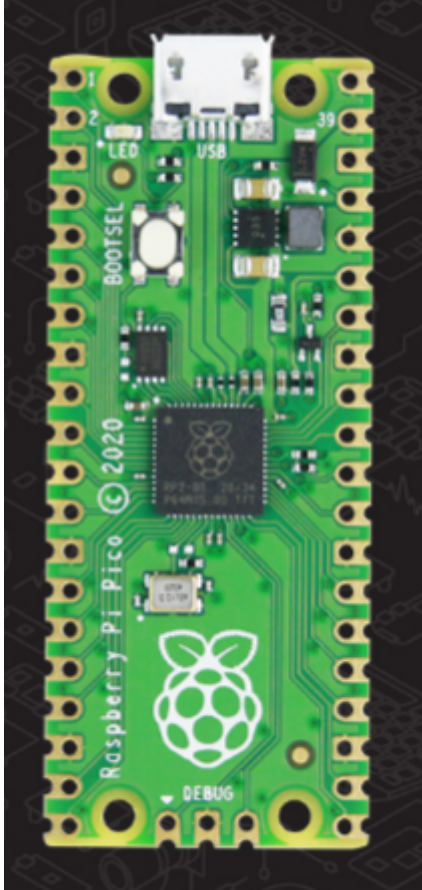


Using of 270° Servo --- Raspebrry Pi Pico

1.Preparation

Raspebrry Pi Pico board *1



270° servo *1



7.4V battery *1

2.Purpose

Control the 270° steering gear rotation (0° --90° -270°) , then keep cycle.

3.Principle

The working principle of the servo: the control signal enters the signal modulation chip from the channel of the receiver to obtain the bias voltage of the DC. It has a reference circuit inside, which generates a reference signal with a period of 20ms and a width of 1.5ms. It will compares the DC bias voltage with the voltage of the potentiometer to obtain a voltage difference and output. The positive and negative of the voltage difference is outputted to the motor drive chip to determine the forward and reverse of the motor. Servo rotation angle is by adjusting the duty ratios of PWM (pulse width modulation) signal. The standard PWM (pulse width modulation) signal has a fixed period of 20ms (50Hz). Theoretically, pulse width distribution should be between 1 ms to 2 ms, but in fact between pulse width can be 0.5 ms and 2.5 ms. Pulse width and the servo rotation angle 0° ~ 180° corresponds, as shown below.

0.5ms-----0°
1.0ms-----45°
1.5ms-----90°
2.0ms-----135°
2.5ms-----180°

4.Wiring

1. The red wire (positive pole) of the steering gear is connected to the positive pole of the 7.4V battery
2. The brown wire (negative pole) of the steering gear is connected to the negative pole of the 7.4V battery
3. The yellow wire (signal wire) of the steering gear is connected to the GP7 pin of the PICO main board
4. The PICO board is powered by the computer, and any GND interface on the board is connected to the negative pole of the battery

5.Code

1.Before writing the code, we need to copy the ulf file to PICO, and then power on the Pico board again.

2.Code as shown below.

```
# PWM steering gear swings back and forth before 0 and 270 degrees.
while True:
    ... servo_control(0)
    ... utime.sleep(1)
    ... servo_control(90)
    ... utime.sleep(1)
    ... servo_control(270)
    ... utime.sleep(1)
```

