Yahboom Superbit MicroPython API

Note:

The superbit-micropython library is a driver for the superbit expansion board added to the official microbit-microPython library. Other APIs can be found on the microbit-microPython website.
URL: https://microbit-micropython.readthedocs.io/en/latest/

A.Superbit library

Import superbit_micropython library

import super:bit

superbit.motor_control(a, b, 0)

Function:

Control the motor on the expansion board to send PWM

Parameter:

a: Select the corresponding motor port (superbit.M1-M4)

b: PWM duty cycle (-255~255, negative value is reverse, positive value is positive)

superbit.motor_control_dual(a, b, c, d, 0)

Function:

Control two motors on the expansion board to send PWM at the same time

Parameter:

a: Select the corresponding motor1 port (superbit.M1-M4)

b: Select the corresponding motor2 port (superbit.M1-M4)

c: PWM duty cycle of motor1(-255~255, negative value is reverse, positive value is positive)

d: PWM duty cycle of motor2(-255~255, negative value is reverse, positive value is positive)

superbit.servo270(a, b)

Function:

Control the control the 270 ° servo on servo interface on the expansion board

Parameter:

a: Select the servo port number (superbit.S1-S8)

b: Set the control angle (0-270)

superbit.servo180(a, b)

Function:

Control the control the 180 ° servo on servo interface on the expansion board

Parameter:

a: Select the servo port number (superbit.S1-S8)

b: Set the control angle (0-180)

superbit.stepper_control(a, b)

Function:

Controlling stepper motors on expansion boards

Parameter:

a: Select the servo port number（superbit.B1-B2）

b: Set the control angle (0-360)

B. ghandle library

1) import ghandle

Import handle library

2) ghandle.B1_is_pressed()

Function: Button B1 (red), press to return to True, release to return to False

3) ghandle.B2_is_pressed()

Function: Button B2 (green), press to return to True, release to return to False

4) ghandle.B3_is_pressed()

Function: Button B3 (blue), press to return to True, release to return to False

5) ghandle.B4_is_pressed()

Function: Button B4 (yellow), press to return to True, release to return to False

6) ghandle.rocker(state)

Function: Detect the direction of the rocker, if consistent with the parameter, return True. If inconsistent with the parameter, return False

Parameters:

state: ghandle.up (up), ghandle.down (down), ghandle.left (left), ghandle.right (right), ghandle.pressed (pressed), ghandle.noState (no operation)

7) ghandle.get_value_x()

Function: return to the analog value of the X channel of the rocker. Move to the left to increase the value. Move to the right to decrease the value.

8) ghandle.get_value_y()

Function: return to the analog value of the Y channel of the rocker. Move to the down to increase the value. Move to the up to decrease the value.