**Yahboom Building:bit MicroPython API**

Input this code to import Yahboom building:bit Micropython library：**import buildingbit**

1、**Car advance** **buildingbit.car\_run(L\_Speed, R\_Speed, delay)**

Description of Parameter:

L\_speed is left motor speed,range is 0-255.

R\_speed is right motor speed,range is 0-255.

delay is delay time(ms)

Eg: buildingbit.car\_run(255, 255, 1000) // The car will advance 1000ms with 255 speed

2、**Car back** **buildingbit.car\_back(L\_Speed, R\_Speed, delay)**

Description of Parameter:

L\_speed is left motor speed,range is 0-255.

R\_speed is right motor speed,range is 0-255.

delay is delay time(ms)

Eg: buildingbit.car\_back(255, 255, 1000) // The car will back 1000ms with 255 speed

3、**Car turn left** **buildingbit.car\_left(Speed, delay)**

Description of Parameter:

speed is right motor speed,range is 0-255.

delay is delay time(ms)

!Note: car turn left is left wheel stop and right wheel advance,so we only input right speed.

Eg: buildingbit.car\_left(255, 1000) // The car will turn left 1000ms with 255 speed

4、**Car turn right** **buildingbit.car\_right(Speed, delay)**

Description of Parameter:

speed is left motor speed,range is 0-255.

delay is delay time(ms)

!Note: car turn right is right wheel stop and left wheel advance,so we only input left speed.

Eg: buildingbit.car\_right(255, 1000) // The car will turn right 1000ms with 255 speed

5、**Car spin left**  **buildingbit.car\_spinleft(L\_Speed, R\_Speed, delay)**

Description of Parameter:

L\_speed is left motor speed,range is 0-255.

R\_speed is right motor speed,range is 0-255.

delay is delay time(ms)

Eg: buildingbit.car\_spinleft(255, 255, 1000) // The car will spin left 1000ms with 255 speed

6、**Car spin right** **buildingbit.car\_spinright(L\_Speed, R\_Speed, delay)**

Description of Parameter:

L\_speed is left motor speed,range is 0-255.

R\_speed is right motor speed,range is 0-255.

delay is delay time(ms)

Eg: buildingbit.car\_spinright(255, 255, 1000) // The car will spin right 1000ms with 255 speed

7、**Car stop** **buildingbit.car\_stop()**

8、**RGB search light**  **buildingbit.car\_HeadRGB(R\_value, G\_value, B\_value)**

Description of Parameter:

R\_value is red value,range is 0-255.

G\_value is green value,range is 0-255.

B\_value is blue value,range is 0-255.

Eg: buildingbit.car\_HeadRGB(255, 255, 255) // The RGB search light will become white

9、**Ultrasonic module return distance** **buildingbit.ultrasonic()**

This function returns the current distance detected by the ultrasonic module(cm).

10、**Infrared obstacle avoidance**  **buildingbit.avoid\_sensor()**

Return value: if there are obstacles returns true, if there are no obstacles return false

! Note: When using this function, you must turn off the dot-matrix display, otherwise pin reuse will cause conflicts, and the method of turning off dot-matrix display: **microbit.display.off ()**

11、**Left tracking sensor** **buildingbit.traking\_sensor\_L()**

Return value: if black lines are detected returns true(Indicator light is off), if white lines are detected return false(Indicator light is on).

12、**Left tracking sensor** **buildingbit.traking\_sensor\_R()**

Return value: if black lines are detected returns true(Indicator light is off), if white lines are detected return false(Indicator light is on).

13、**Servo control**  **buildingbit.servo(num, angle)**

Description of Parameter:

num is number of servo(1-3)

angle is angle of servo,range is 0-180.

Eg: buildingbit.servo(1, 180) // No.1 servo will rotate to 180°

14、**IR control buildingbit.init\_IR(IR\_pin)**

Description of Parameter:

IR\_pin is Pins connected to the infrared receiver. **For building:bit, we need to select pin8**

Eg: buildingbit.init\_IR(pin8)

15、**Return button value of IR controller** **buildingbit.get\_IR(IR\_pin)**

Description of Parameter:

IR\_pin is Pins connected to the infrared receiver. **For building:bit, we need to select pin8**

Eg: buildingbit.get\_IR(pin8)

