

PWM 与伺服舵机

CanMV K230 翻转课堂

模块一 PWM：波形、接口与板载 LED 实验

模块二 舵机：脉宽、反馈与 MG995 实验

PWM：周期、频率和占空比

周期 T

高低电平重复一次
所需的时间

频率 f

每秒重复的次数
 $f = 1 / T$

占空比 D

高电平时间所占比例
 $D = t_{\text{高}} / T$

同一条 PWM 波形

高电平 $t_{\text{高}}$



一个周期 T

例： $f = 50 \text{ Hz} \rightarrow T = 20 \text{ ms}$; $t_{\text{高}} = 1.5 \text{ ms} \rightarrow D = 7.5\%$

硬件 PWM：从 Python 到引脚电平

01 Python 程序

`PWM(0, freq=1000, duty=50)`



02 驱动与寄存器

把频率、占空比写入硬件



03 PWM 硬件

依据时钟持续计数、切换高低电平



04 FGPIOA 与 IO

把 PWM0 输出映射到 GPIO42

硬件 PWM 在 sleep 时持续输出；板载蓝灯实验则由 Python 循环切换电平。

K230 / K210: 硬件 PWM API

K230 • CanMV MicroPython

```
fpioa.set_function(42, FPIOA.PWM0)

pwm = PWM(0, freq=1000, duty=50)

pwm.duty(25)

pwm.duty(75)
```

固定 1 kHz; 示例调整占空比

K210 • Standalone SDK (C)

```
fpioa_set_function(10,
    FUNC_TIMER1_TOGGLE1);

pwm_init(TIMER_PWM);

pwm_set_frequency(..., 200000, 0.5);

pwm_set_enable(..., 1);
```

200 kHz; 每 10 ms 调整占空比

参数差异: K230 duty=50 表示 50%; K210 C demo duty=0.5 表示 50%。

板载蓝灯：用软件控制占空比

show_brightness(duty) 核心代码

```
high_us = 50 * duty
low_us = 5000 - high_us
for _ in range(20):
    if high_us > 0:
        led.value(1)
        time.sleep_us(high_us)
    if low_us > 0:
        led.value(0)
        time.sleep_us(low_us)
```

固定名义周期

5000 μ s = 5 ms
名义频率约 200 Hz
实际含执行开销

例：duty = 25

高电平 1250 μ s
低电平 3750 μ s
占空比约 25%

GPIO52 → 板载蓝灯；高电平时间占比增大，LED 平均亮度提高。

现场演示一：板载蓝灯明暗变化

运行前

开发板连接电脑
GPIO52 → 板载蓝灯
运行 LED 演示代码
无需外部接线

运行中

占空比 0% → 100%
再由 100% → 0%
每次变化 5%
每档保持约 0.1 秒
蓝灯渐亮、渐暗

现场讲解

0%：蓝灯熄灭
100%：持续点亮
占空比越大越亮
停止：IDE 停止按钮

软件 PWM：Python 反复写 GPIO，高低电平时间由延时控制。

名义周期 5 ms（约 200 Hz）；实际时序含代码执行与调度开销。

第二部分

伺服舵机 MG995

内部反馈怎样把目标信号变成转动

K230 如何用 50 Hz PWM 指定高电平脉宽

现场观察五档位置的往返运动

MG995：舵机内部怎样到位

控制信号给出目标位置；舵机内部完成位置反馈。

控制电路

读入脉冲宽度

小电机

产生转动力矩

减速齿轮

带动输出轴

位置传感器

报告轴的位置

位置反馈

当前位置与目标位置有偏差 → 继续驱动电机；接近目标 → 减弱驱动。

普通三线舵机不会把实际角度直接回传给 K230。

舵机信号：固定周期，改变脉宽

频率

约 50 Hz

周期

约 20 ms

角度指令

由高电平脉宽表示



短脉冲 → 一个方向



1.5 ms 左右 → 中位附近



长脉冲 → 另一方向

例程映射 0.5 ~ 2.5 ms; MG995 实物端点须在演示前确认。

K230 → MG995：信号与供电

CanMV K230

GPIO42 → PWM0 信号输出

GND → 共地

```
PWM(0, freq=50,  
      duty_ns=0)
```

信号

共地

MG995 与独立电源

信号线 ← GPIO42

地线 ← 开发板与电源 GND

电源线 ← 合适的外部电源

演示前核对

引脚线序、电源规格和 MG995 的无负载可动范围。

K230 / K210: 舵机控制接口

K230 • CanMV Python

```
PWM(0, freq=50, duty_ns=0)
```

```
duty_ns(1500000)
```

直接以 ns 设置高电平时间
50 Hz 对应 20 ms 周期

原 Servo() 的角度换算:

```
int((angle+90)/180  
    *2000000+500000)
```

K210 • PaddlePi C demo

```
pwm_init(TIMER_PWM)
```

```
pwm_set_frequency(..., f, duty)
```

duty 参数是 0 ~ 1 的比例
该 demo 未封装 Servo 类

以同样脉宽为目标的换算:

```
duty = pulse_us / 20000.0
```

同样要输出 50 Hz、1.5 ms 高电平:

K230 可设置 `duty_ns(1500000)`; K210 C 接口换算 $duty = 1.5 / 20 = 0.075$ 。

现场演示二：MG995 五档位置

原例程：50 Hz，依次下达五个目标位置，每档保持 1 秒并循环。

-90°

0.5 ms

-45°

1.0 ms

0°

1.5 ms

45°

2.0 ms

90°

2.5 ms

屏幕与实物同步观察

代码中显示的是目标角度；舵盘转动是实际可见的运行结果。

MG995 先空载核对端点；如行程不符，按实物安全范围校准脉宽。

保留五档、每档 1 秒、循环往返的原例程演示效果。

两个实验：占空比与脉冲宽度

板载蓝灯 · 软件 PWM

名义周期保持 5 ms，占空比由 0% 升至 100% 再降回。

Python 循环切换 GPIO52，观察蓝灯逐渐变亮、变暗。

MG995 位置舵机 · 硬件 PWM

保持频率约 50 Hz，改变每周期的电平脉宽。

脉宽代表目标位置；内部反馈让输出轴靠近目标。

LED 看高电平所占比例；舵机用每次高电平的脉宽表示目标位置。

资料来源与实验文件

01 板载蓝灯软件 PWM 演示

led_soft_pwm.py (用户已跑通) ; GPIO52 依据 01Studio 点亮 LED 教程

02 01Studio K230 舵机例程

配套资料: 4.传感器和模块 / 3.舵机 / servo.py

03 PaddlePi K210 PWM demo

github.com/kendryte/PaddlePi / standalone-demos / pwm / main.c

04 CanMV K230 PWM API

kendryte.com/k230_canmv / PWM Module API Manual

05 MG995 器件资料

MG995 High Speed Metal Gear Servo; 实际端点以手中器件为准