstacle in cm.

Module Contents

class robotlibrary.ultrasonic.**Ultra**(*pinNo*)

This class manages the ultrasonic sensor. It returns the distance to an obstacle in cm.

trigger

echo

get_dist()

This returns the measured distance in cm. (float)

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

r

robotlibrary, 3
robotlibrary.bluetooth, 3
robotlibrary.bluetooth.advertising, 3
robotlibrary.bluetooth.ble_flags, 4
robotlibrary.bluetooth.ble_services_definitions, 8
robotlibrary.bluetooth.central, 9
robotlibrary.bluetooth.main_central, 10
robotlibrary.bluetooth.main_peripheral, 11
robotlibrary.bluetooth.message_parser, 11
robotlibrary.bluetooth.motor, 11
robotlibrary.bluetooth.parser, 12
robotlibrary.bluetooth.peripheral, 12
robotlibrary.bluetooth.pin_map, 13
robotlibrary.conf, 14
robotlibrary.config, 15
robotlibrary.infrared, 16
robotlibrary.joystick, 17
robotlibrary.motor, 18
robotlibrary.rc, 18
robotlibrary.robot, 20
robotlibrary.rotary, 21
robotlibrary.servo, 22
robotlibrary.ultrasonic, 23

INDEX

Symbols

_ADV_MAX_PAYLOAD (*in module robotlibrary.bluetooth.advertising*), 4

_ble_irq_dict (*robotlibrary.bluetooth.peripheral.BLEPeripheral attribute*), 13

_callback (*robotlibrary.bluetooth.motor.Motor attribute*), 12

_connections (*robotlibrary.bluetooth.central.BLECentral attribute*), 10

_connections (*robotlibrary.bluetooth.peripheral.BLEPeripheral attribute*), 13

_drive() (*robotlibrary.robot.Robot method*), 20

_drive_instantly() (*robotlibrary.robot.Robot method*), 20

_get_duty() (*robotlibrary.servo.Servo method*), 23

_handle_characteristics() (*robotlibrary.bluetooth.central.BLECentral method*), 10

_handle_connect() (*robotlibrary.bluetooth.central.BLECentral method*), 10

_handle_connect() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13

_handle_disconnect() (*robotlibrary.bluetooth.central.BLECentral method*), 10

_handle_disconnect() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13

_handle_read() (*robotlibrary.bluetooth.central.BLECentral method*), 10

_handle_read() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13

_handle_scan() (*robotlibrary.bluetooth.central.BLECentral method*), 10

_handle_services() (*robotlibrary.bluetooth.central.BLECentral method*), 10

_handles (*robotlibrary.bluetooth.central.BLECentral attribute*), 10

_handles (*robotlibrary.bluetooth.peripheral.BLEPeripheral attribute*), 13

_irq() (*robotlibrary.bluetooth.central.BLECentral method*), 10

_irq() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13

_irq_dict (*robotlibrary.bluetooth.central.BLECentral attribute*), 10

_moving_forwards (*robotlibrary.bluetooth.motor.Motor attribute*), 12

_on_service_discovery_complete() (*robotlibrary.bluetooth.central.BLECentral method*), 10

_payload (*robotlibrary.bluetooth.peripheral.BLEPeripheral attribute*), 13

_read_callbacks (*robotlibrary.bluetooth.central.BLECentral attribute*), 10

_read_callbacks (*robotlibrary.bluetooth.peripheral.BLEPeripheral attribute*), 13

_read_cb (*robotlibrary.bluetooth.central.BLECentral attribute*), 10

_service_to_scan (*robotlibrary.bluetooth.central.BLECentral attribute*), 10

_speed (*robotlibrary.bluetooth.motor.Motor attribute*), 12

_to_connect_name (*robotlibrary.bluetooth.central.BLECentral attribute*), 10

A

ADV_DIRECT_IND (*in module robotlibrary.bluetooth.ble_flags*), 8

ADV_IND (*in module robotlibrary.bluetooth.ble_flags*), 8

ADV_NONCONN_IND (*in module robotli-

brary.bluetooth.ble_flags), 8
ADV_SCAN_IND (*in module robotlibrary.bluetooth.ble_flags*), 8
ADV_TYPE_APPEARANCE (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_FLAGS (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_MANUFACTURER_DATA (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_NAME (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_UUID128_COMPLETE (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_UUID128_MORE (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_UUID16_COMPLETE (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_UUID16_MORE (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_UUID32_COMPLETE (*in module robotlibrary.bluetooth.ble_flags*), 6
ADV_TYPE_UUID32_MORE (*in module robotlibrary.bluetooth.ble_flags*), 6
advertise() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13
advertising_payload() (*in module robotlibrary.bluetooth.advertising*), 4
angle (*robotlibrary.servo.Servo attribute*), 22
author (*in module robotlibrary.conf*), 14
autoapi_dirs (*in module robotlibrary.conf*), 15

B

b (*robotlibrary.joystick.Joystick attribute*), 17
BLACK_DETECTED (*in module robotlibrary.config*), 16
ble (*robotlibrary.bluetooth.central.BLECentral attribute*), 10
ble (*robotlibrary.bluetooth.peripheral.BLEPeripheral attribute*), 13
BLECentral (*class in robotlibrary.bluetooth.central*), 10
BLEPeripheral (*class in robotlibrary.bluetooth.peripheral*), 13
button() (*robotlibrary.rc.RC method*), 19
button_handler() (*robotlibrary.joystick.Joystick method*), 17

C

change (*robotlibrary.rc.RC attribute*), 19
change_speed() (*robotlibrary.motor.Motor method*), 18
clk_pin (*robotlibrary.rotary.Rotary attribute*), 22
copyright (*in module robotlibrary.conf*), 14

D

DEBOUNCE_WAIT (*in module robotlibrary.config*), 16
decode_field() (*in module robotlibrary.bluetooth.advertising*), 4
decode_motor() (*in module robotlibrary.bluetooth.message_parser*), 11
decode_motor() (*in module robotlibrary.bluetooth.parser*), 12
decode_services() (*in module robotlibrary.bluetooth.advertising*), 4
detected (*robotlibrary.infrared.IR attribute*), 17
dt_pin (*robotlibrary.rotary.Rotary attribute*), 22
duty_cycle (*robotlibrary.rc.RC attribute*), 19

E

echo (*robotlibrary.ultrasonic.Ultra attribute*), 23
emergency_stop() (*robotlibrary.robot.Robot method*), 21
encode_motor() (*in module robotlibrary.bluetooth.message_parser*), 11
encode_motor() (*in module robotlibrary.bluetooth.parser*), 12
exclude_patterns (*in module robotlibrary.conf*), 15
extensions (*in module robotlibrary.conf*), 14

F

FLAG_NOTIFY (*in module robotlibrary.bluetooth.ble_flags*), 8
FLAG_READ (*in module robotlibrary.bluetooth.ble_flags*), 8
FLAG_WRITE (*in module robotlibrary.bluetooth.ble_flags*), 8
FLAG_WRITE_NO_RESPONSE (*in module robotlibrary.bluetooth.ble_flags*), 8
forward (*robotlibrary.motor.Motor attribute*), 18
forward (*robotlibrary.rc.RC attribute*), 19

G

GATTS_ERROR_INSUFFICIENT_AUTHENTICATION (*in module robotlibrary.bluetooth.ble_flags*), 8
GATTS_ERROR_INSUFFICIENT_AUTHORIZATION (*in module robotlibrary.bluetooth.ble_flags*), 8
GATTS_ERROR_INSUFFICIENT_ENCRYPTION (*in module robotlibrary.bluetooth.ble_flags*), 8
GATTS_ERROR_READ_NOT_PERMITTED (*in module robotlibrary.bluetooth.ble_flags*), 8
GATTS_ERROR_WRITE_NOT_PERMITTED (*in module robotlibrary.bluetooth.ble_flags*), 8
GATTS_NO_ERROR (*in module robotlibrary.bluetooth.ble_flags*), 7
get_direction() (*robotlibrary.joystick.Joystick method*), 18
get_dist() (*robotlibrary.robot.Robot method*), 21
get_dist() (*robotlibrary.ultrasonic.Ultra method*), 23
get_longest_distance() (*robotlibrary.robot.Robot method*), 21

get_smallest_distance() (*robotlibrary.robot.Robot method*), 21
get_speed() (*robotlibrary.bluetooth.motor.Motor method*), 12
get_speed() (*robotlibrary.joystick.Joystick method*), 18
go_straight() (*robotlibrary.robot.Robot method*), 21
gpio (*robotlibrary.motor.Motor attribute*), 18

H

html_static_path (*in module robotlibrary.conf*), 15
html_theme (*in module robotlibrary.conf*), 15

I

INFRARED_UUID (*in module robotlibrary.bluetooth.ble_services_definitions*), 9
IR (*class in robotlibrary.infrared*), 17
IR (*in module robotlibrary.config*), 16
ir (*robotlibrary.infrared.IR attribute*), 17
ir_detected() (*robotlibrary.robot.Robot method*), 21
IRQ_CENTRAL_CONNECT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_CENTRAL_DISCONNECT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_CONNECTION_UPDATE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_ENCRYPTION_UPDATE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_CHARACTERISTIC_DONE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_CHARACTERISTIC_RESULT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_DESCRIPTOR_DONE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_DESCRIPTOR_RESULT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_INDICATE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_NOTIFY (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_READ_DONE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_READ_RESULT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_SERVICE_DONE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_SERVICE_RESULT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTC_WRITE_DONE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTS_INDICATE_DONE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTS_READ_REQUEST (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GATTS_WRITE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_GET_SECRET (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_L2CAP_ACCEPT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_L2CAP_CONNECT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_L2CAP_DISCONNECT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_L2CAP_RECV (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_L2CAP_SEND_READY (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_MTU_EXCHANGED (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_PERIPHERAL_CONNECT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_PERIPHERAL_DISCONNECT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_SCAN_DONE (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_SCAN_RESULT (*in module robotlibrary.bluetooth.ble_flags*), 7
IRQ_SET_SECRET (*in module robotlibrary.bluetooth.ble_flags*), 7
is_connected() (*robotlibrary.bluetooth.central.BLECentral method*), 10
is_connected() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13

J

Joystick (*class in robotlibrary.joystick*), 17
joystick (*in module robotlibrary.joystick*), 18
JS_MAX_DUTY (*in module robotlibrary.config*), 16
JS_MIN_DUTY (*in module robotlibrary.config*), 16
JS_X_MEDIAN (*in module robotlibrary.config*), 16
JS_Y_MEDIAN (*in module robotlibrary.config*), 16

L

last_button_status (*robotlibrary.rotary.Rotary attribute*), 22
last_change (*robotlibrary.rotary.Rotary attribute*), 22
last_pressed (*robotlibrary.joystick.Joystick attribute*), 17
last_status (*robotlibrary.rotary.Rotary attribute*), 22
last_turn_right (*robotlibrary.robot.Robot attribute*), 20

M

main() (*in module robotlibrary.bluetooth.main_central*), 11

main() (*in module robotlibrary.bluetooth.main_peripheral*), 11
main() (*in module robotlibrary.rc*), 19
main() (*in module robotlibrary.robot*), 21
max (*robotlibrary.servo.Servo attribute*), 22
MAX_DUTY (*in module robotlibrary.config*), 16
MAX_SPEED (*in module robotlibrary.config*), 16
MAX_SPEED (*robotlibrary.bluetooth.motor.Motor attribute*), 12
min (*robotlibrary.servo.Servo attribute*), 22
MIN_DUTY (*in module robotlibrary.config*), 16
MIN_SPEED (*in module robotlibrary.config*), 16
ML (*in module robotlibrary.config*), 16
module
- robotlibrary, 3
- robotlibrary.bluetooth, 3
- robotlibrary.bluetooth.advertising, 3
- robotlibrary.bluetooth.ble_flags, 4
- robotlibrary.bluetooth.ble_services_definitions, 8
- robotlibrary.bluetooth.central, 9
- robotlibrary.bluetooth.main_central, 10
- robotlibrary.bluetooth.main_peripheral, 11
- robotlibrary.bluetooth.message_parser, 11
- robotlibrary.bluetooth.motor, 11
- robotlibrary.bluetooth.parser, 12
- robotlibrary.bluetooth.peripheral, 12
- robotlibrary.bluetooth.pin_map, 13
- robotlibrary.conf, 14
- robotlibrary.config, 15
- robotlibrary.infrared, 16
- robotlibrary.joystick, 17
- robotlibrary.motor, 18
- robotlibrary.rc, 18
- robotlibrary.robot, 20
- robotlibrary.rotary, 21
- robotlibrary.servo, 22
- robotlibrary.ultrasonic, 23

Motor (*class in robotlibrary.bluetooth.motor*), 12
Motor (*class in robotlibrary.motor*), 18
MOTOR_LEFT_BACKWARD (*in module robotlibrary.bluetooth.pin_map*), 14
MOTOR_LEFT_FORWARD (*in module robotlibrary.bluetooth.pin_map*), 14
MOTOR_RIGHT_BACKWARD (*in module robotlibrary.bluetooth.pin_map*), 14
MOTOR_RIGHT_FORWARD (*in module robotlibrary.bluetooth.pin_map*), 14
MOTOR_RX (*in module robotlibrary.bluetooth.ble_services_definitions*), 9
MOTOR_RX_UUID (*in module robotlibrary.bluetooth.ble_services_definitions*), 9
MOTOR_TX (*in module robotlibrary.bluetooth.ble_services_definitions*), 9
MOTOR_TX_UUID (*in module robotlibrary.bluetooth.ble_services_definitions*), 9
moving_forwards() (*robotlibrary.bluetooth.motor.Motor method*), 12
MR (*in module robotlibrary.config*), 16

N

new_speed (*robotlibrary.robot.Robot attribute*), 20

O

obstacle() (*robotlibrary.infrared.IR method*), 17
off() (*robotlibrary.motor.Motor method*), 18
out (*robotlibrary.infrared.IR attribute*), 17

P

p (*robotlibrary.rc.RC attribute*), 19
PASSKEY_ACTION_DISPLAY (*in module robotlibrary.bluetooth.ble_flags*), 8
PASSKEY_ACTION_INPUT (*in module robotlibrary.bluetooth.ble_flags*), 8
PASSKEY_ACTION_NONE (*in module robotlibrary.bluetooth.ble_flags*), 8
PASSKEY_ACTION_NUMERIC_COMPARISON (*in module robotlibrary.bluetooth.ble_flags*), 8
pin (*robotlibrary.servo.Servo attribute*), 22
pin_map (*in module robotlibrary.bluetooth.pin_map*), 14
pressed (*robotlibrary.joystick.Joystick attribute*), 17
project (*in module robotlibrary.conf*), 14
pwm1 (*robotlibrary.motor.Motor attribute*), 18
pwm2 (*robotlibrary.motor.Motor attribute*), 18

R

random_spin() (*robotlibrary.robot.Robot method*), 21
RC (*class in robotlibrary.rc*), 19
rc (*robotlibrary.rotary.Rotary attribute*), 22
read() (*in module robotlibrary.bluetooth.main_central*), 11
read() (*robotlibrary.rc.RC method*), 19
register_irq() (*robotlibrary.bluetooth.central.BLECentral method*), 10
register_irq() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13
register_read_callback() (*robotlibrary.bluetooth.central.BLECentral method*), 10

register_read_callback() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13
release (*in module robotlibrary.conf*), 14
reset() (*robotlibrary.joystick.Joystick method*), 17
reset_detected() (*robotlibrary.infrared.IR method*), 17
reset_offset() (*robotlibrary.motor.Motor method*), 18
Robot (*class in robotlibrary.robot*), 20
robot (*robotlibrary.infrared.IR attribute*), 17
ROBOT_NAME (*in module robotlibrary.config*), 16
ROBOT_SERVICE (*in module robotlibrary.bluetooth.ble_services_definitions*), 9
ROBOT_UUID (*in module robotlibrary.bluetooth.ble_services_definitions*), 9
robotlibrary
module, 3
robotlibrary.bluetooth
module, 3
robotlibrary.bluetooth.advertising
module, 3
robotlibrary.bluetooth.ble_flags
module, 4
robotlibrary.bluetooth.ble_services_definitions
module, 8
robotlibrary.bluetooth.central
module, 9
robotlibrary.bluetooth.main_central
module, 10
robotlibrary.bluetooth.main_peripheral
module, 11
robotlibrary.bluetooth.message_parser
module, 11
robotlibrary.bluetooth.motor
module, 11
robotlibrary.bluetooth.parser
module, 12
robotlibrary.bluetooth.peripheral
module, 12
robotlibrary.bluetooth.pin_map
module, 13
robotlibrary.conf
module, 14
robotlibrary.config
module, 15
robotlibrary.infrared
module, 16
robotlibrary.joystick
module, 17
robotlibrary.motor
module, 18
robotlibrary.rc
module, 18
robotlibrary.robot
module, 20
robotlibrary.rotary
module, 21
robotlibrary.servo
module, 22
robotlibrary.ultrasonic
module, 23
ROT_CCW (*robotlibrary.rotary.Rotary attribute*), 22
ROT_CW (*robotlibrary.rotary.Rotary attribute*), 22
Rotary (*class in robotlibrary.rotary*), 22
rotary_bottom (*robotlibrary.rc.RC attribute*), 19
rotary_change() (*robotlibrary.rotary.Rotary method*), 22
rotary_changed() (*robotlibrary.rc.RC method*), 19
rotary_top (*robotlibrary.rc.RC attribute*), 19

S

scan() (*robotlibrary.bluetooth.central.BLECentral method*), 10
SCAN_RSP (*in module robotlibrary.bluetooth.ble_flags*), 8
send() (*robotlibrary.bluetooth.central.BLECentral method*), 10
send() (*robotlibrary.bluetooth.peripheral.BLEPeripheral method*), 13
send() (*robotlibrary.rc.RC method*), 19
send_timer (*robotlibrary.rc.RC attribute*), 19
server (*robotlibrary.rc.RC attribute*), 19
Servo (*class in robotlibrary.servo*), 22
SERVO (*in module robotlibrary.config*), 16
set_angle() (*robotlibrary.robot.Robot method*), 21
set_angle() (*robotlibrary.servo.Servo method*), 22
set_direction() (*robotlibrary.bluetooth.motor.Motor method*), 12
set_forward() (*robotlibrary.motor.Motor method*), 18
set_forward() (*robotlibrary.robot.Robot method*), 20
set_speed() (*robotlibrary.bluetooth.motor.Motor method*), 12
set_speed() (*robotlibrary.motor.Motor method*), 18
set_speed() (*robotlibrary.rc.RC method*), 19
set_speed() (*robotlibrary.robot.Robot method*), 20
set_speed_instantly() (*robotlibrary.robot.Robot method*), 20
speed (*robotlibrary.motor.Motor attribute*), 18
speed (*robotlibrary.rc.RC attribute*), 19
speed (*robotlibrary.robot.Robot attribute*), 20
speed_offset (*robotlibrary.motor.Motor attribute*), 18
spin_before_obstacle() (*robotlibrary.robot.Robot method*), 21
spin_left() (*robotlibrary.robot.Robot method*), 20
spin_right() (*robotlibrary.robot.Robot method*), 20
stop() (*robotlibrary.robot.Robot method*), 21
sw_pin (*robotlibrary.rotary.Rotary attribute*), 22

SW_PRESS (*robotlibrary.rotary.Rotary attribute*), 22
SW_RELEASE (*robotlibrary.rotary.Rotary attribute*), 22
switch_detect() (*robotlibrary.rotary.Rotary method*), 22

T

templates_path (*in module robotlibrary.conf*), 14
timer (*robotlibrary.infrared.IR attribute*), 17
timer (*robotlibrary.joystick.Joystick attribute*), 17
timer (*robotlibrary.rc.RC attribute*), 19
toggle_spin() (*robotlibrary.robot.Robot method*), 21
trigger (*robotlibrary.ultrasonic.Ultra attribute*), 23
turn_left() (*robotlibrary.robot.Robot method*), 20
turn_right() (*robotlibrary.robot.Robot method*), 20
turn_val (*robotlibrary.rc.RC attribute*), 19

U

Ultra (*class in robotlibrary.ultrasonic*), 23
ULTRASONIC_UUID (*in module robotlibrary.bluetooth.ble_services_definitions*), 9
US (*in module robotlibrary.config*), 16

W

WHITE_DETECTED (*in module robotlibrary.config*), 16

X

x (*robotlibrary.joystick.Joystick attribute*), 17

Y

y (*robotlibrary.joystick.Joystick attribute*), 17