

# Using PWM

## Using PWM

1. Test code
2. Result

The Hobot-GPIO library only supports PWM on pins with additional hardware PWM controllers. Compared to RPi The GPIO library is different, Hobot-GPIO library does not implement software simulation PWM. Both RDK X3 and RDK Ultra support 2 PWM channels.

Please refer to `/app/40pin_stamples/sample_pwm.py` for detailed information on how to use PWM channels.

## 1. Test code

Open the PWM channel specified by `output_pin`, with an initial duty cycle of 25%.

First, increase the duty cycle by 5% every 0.25 seconds, and then reduce the duty cycle by 5% every 0.25 seconds after reaching 100%.

When the waveform is output normally, you can measure the output signal with an oscilloscope or logic analyzer to observe the waveform.

```
#!/usr/bin/env python3

import Hobot.GPIO as GPIO
import time

# PWM supported pins: 32 and 33
output_pin = 33

def main():
    # Pin Setup:
    # Board pin-numbering scheme
    GPIO.setmode(GPIO.BOARD)
    # Frequency range supported by RDK X3: 48KHz ~ 192MHz
    # Frequency range supported by RDK Ultra: 1Hz ~ 12MHz
    p = GPIO.PWM(output_pin, 48000)
    # Initial duty cycle is 25%. First, increase the duty cycle by 5% every 0.25
    seconds. After reaching 100%, decrease the duty cycle by 5% every 0.25 seconds.
    val = 25
    incr = 5
    p.start(val)

    print("PWM running. Press CTRL+C to exit.")
    try:
        while True:
            time.sleep(0.25)
            if val >= 100:
                incr = -incr
            if val <= 0:
```

```

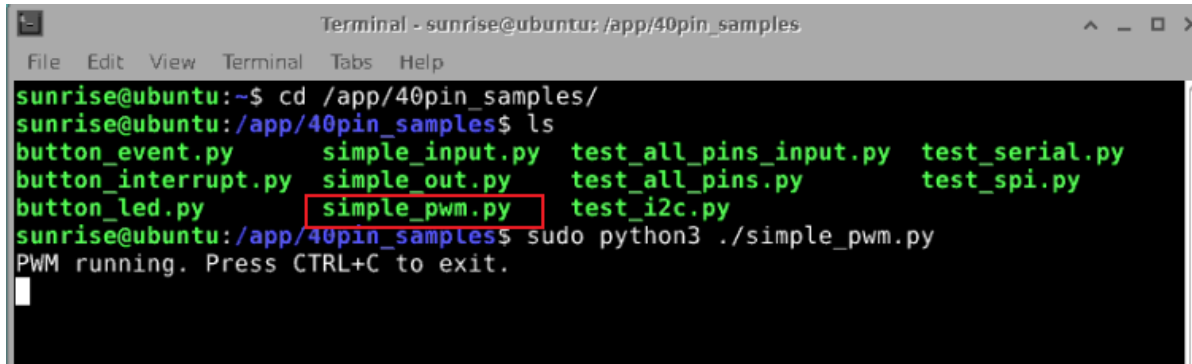
        incr = -incr
        val += incr
        p.ChangeDutyCycle(val)
    finally:
        p.stop()
        GPIO.cleanup()

if __name__ == '__main__':
    main()

```

## 2. Result

The successful running code is as shown below.



```

Terminal - sunrise@ubuntu: /app/40pin_samples
File Edit View Terminal Tabs Help
sunrise@ubuntu:~$ cd /app/40pin_samples/
sunrise@ubuntu:/app/40pin_samples$ ls
button_event.py      simple_input.py      test_all_pins_input.py  test_serial.py
button_interrupt.py  simple_out.py        test_all_pins.py        test_spi.py
button_led.py        simple_pwm.py        test_i2c.py
sunrise@ubuntu:/app/40pin_samples$ sudo python3 ./simple_pwm.py
PWM running. Press CTRL+C to exit.

```

Using a logic analyzer or oscilloscope, you can measure the output signal and observe the waveform.

