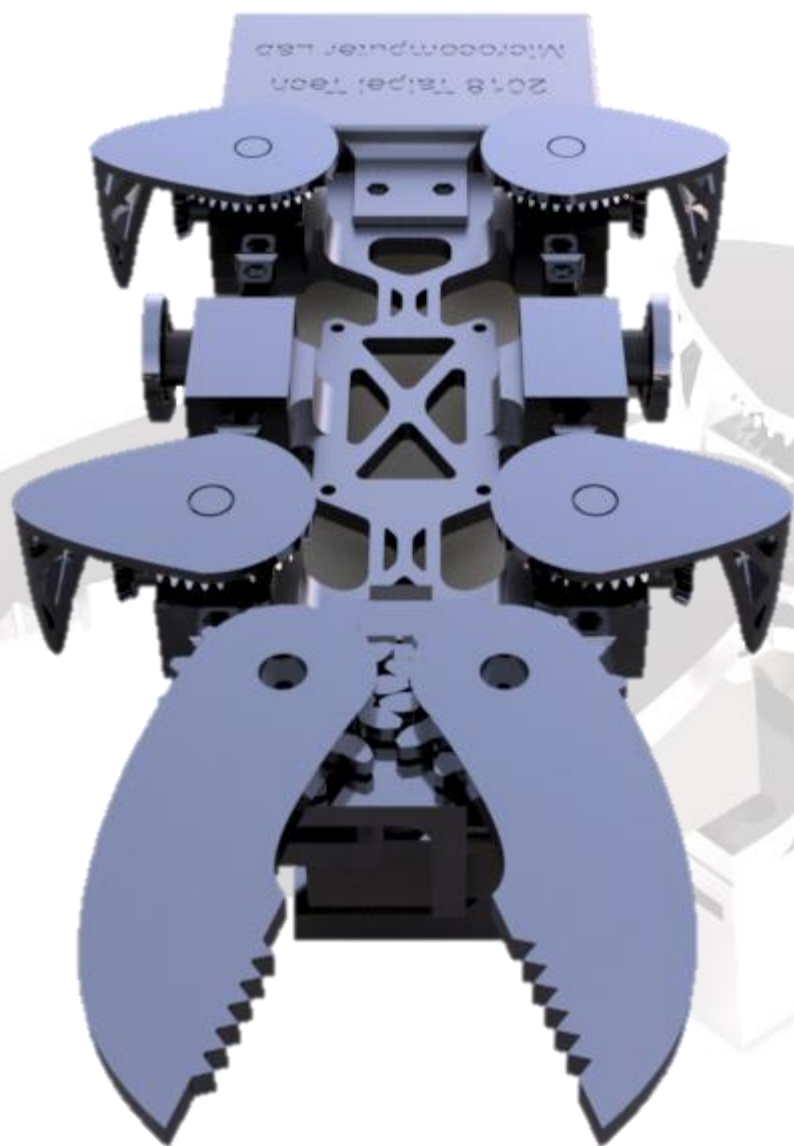


# 微算機原理與應用

106360210劉紹祥

106360218莊詠鈞

106360231黃思齊



機構

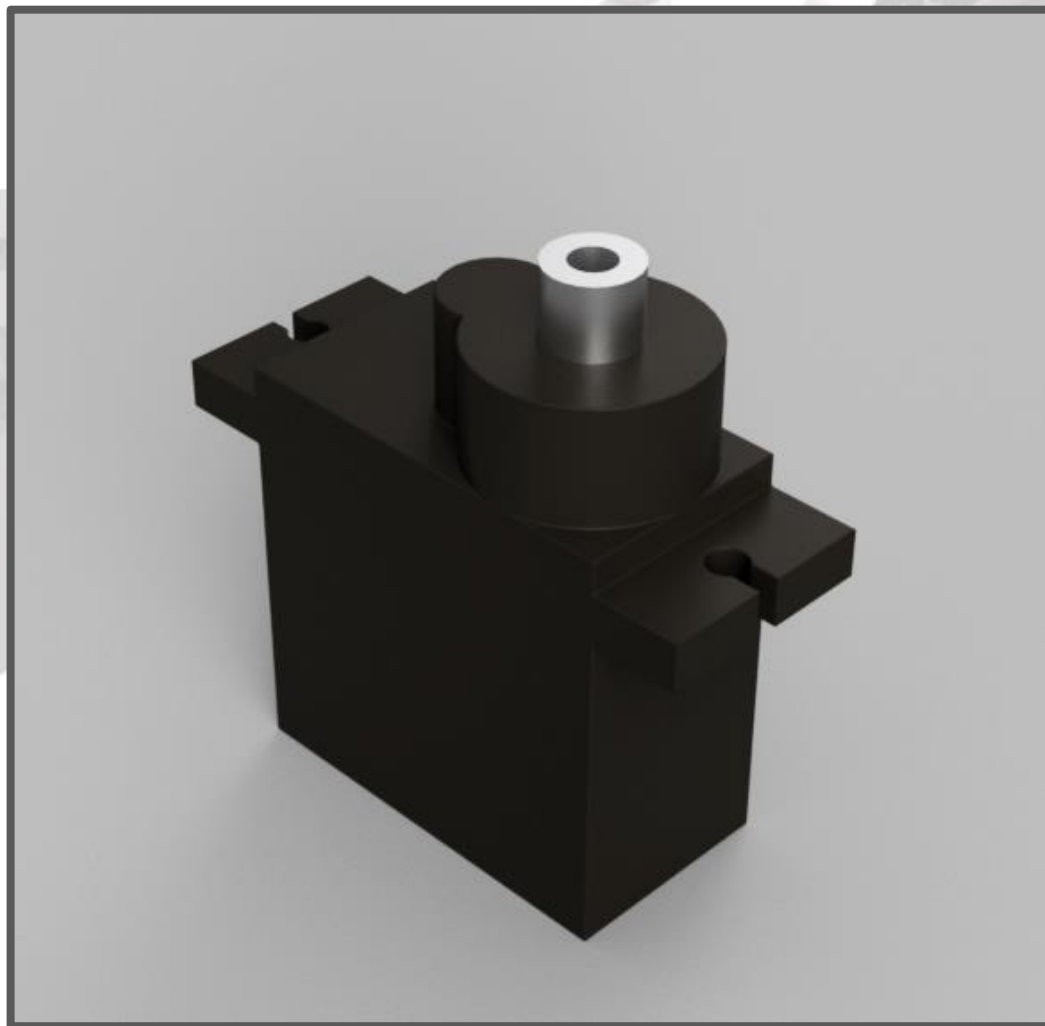
電路

程式

# 機構



# 伺服馬達 MG90



機構

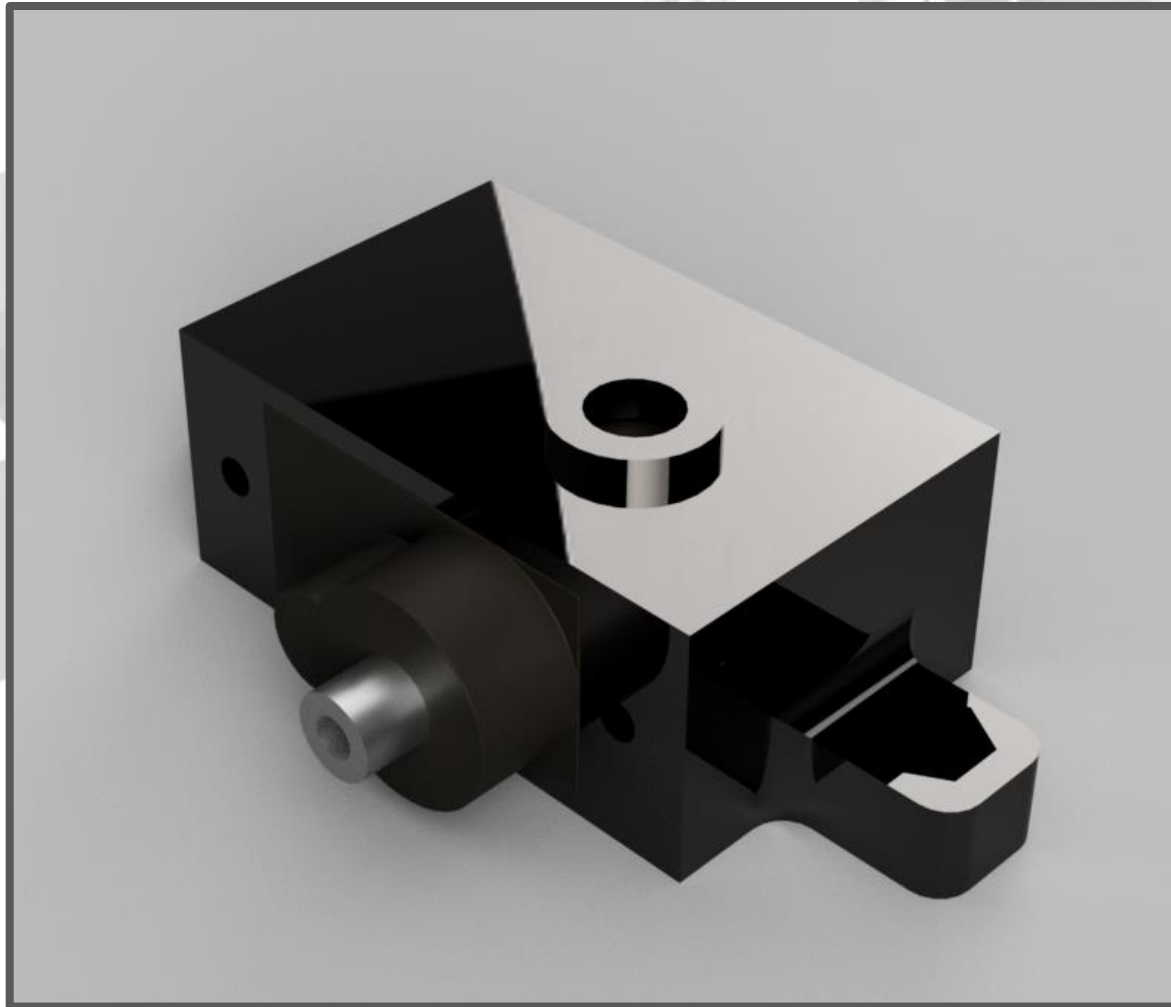


電路



程式

# 馬達座



機構

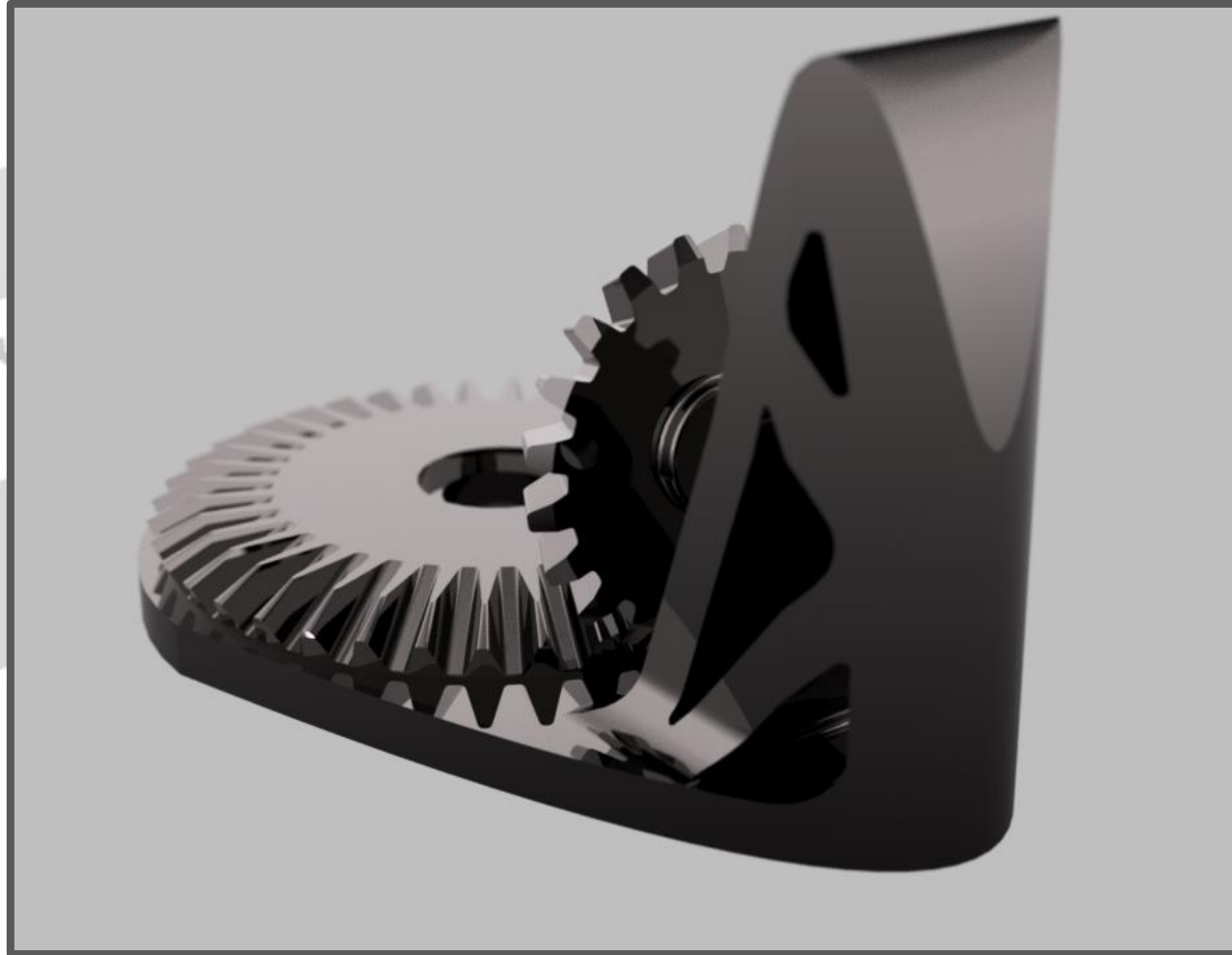


電路



程式

# 傘齒輪設計



機構



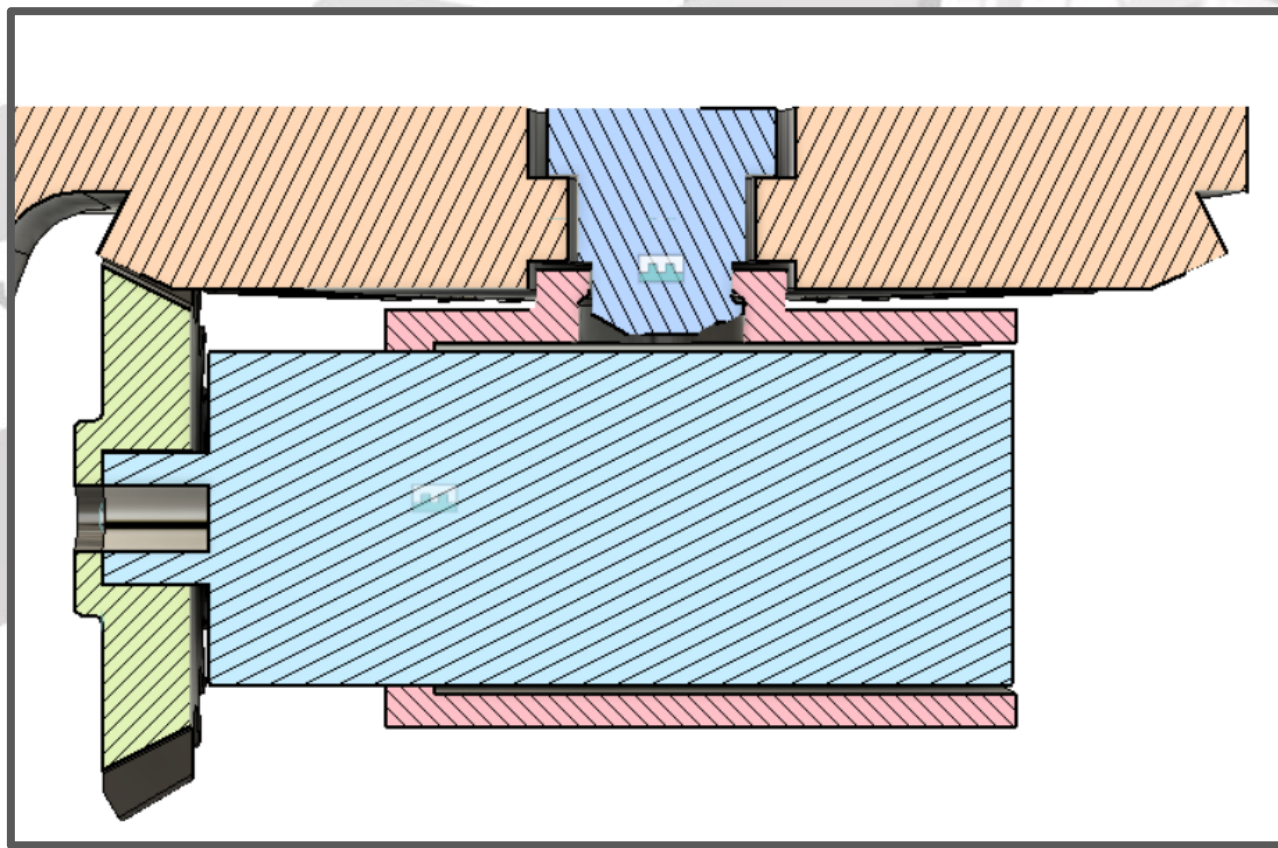
電路



程式



# 馬達齒輪組固定



機構

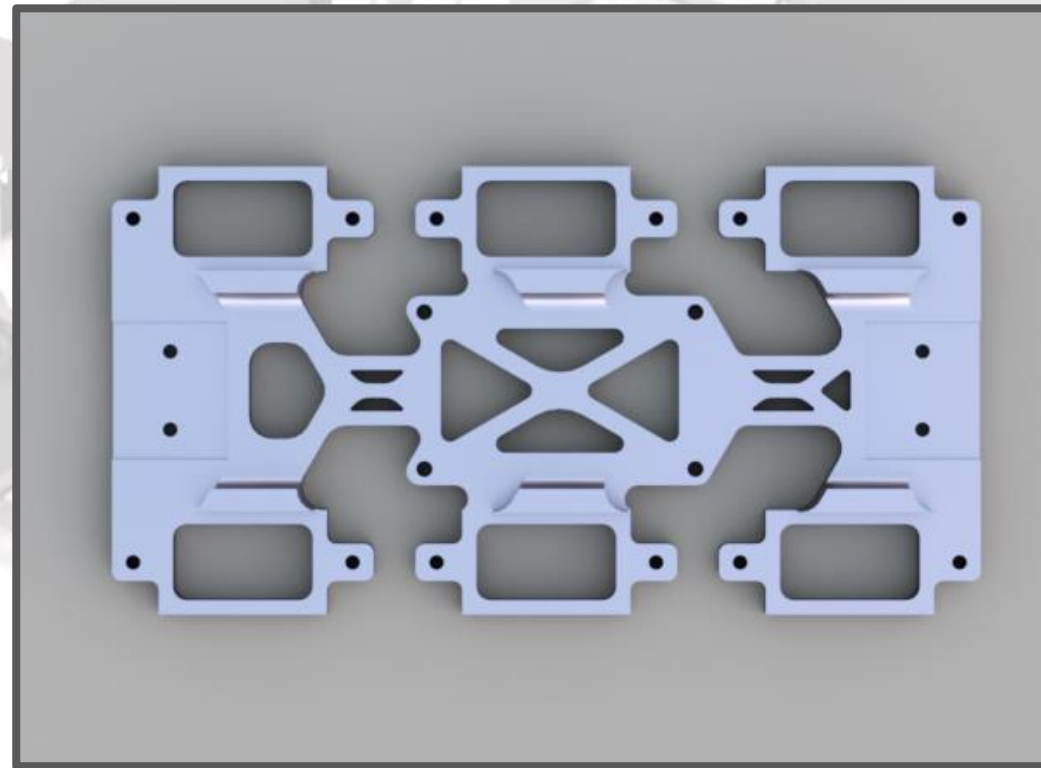
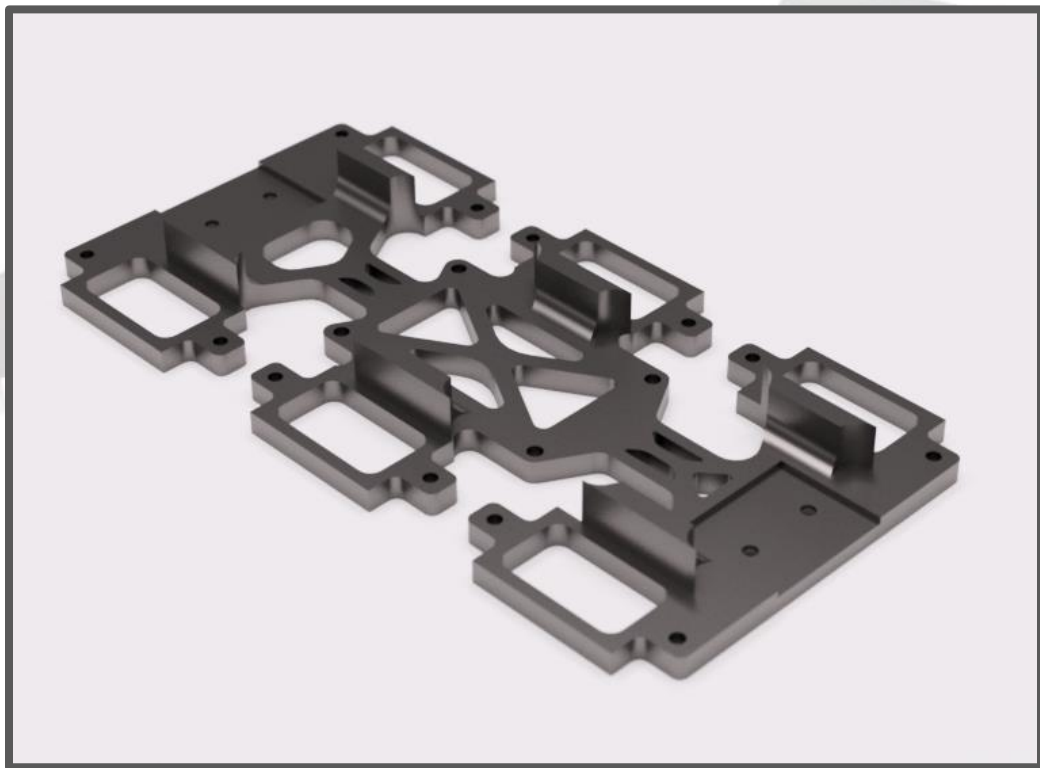


電路



程式

# 組裝平台



機構



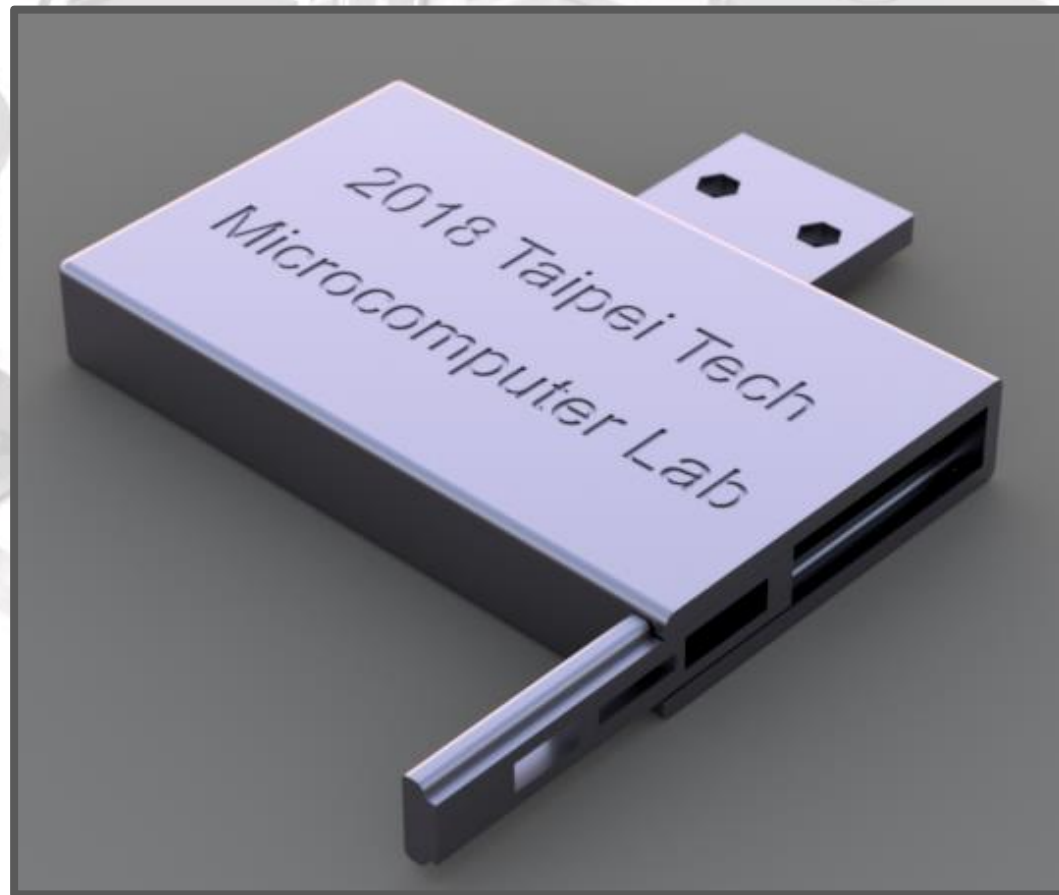
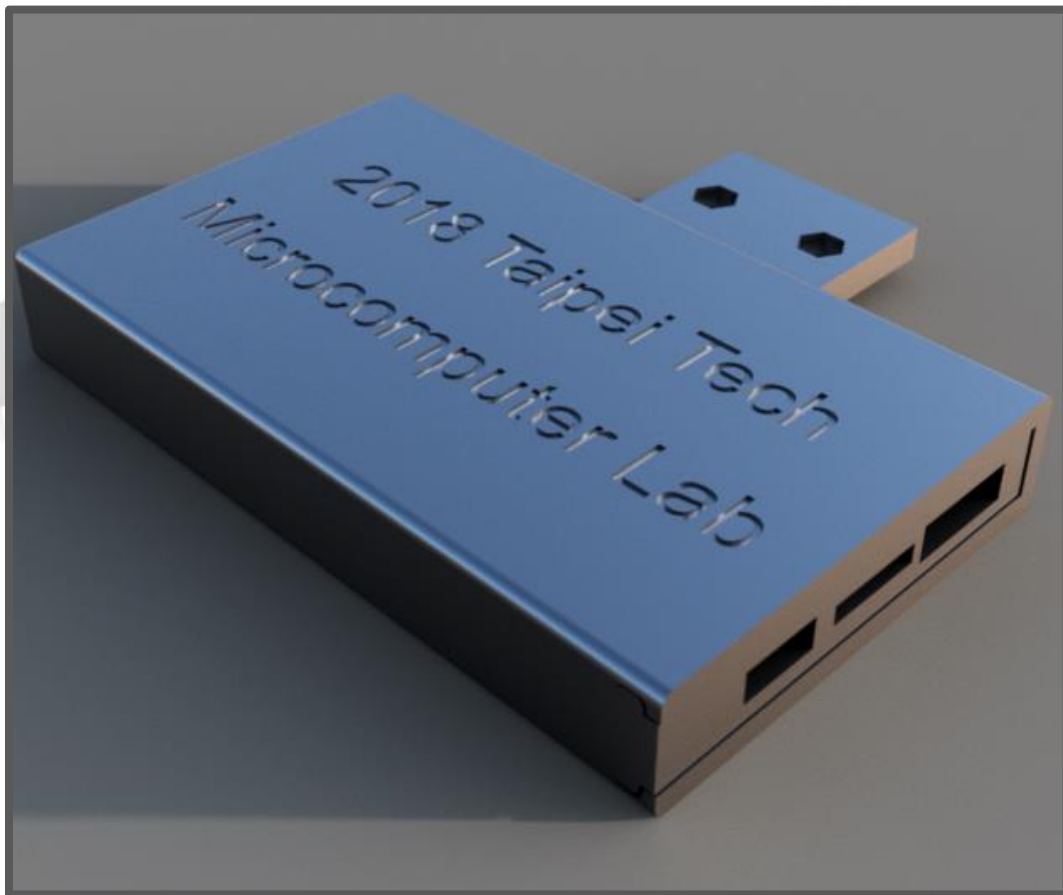
電路



程式



# 電池盒



機構

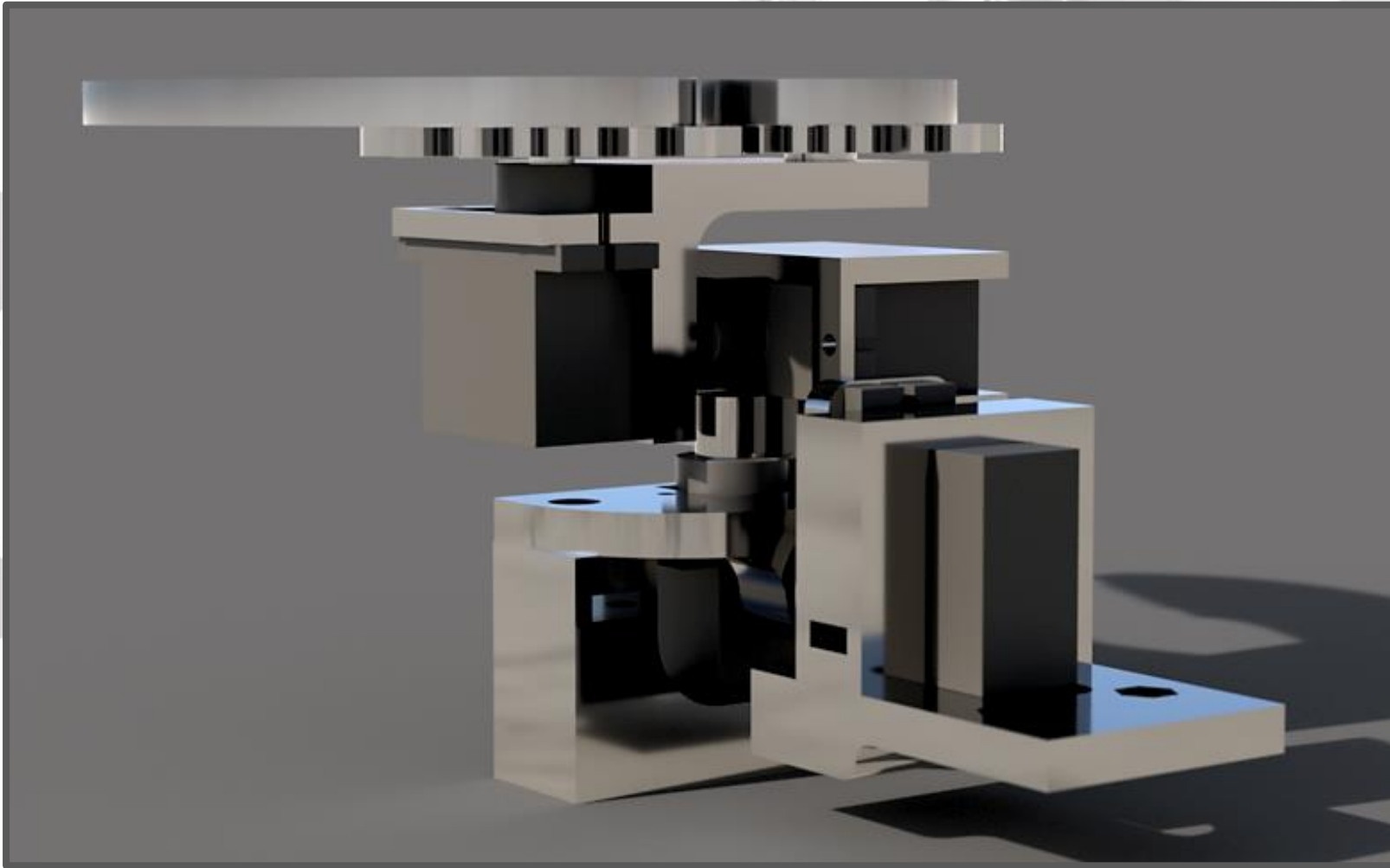


電路



程式

# 三軸平台



機構

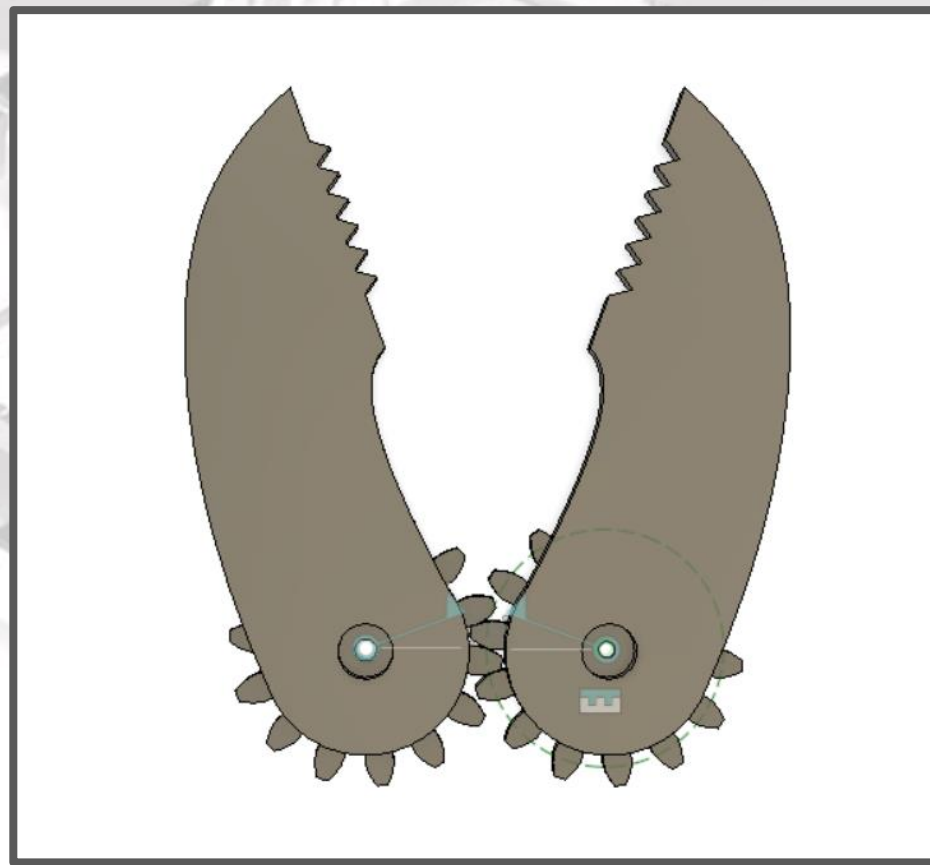
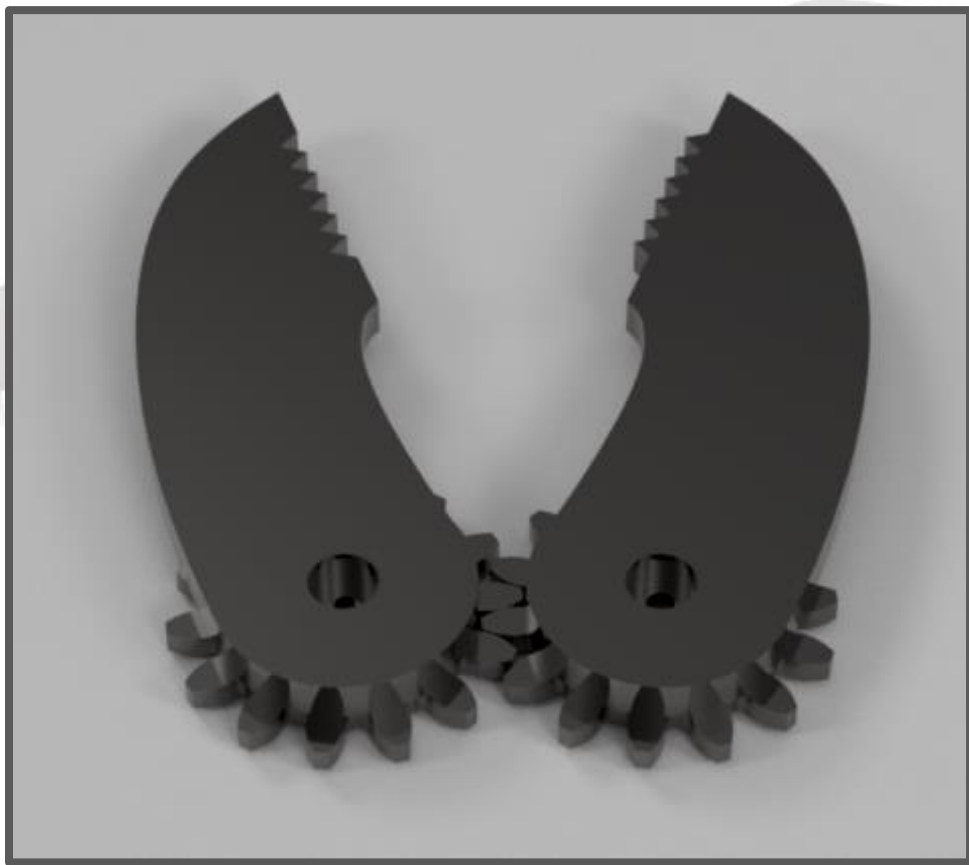


電路



程式

# 夾子



機構

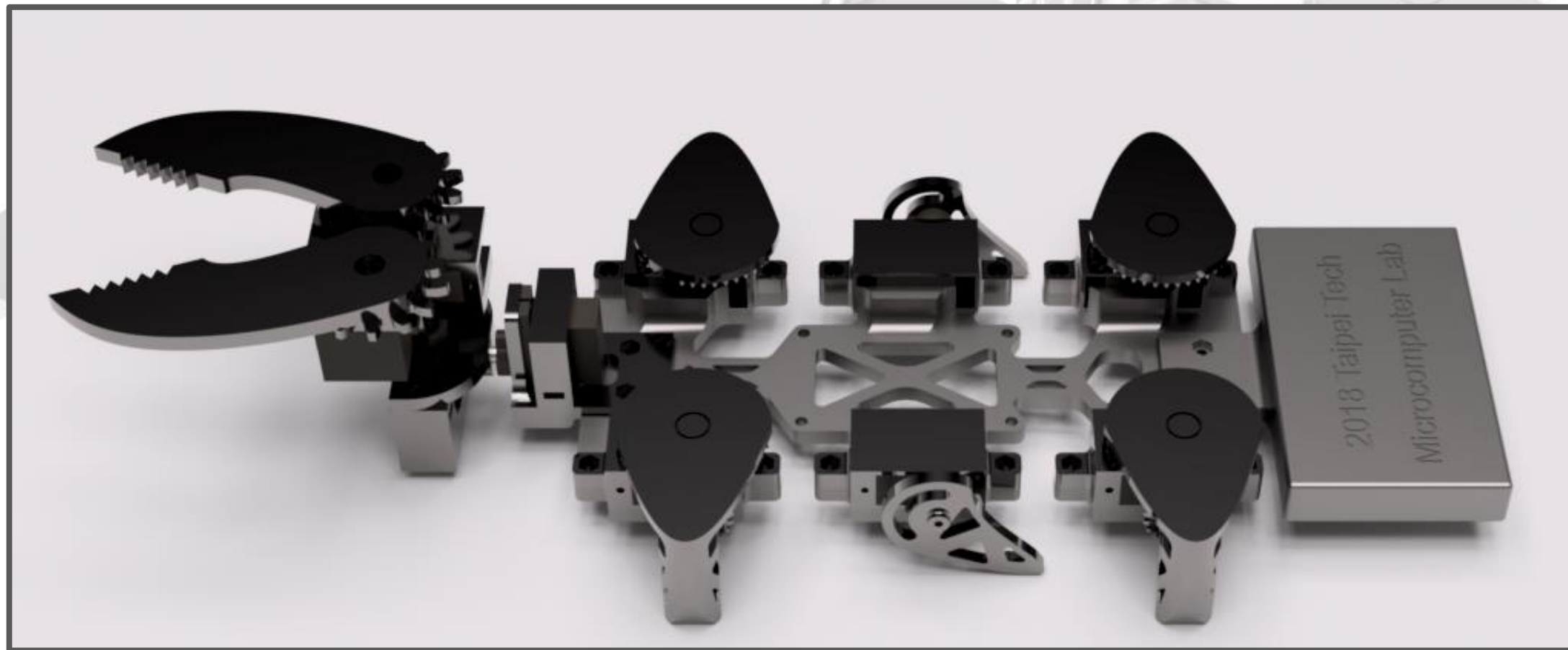


電路



程式

# 整體配置



機構



電路



程式

# 整體配置

夾子  
&  
三軸平台

控制電路  
&  
移動機構

電源

機構



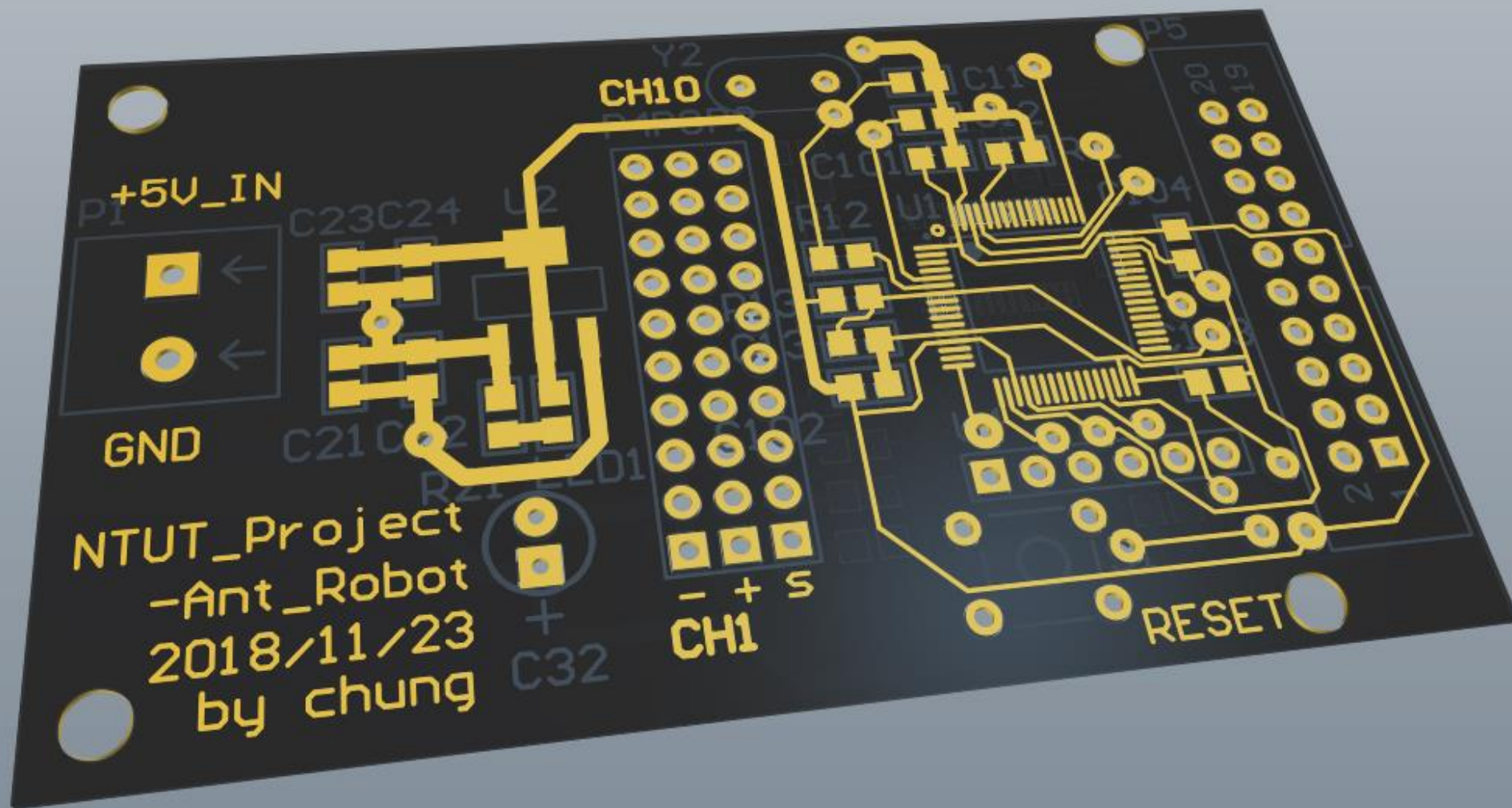
電路



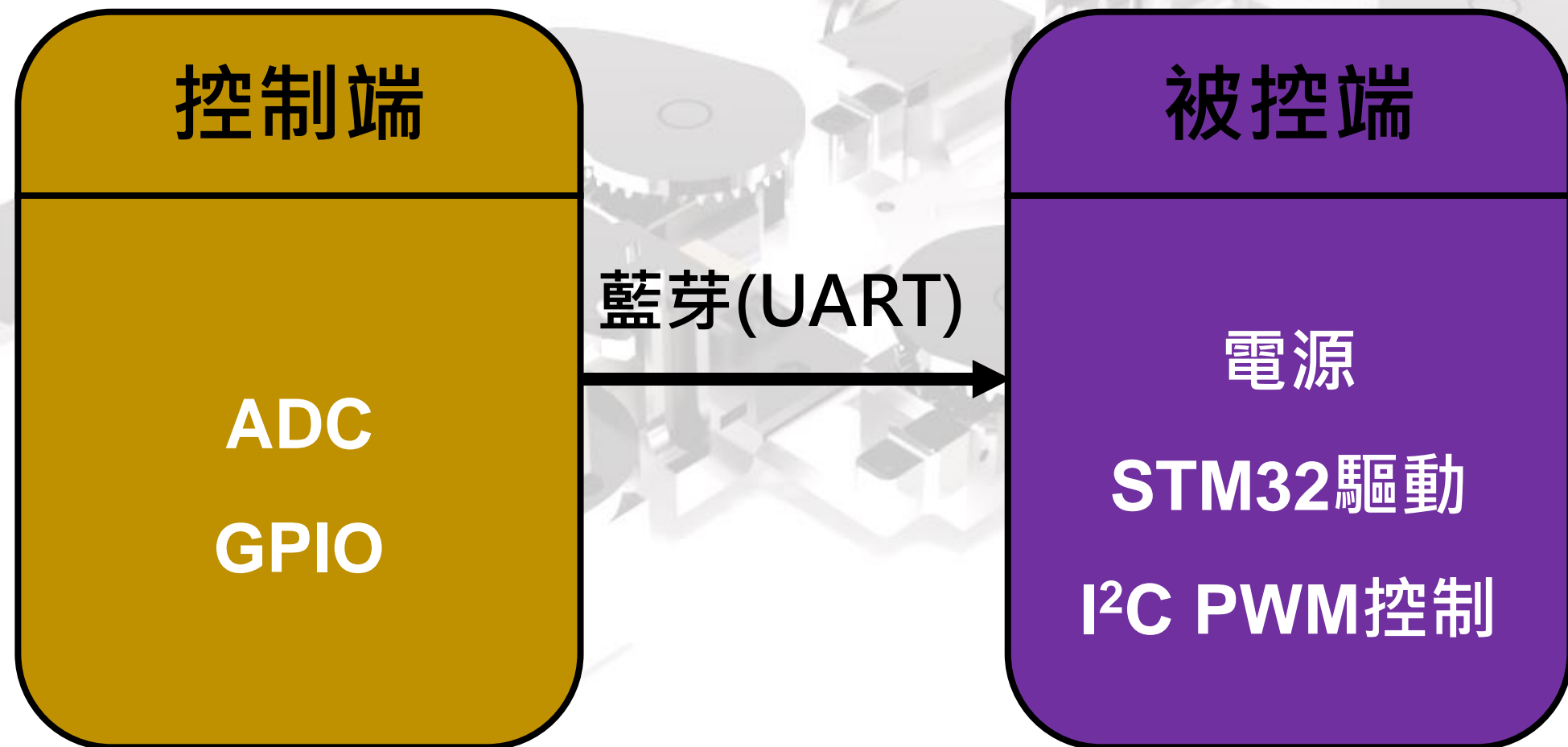
程式



# 電路



# 電路



機構

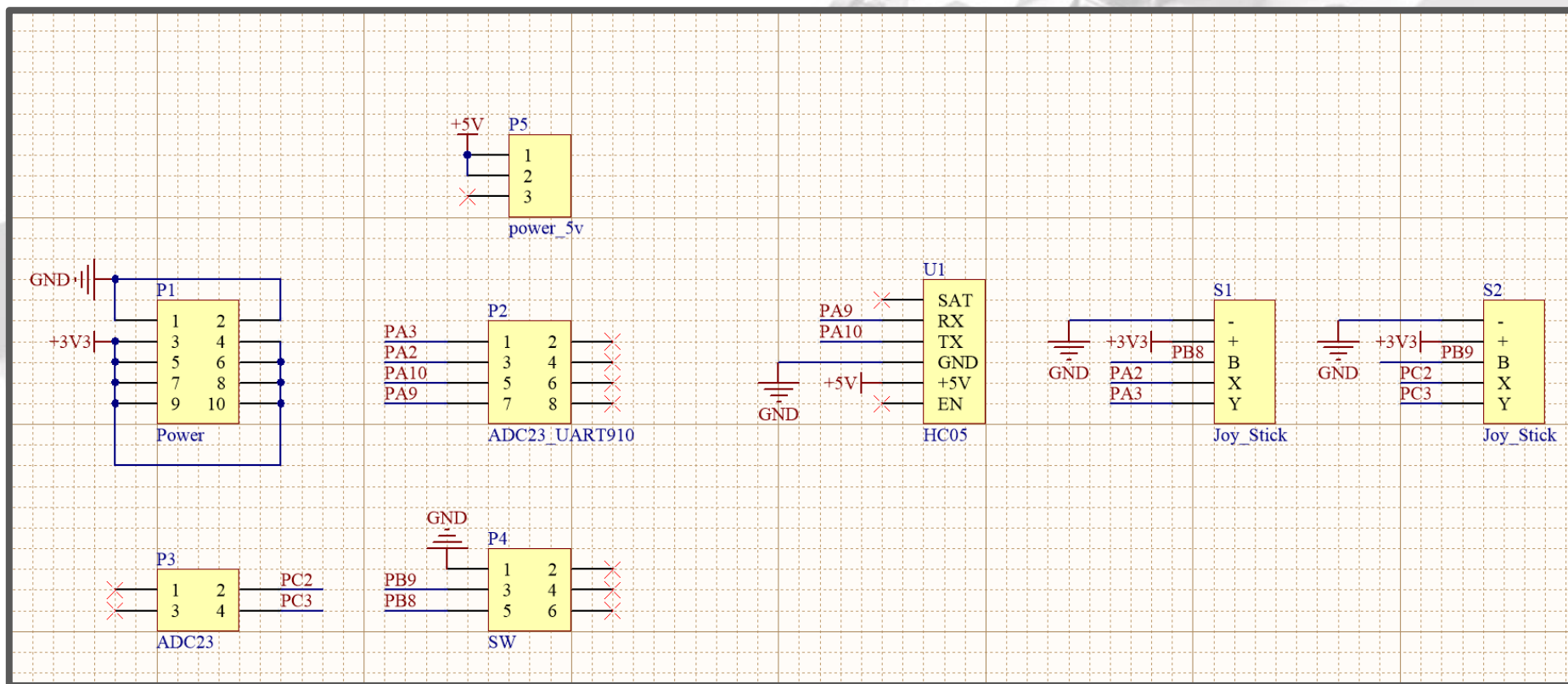


電路



程式

# 控制端電路

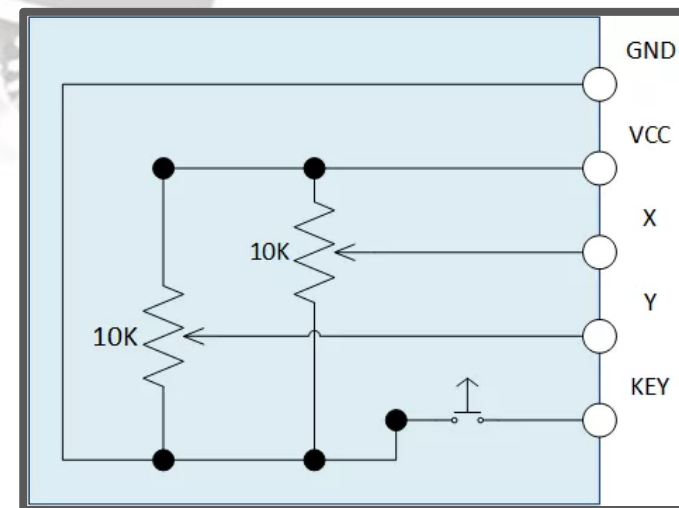
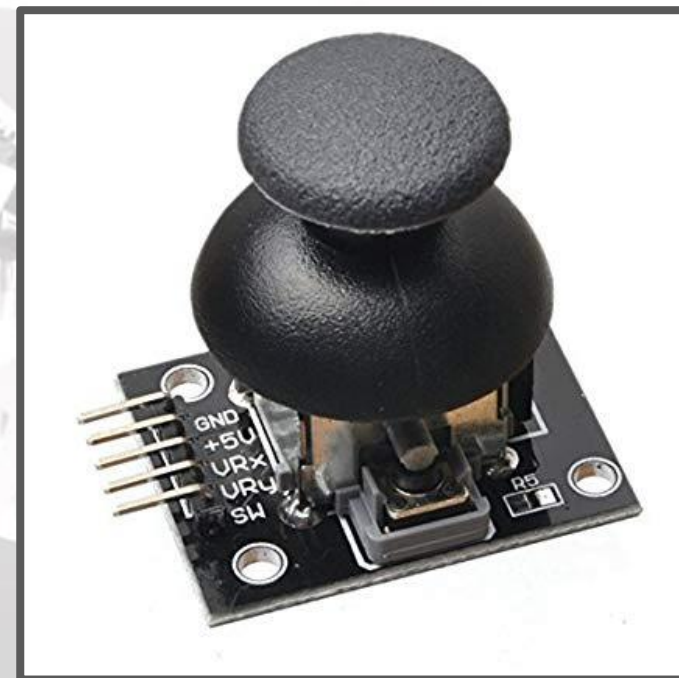
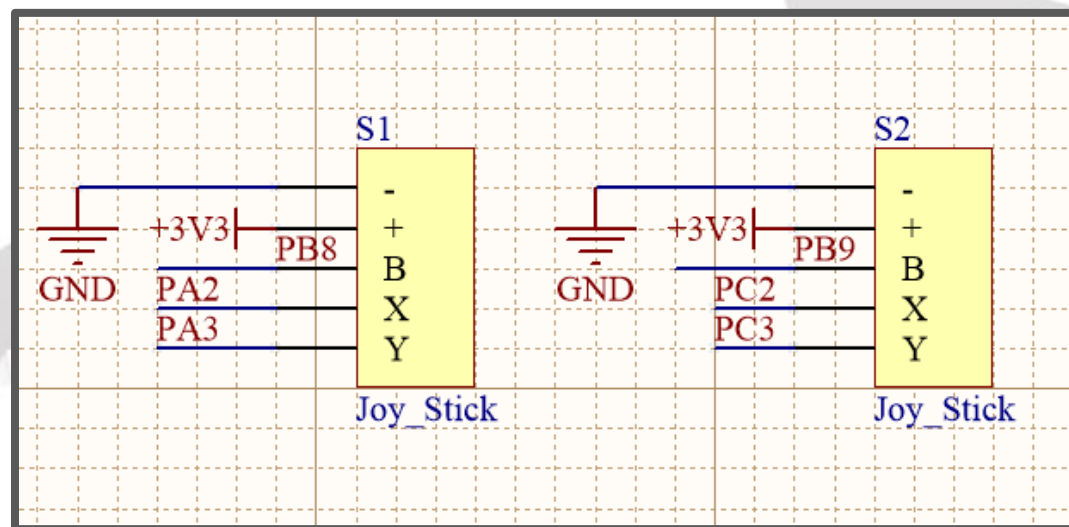


機構

▼  
電路

▼  
程式

# 搖桿

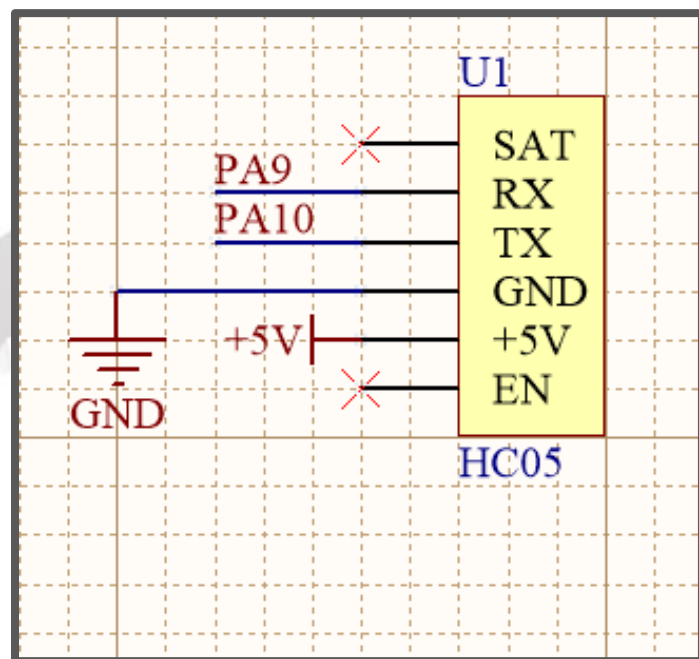


機構

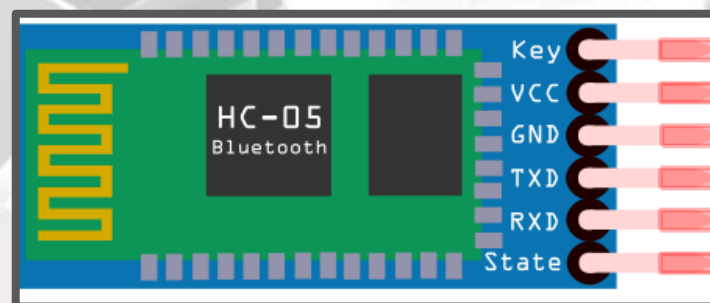
▼  
電路

▼  
程式

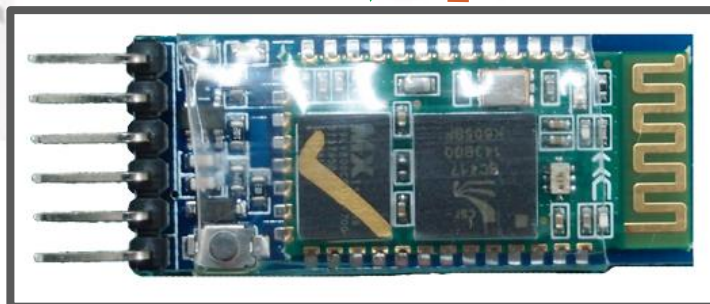
# 藍芽模組(HC-05)



Master

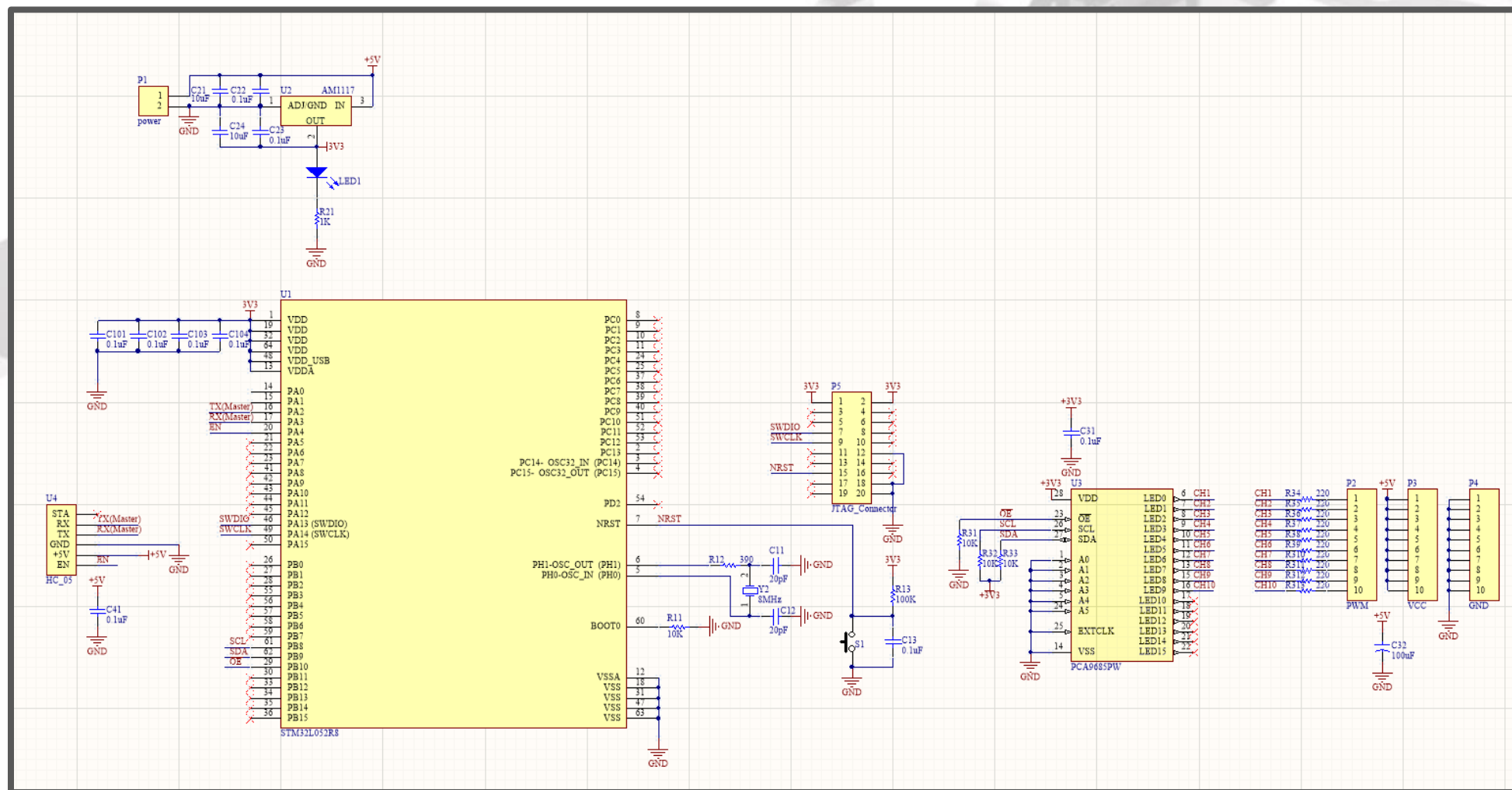


Slave





# 被控端電路



機構

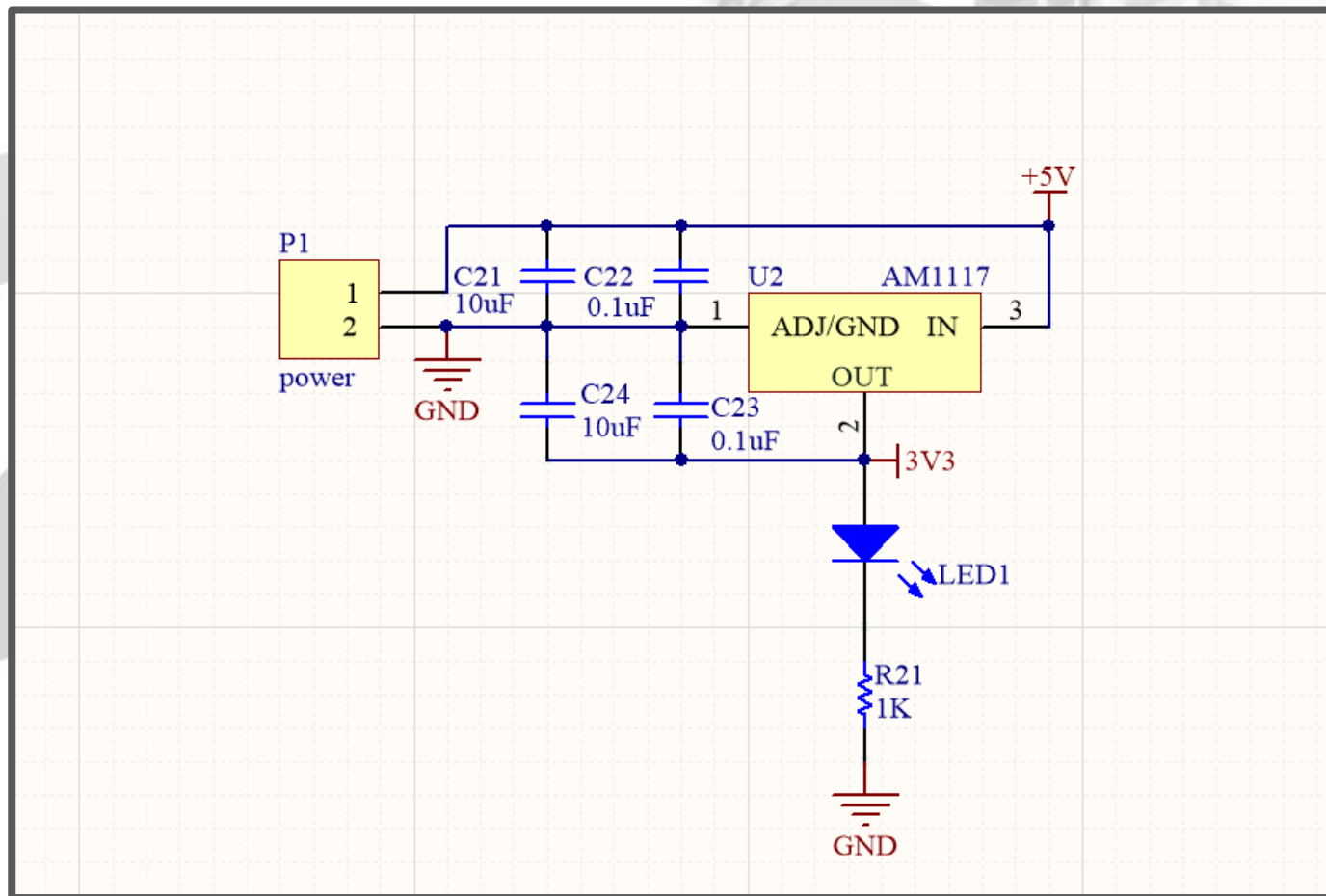


電路



程式

# 電源電路

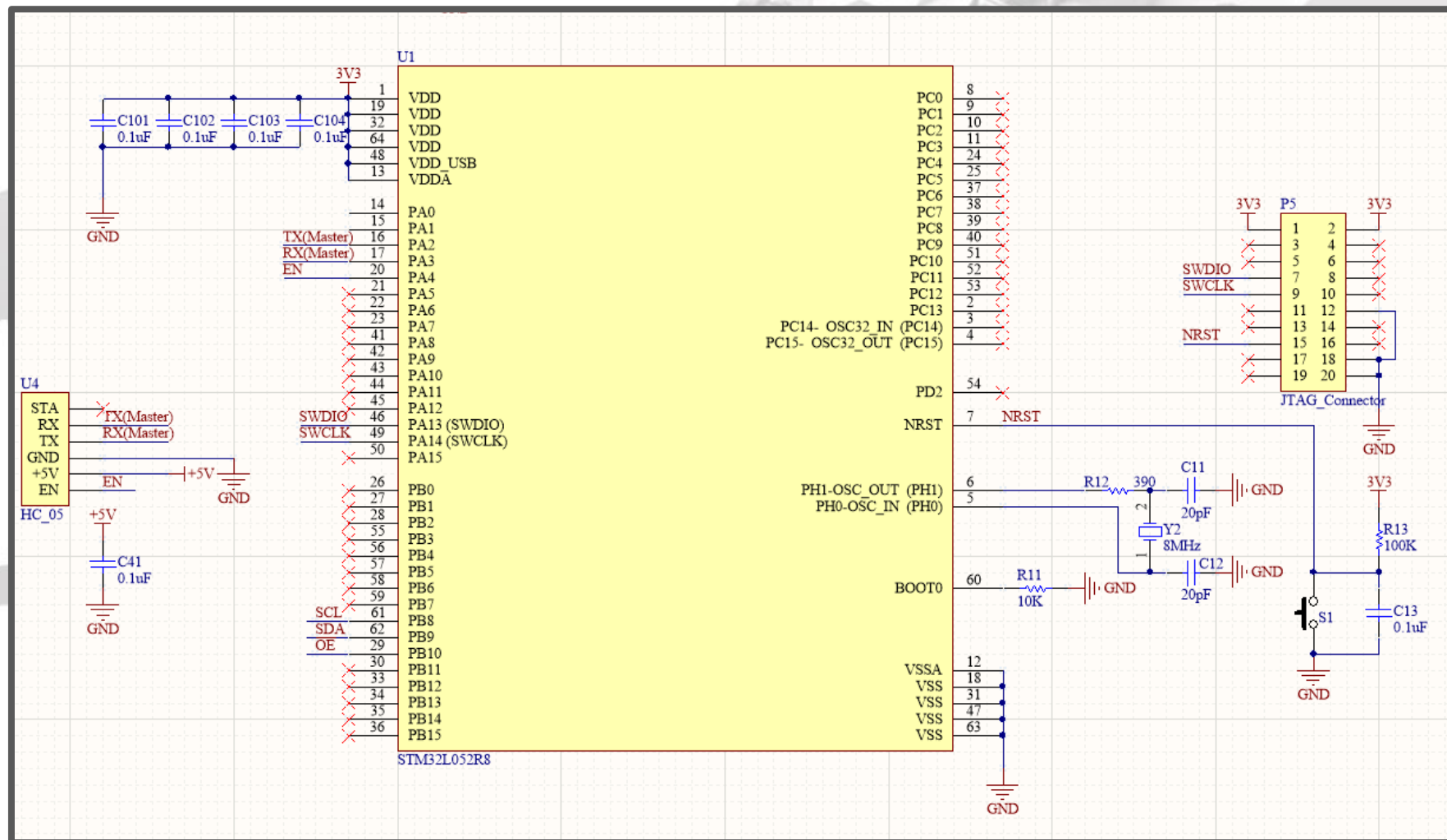


機構

▼  
電路

▼  
程式

# STM32驅動電路

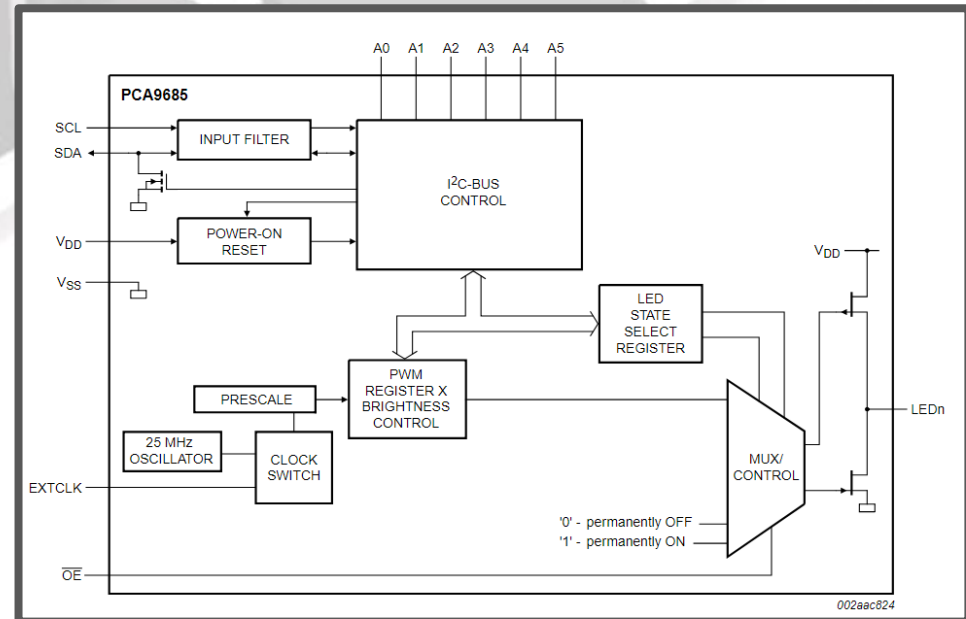
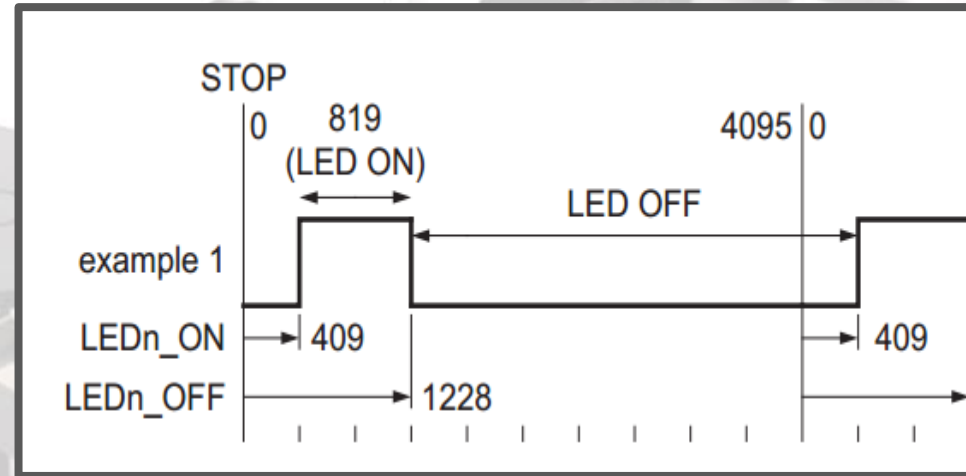
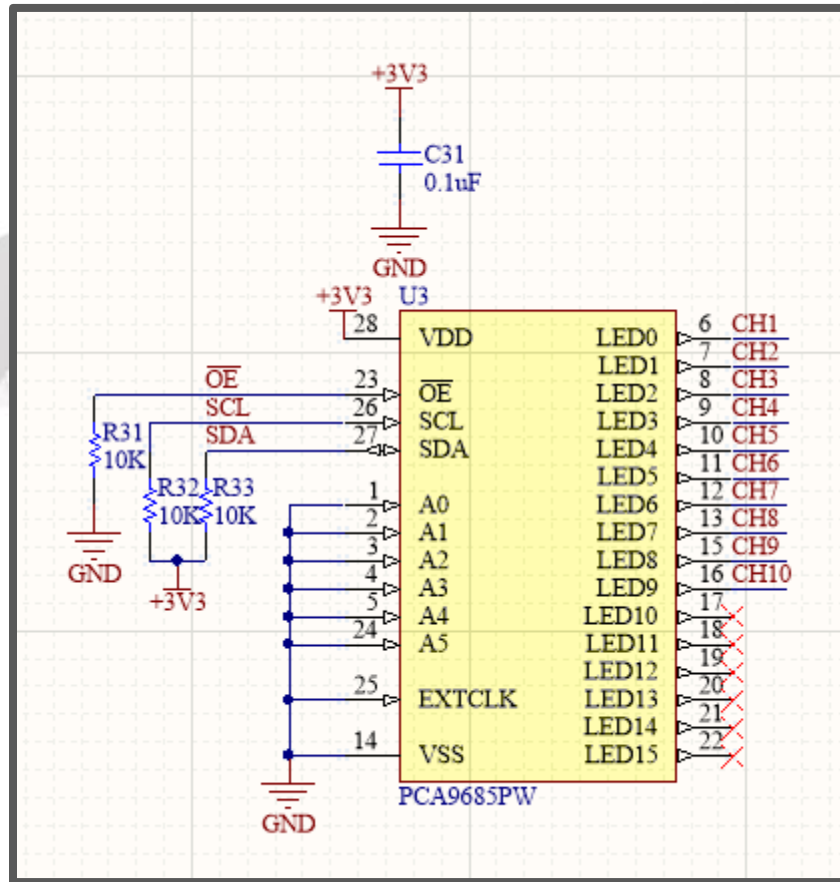


機構

電路

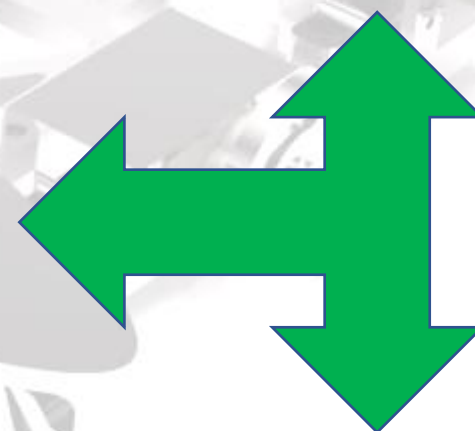
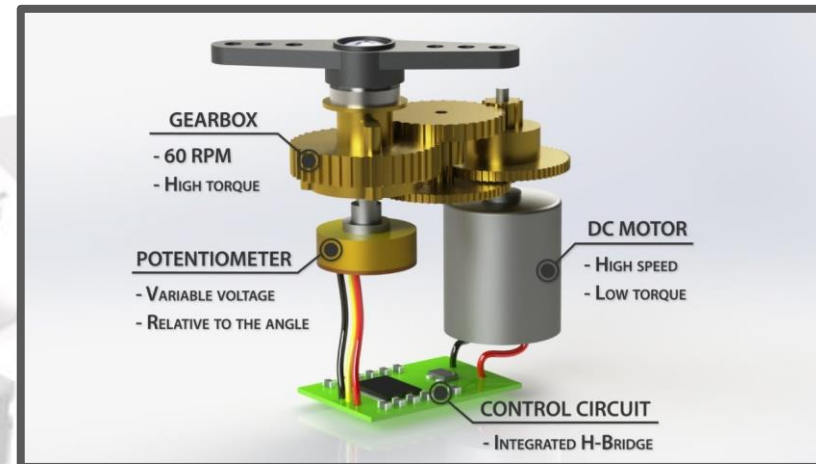
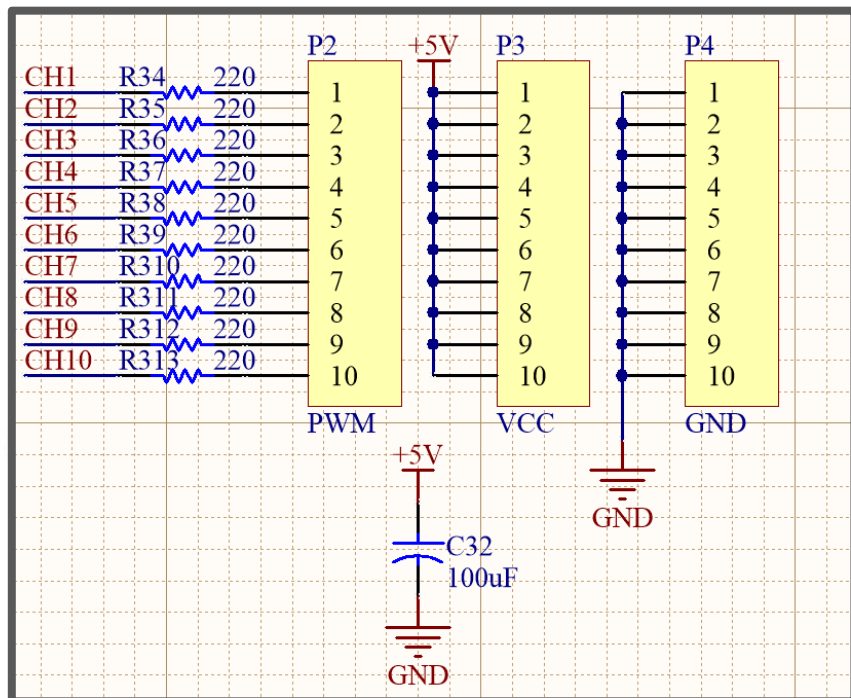
程式

# I2C PWM驅動IC(PC9685)

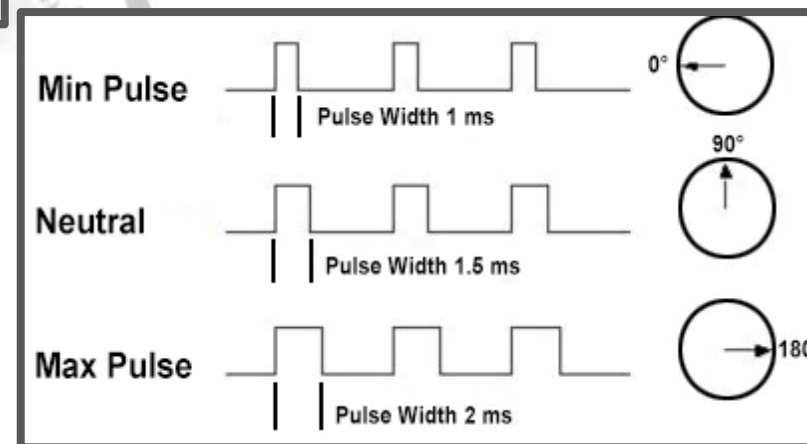


機構  
▼  
電路  
▼  
程式

# 伺服馬達

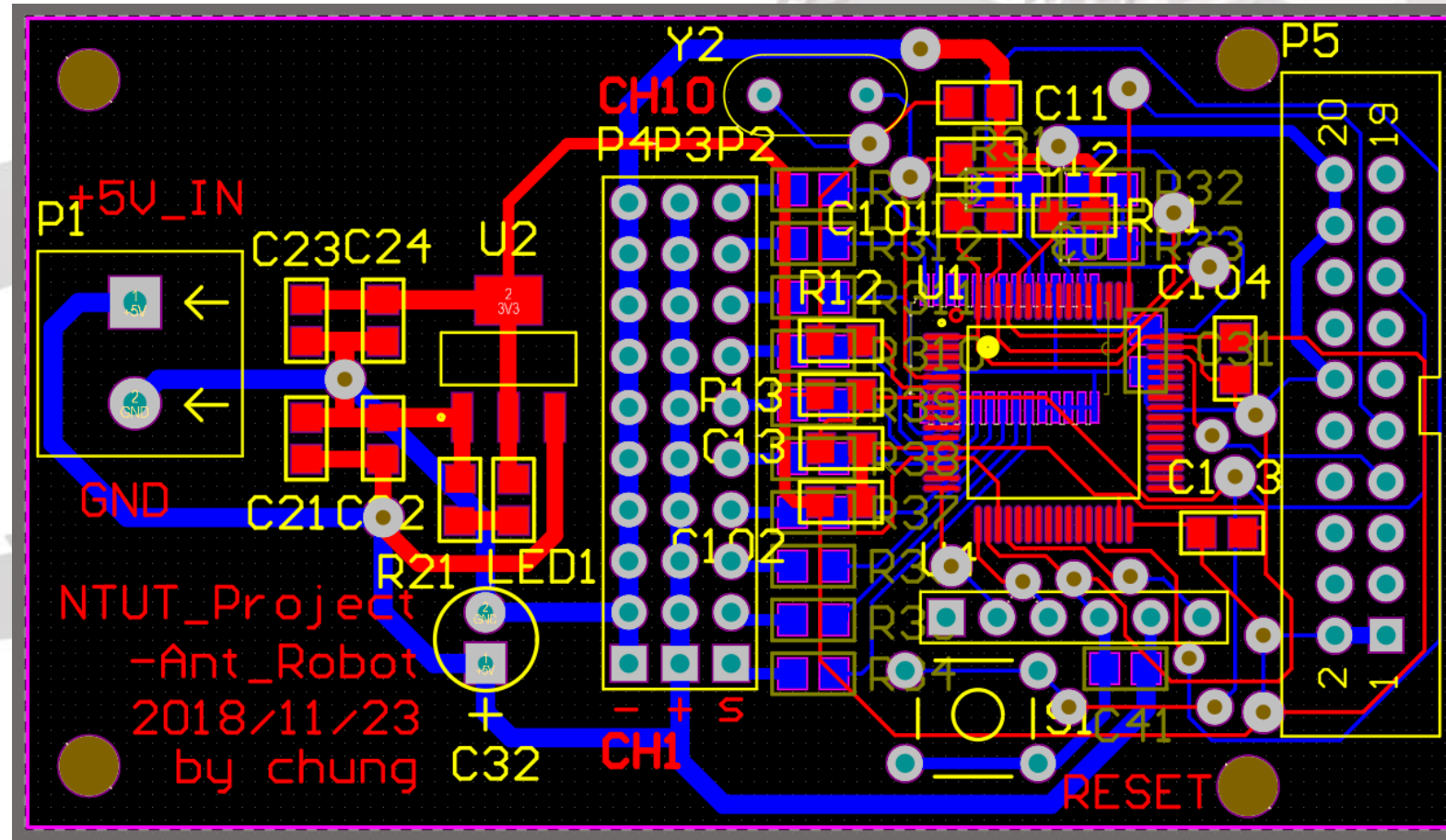


機構  
▼  
電路  
▼  
程式





# PCB Layout



機構

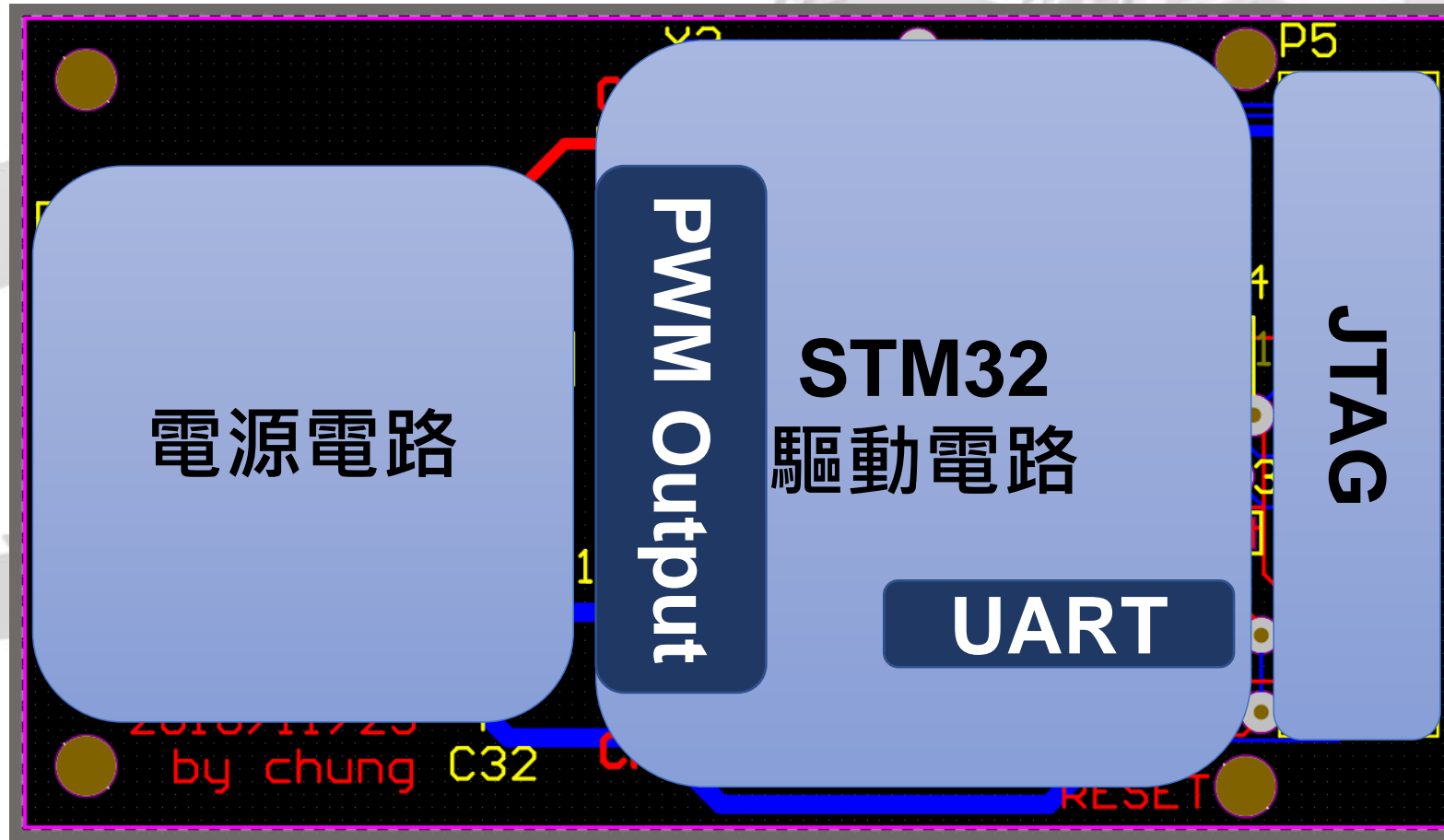


電路



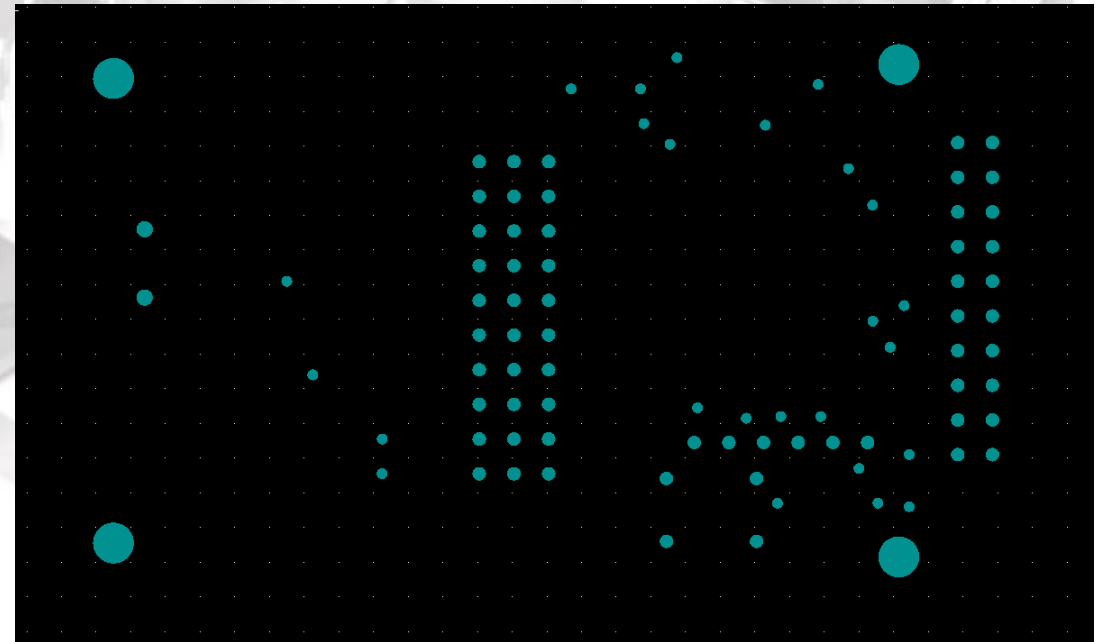
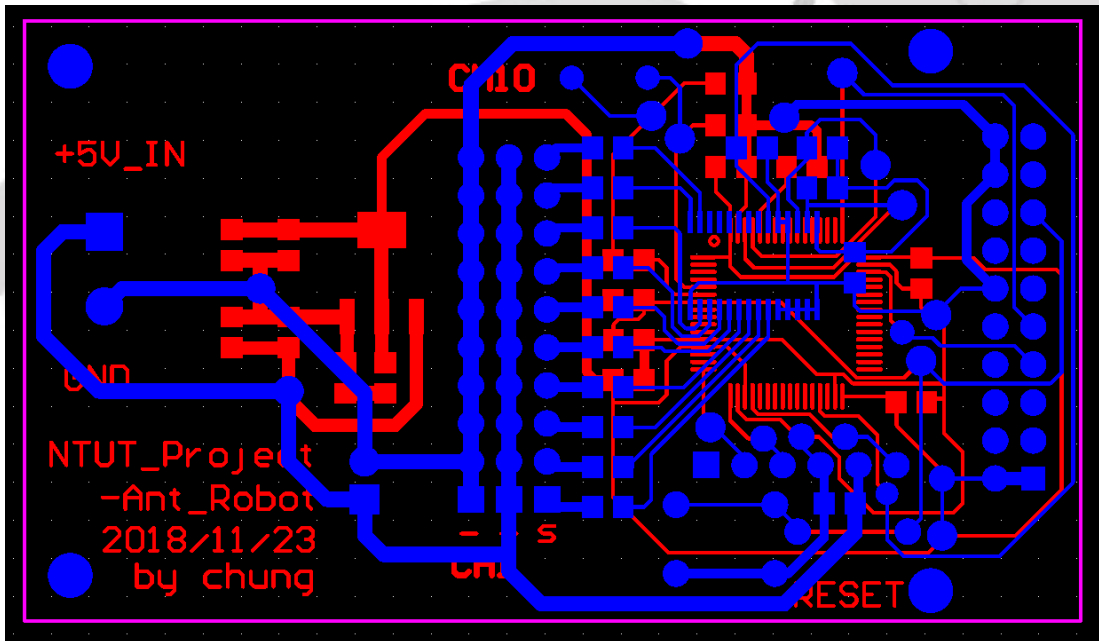
程式

# PCB Layout



機構  
▼  
電路  
▼  
程式

# Gerber file & NC drill



機構

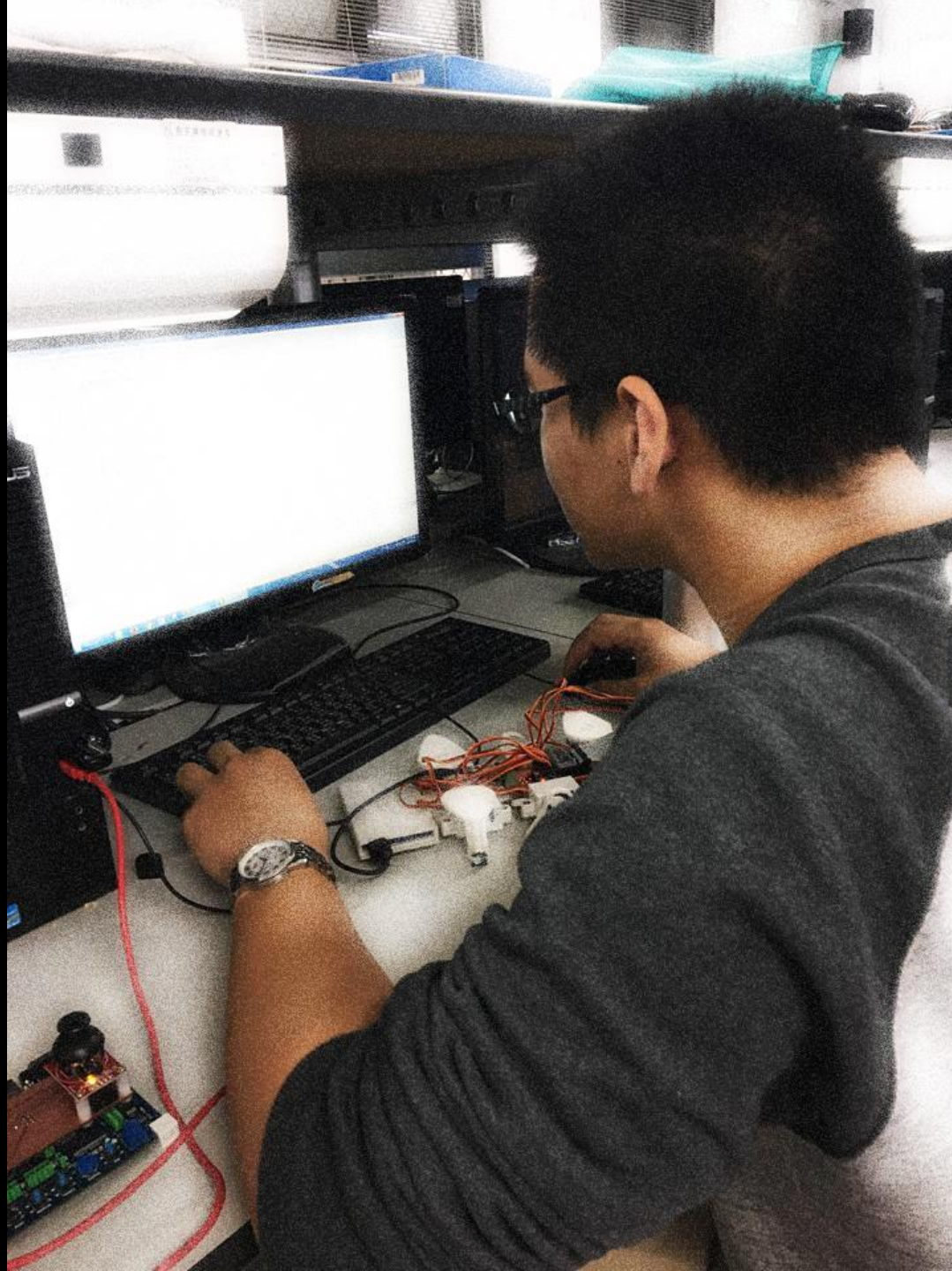


電路



程式

# 程式



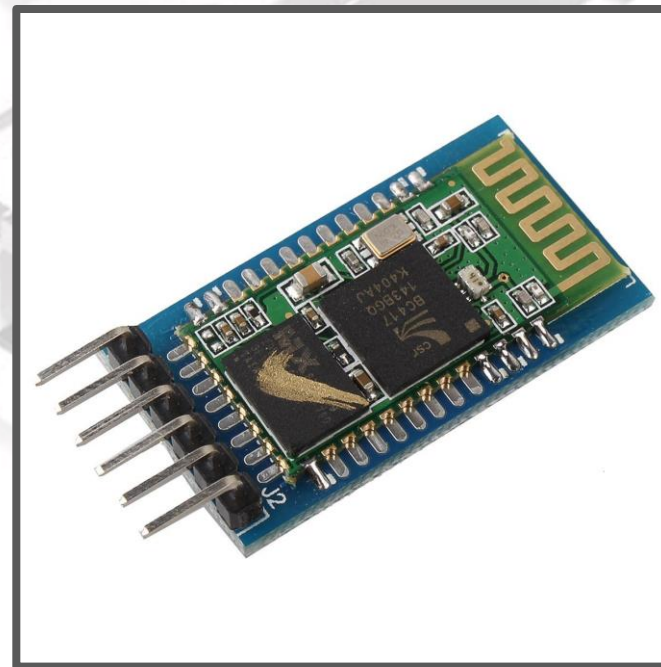


# 控制端

ADC DMA



UART



機構



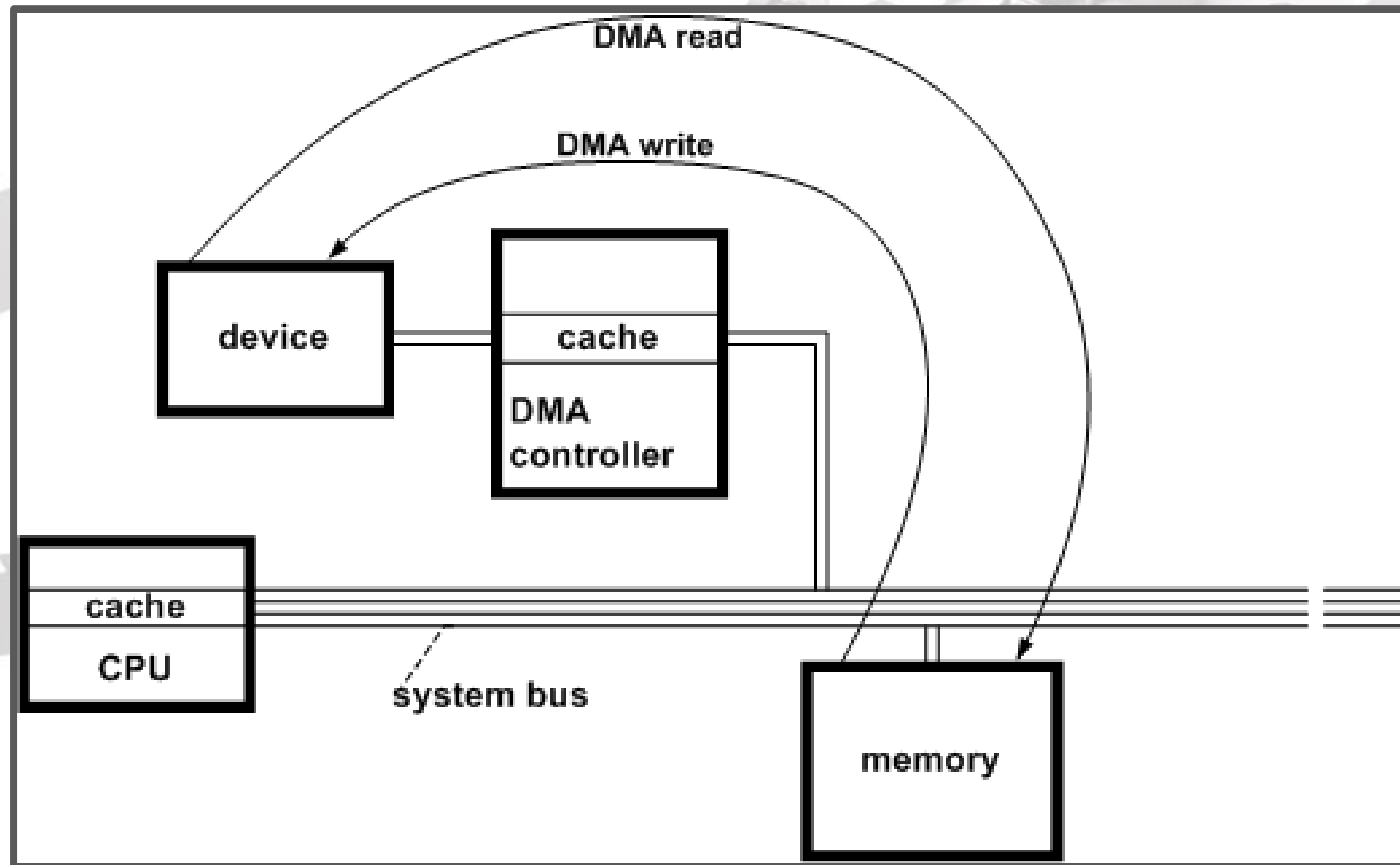
電路



程式



# DMA



機構  
▼  
電路  
▼  
程式

# ADC DMA

```
HAL_ADC_Start_DMA(&hadc1,adc_temp,5);
```

```
uint32_t X1_Value=2048,Y1_Value=2048,X2_Value=0,Y2_Value=0,VR_Value=0;
```

```
void HAL_ADC_ConvCpltCallback(ADC_HandleTypeDef* hadc) {  
    X1_Value = adc_temp[0];  
    Y1_Value = adc_temp[1];  
    X2_Value = adc_temp[2];  
    Y2_Value = adc_temp[3];  
    VR_Value = adc_temp[4];  
}
```

機構



電路



程式

# UART Package

0	1	2	3
Command	Speed	Angle1	Angle2
4	5	6	7
Angle3	SW_Clip	End	NC

機構



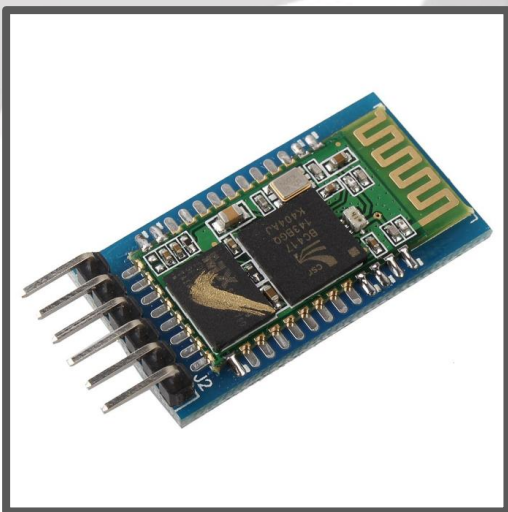
電路



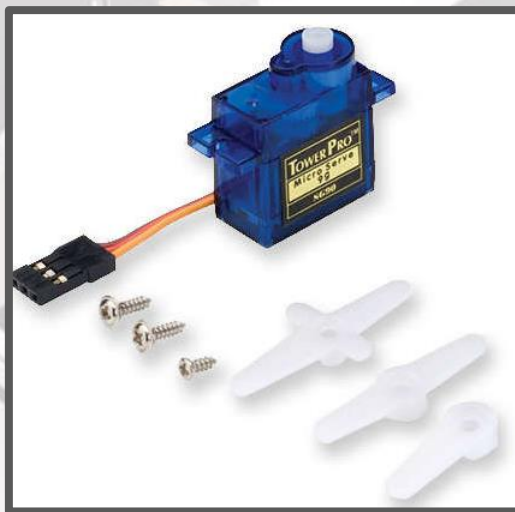
程式

# 被控端

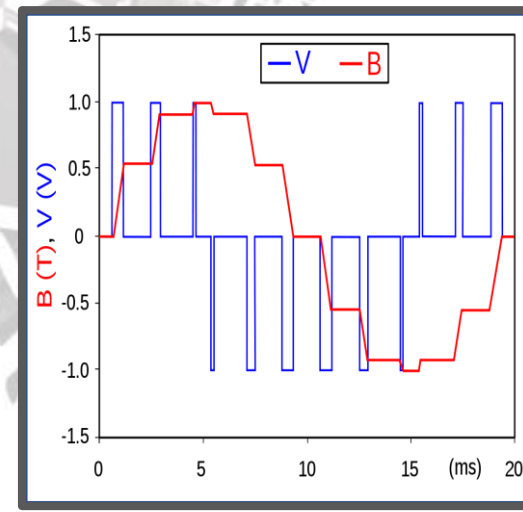
UART



setAngle



setPWM



機構



電路



程式

# Main loop

```
while (1) {  
    if(command=='F') { Move_Forword(); }  
    else if(command=='R') { Turn_Right(); }  
    else if(command=='L') { Turn_Left(); }  
    else if(command=='S') { Reset_Position(); }  
}
```

機構



電路



程式

# Decoding UART Package

```
void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart) {  
    if(huart->Instance==USART2) {  
        if(str1[6]=='E') {  
            command = str1[0];    speed = str1[1];  
            angle1 = (int)str1[2]; angle2 = (int)str1[3];  
            angle3 = (int)str1[4]; clip = str1[5];  
        }  
        HAL_UART_Receive_IT(&huart2, str1, 7);  
        //Enable rx interrupt every time  
    }  
}
```

機構



電路



程式



# setAngle

```
void setAngle(uint8_t num, uint8_t angle){  
    int pulse_wide, analog_value;  
    pulse_wide = map(angle, 0, 180, MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);  
    analog_value = (int)((float)(pulse_wide) / 1000000 * 50 * 4096);  
    setPWM(num, 0, analog_value);  
}
```

機構

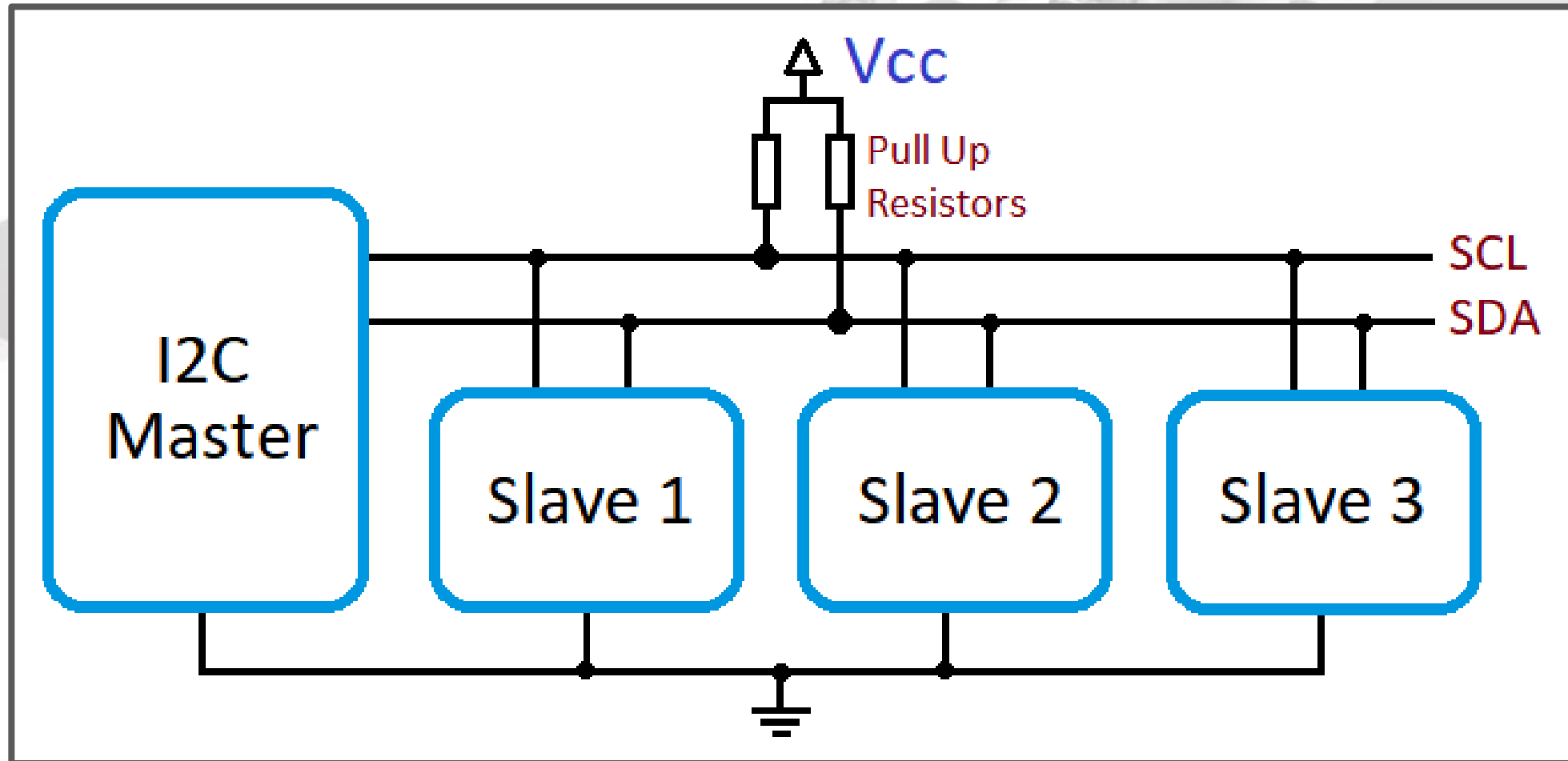


電路



程式

# setPWM - I<sup>2</sup>C



# setPWM

```
void setPWM(uint8_t num, uint16_t on, uint16_t off) {  
    uint8_t data_buf[] = {on, (on >> 8), off, (off >> 8)};  
    I2C_WRITE_buf(LED0_ON_L + 4 * (num), data_buf, 4);  
}
```

Slave address		Control register	Data for register
1	A5:A0	D7:D0	D7:D0

機構



電路

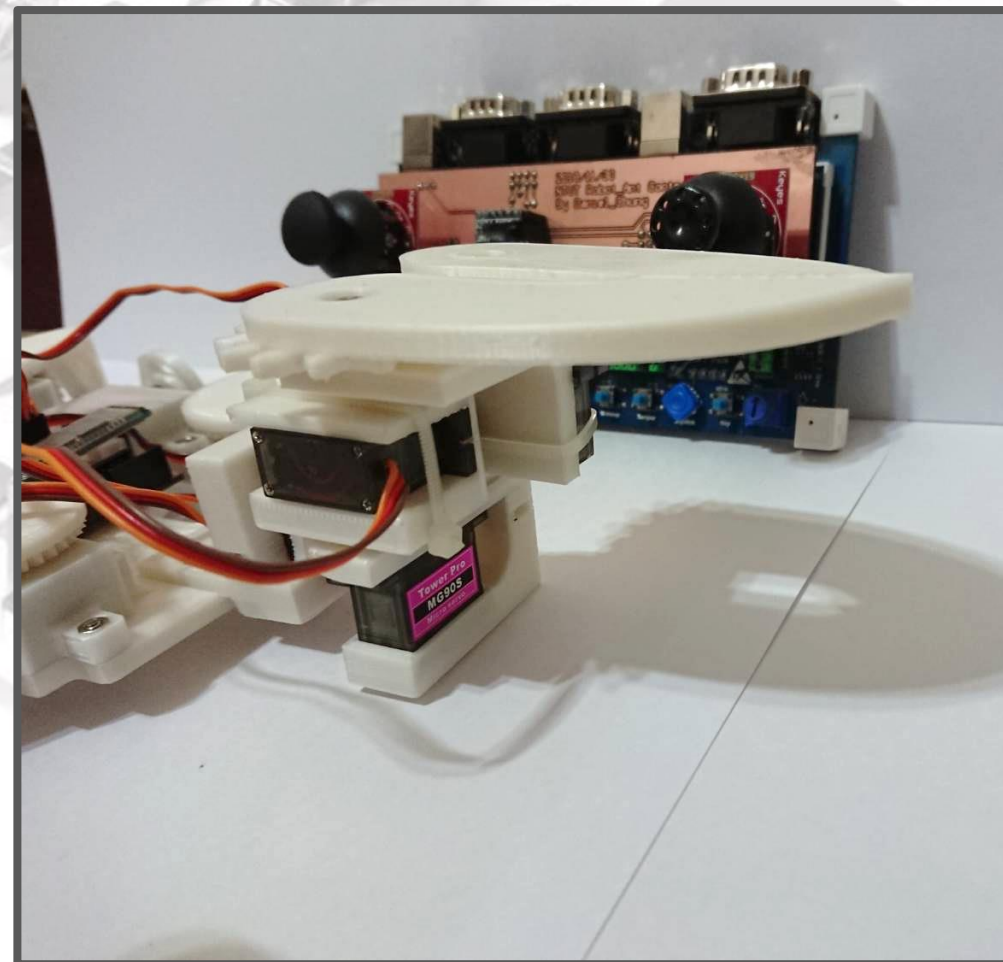
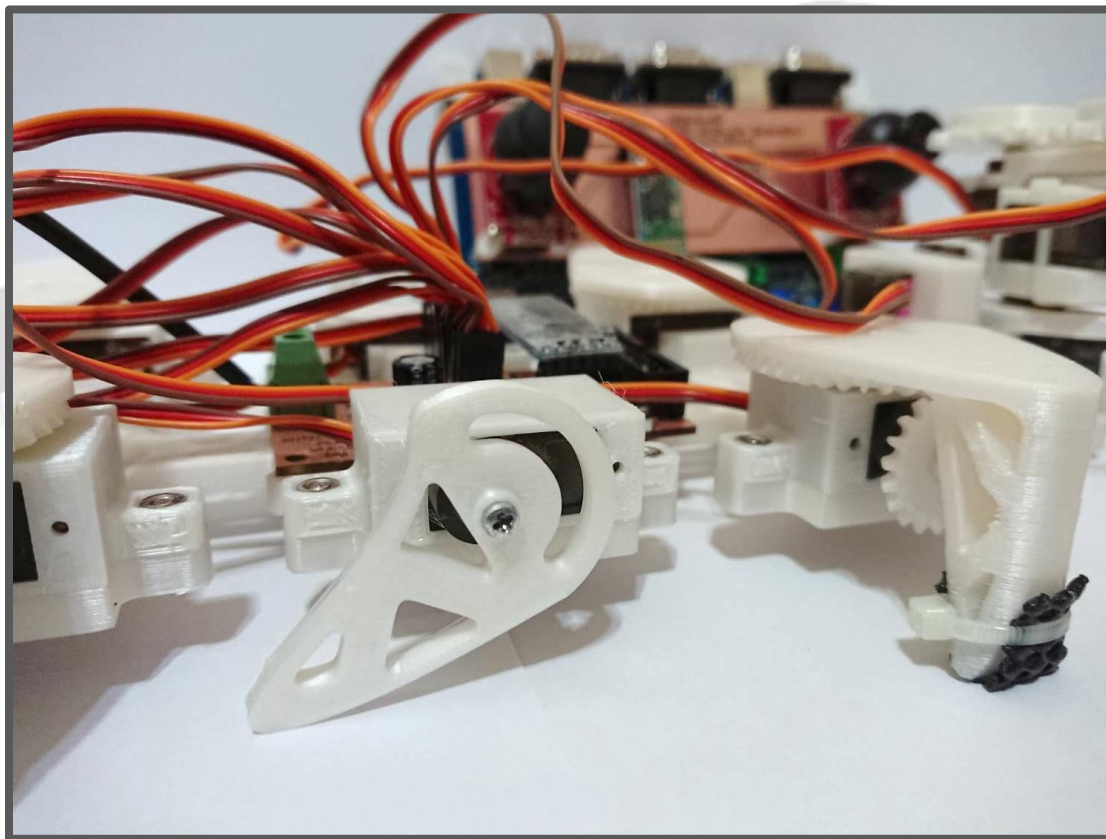


程式

# 成品照1



# 成品照2





# 成品照3

