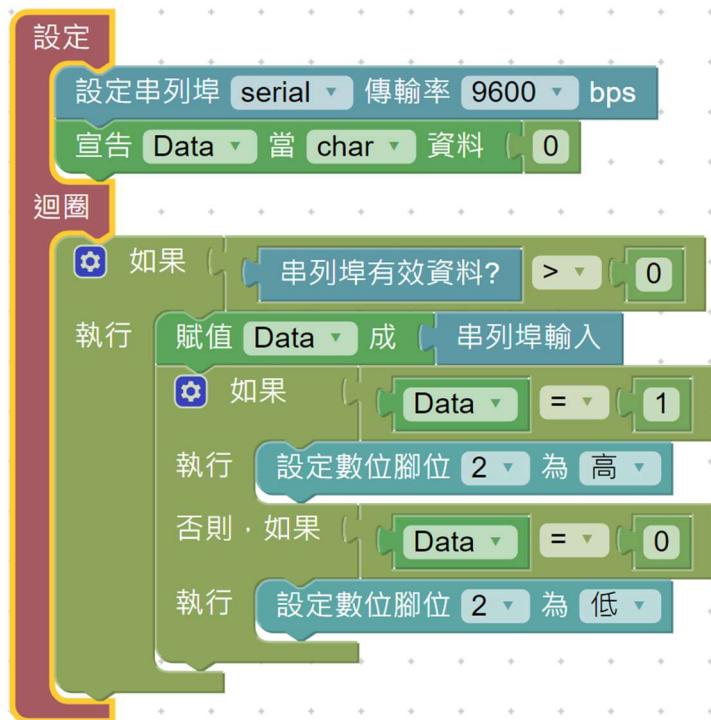
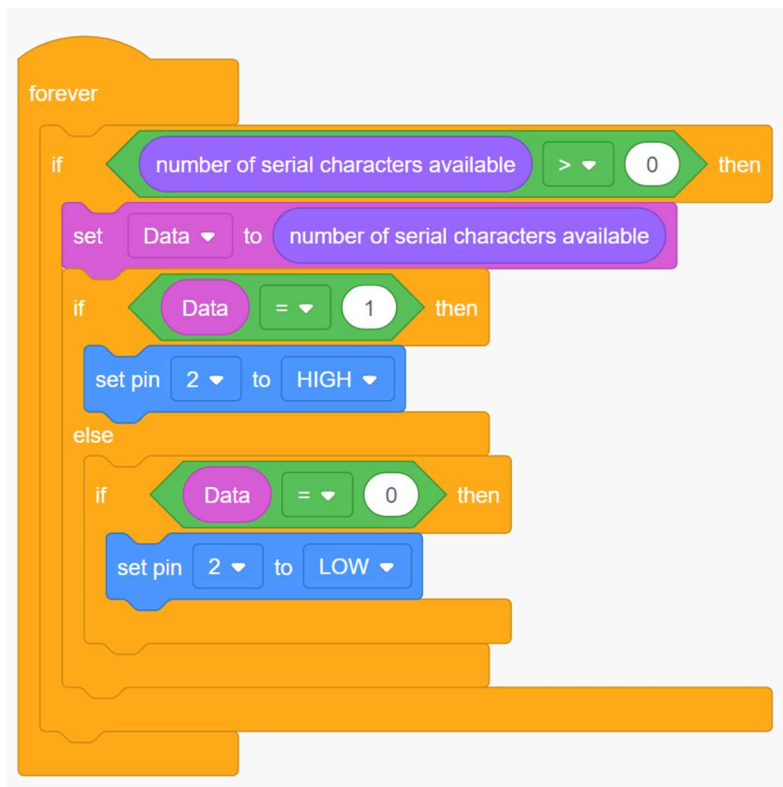




```

char Data=0;
void setup()
{
  Serial.begin(9600);
  pinMode(2, OUTPUT);
}

void loop()
{
  if (Serial.available() > 0) {
    Data = Serial.read();
    if (Data == 1) {
      digitalWrite(2,HIGH);
    } else if (Data == 0) {
      digitalWrite(2,LOW);
    }
  }
}
  
```



```
int Data = 0;
```

```
void setup()
```

```
{
```

```
  Serial.begin(9600);
```

```
  pinMode(2, OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
  if (Serial.available() > 0) {
```

```
    Data = Serial.available();
```

```
    if (Data == 1) {
```

```
      digitalWrite(2, HIGH);
```

```
    } else {
```

```
      if (Data == 0) {
```

```
        digitalWrite(2, LOW);
```

```
      }
```

```
    }
```

```
}
```

```
  delay(10); // Delay a little bit to improve simulation performance
```

```
}
```

設定

設定串列埠 serial 傳輸率 9600 bps

宣告 Data 當 char 資料 0

伺服馬達 腳位# 6
角度(0~180) 0
延遲 0

迴圈

如果 串列埠有效資料? > 0

執行 賦值 Data 成 串列埠輸入

如果 Data = 1

執行 設定數位腳位 2 為 高

伺服馬達 腳位# 6
角度(0~180) 180
延遲 0

否則, 如果 Data = 0

執行 設定數位腳位 2 為 低

伺服馬達 腳位# 6
角度(0~180) 0
延遲 0

否則, 如果 Data = 2

執行 360度伺服馬達 腳位# 9

旋轉方向 順時針
速度 快速

否則, 如果 Data = 3

執行 360度伺服馬達 腳位# 9 停止轉動

否則, 如果 Data = 4

執行 360度伺服馬達 腳位# 9

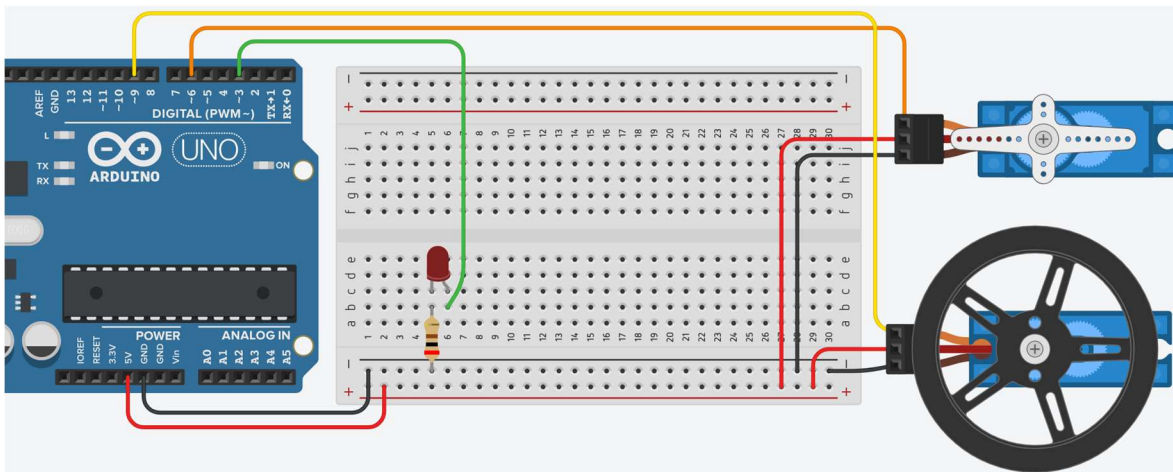
旋轉方向 逆時針
速度 快速

```

char Data=0;
#include <Servo.h>
Servo servo_6;
Servo servo_9;
void setup()
{
    Serial.begin(9600);
    pinMode(2, OUTPUT);
    servo_9.attach(9);
    servo_6.attach(6);
    servo_6.write(0);
    delay(0);
}

void loop()
{
    if (Serial.available() > 0) {
        Data = Serial.read();
        if (Data == 1) {
            digitalWrite(2,HIGH);
            servo_6.attach(6);
            servo_6.write(180);
            delay(0);
        } else if (Data == 0) {
            digitalWrite(2,LOW);
            servo_6.attach(6);
            servo_6.write(0);
            delay(0);
        } else if (Data == 2) {
            servo_9.write(40);
        } else if (Data == 3) {
            servo_9.write(90);
        } else if (Data == 4) {
            servo_9.write(140);
        }
    }
}

```



[Servo Library]

- attach(pin, min, max) : 連接伺服馬達的控制線給 Arduino 的一個 Pin 腳
pin: 在早期的 Arduino 只能指定 Pin 9 跟 Pin 10
min (選項): 對應於伺服上的最小 (0 度) 角度的脈衝寬度 (以微秒為單位) · 預設為 544
max (選項): 對應於伺服上的最大 (180 度) 角度的脈衝寬度 (微秒) · 預設為 2400
- write(angle) : 將值寫到伺服機相對應的控制軸
angle: 寫到伺服機的值 · 從 0 到 180
- read() : 讀取目前的伺服機值

```
#include <Servo.h>

int Data = 0;

Servo servo_6;
Servo servo_9;

void setup()
{
    servo_6.attach(6, 500, 2500);
    Serial.begin(9600);
    pinMode(2, OUTPUT);
    servo_9.attach(9, 500, 2500);
    servo_6.write(0);
}

void loop()
{
    if (Serial.available() > 0) {
        Data = Serial.available();
        if (Data == 1) {
            digitalWrite(2, HIGH);
            servo_6.write(180);
        } else {
            if (Data == 0) {
                digitalWrite(2, LOW);
                servo_6.write(0);
            } else {
                if (Data == 2) {
                    servo_9.write(0);
                } else {
                    if (Data == 3) {
                        servo_9.write(90);
                    } else {
                        if (Data == 4) {
```

```
        servo_9.write(180);
    }
}
}
}
}
Serial.println(Data);
}
delay(10); // Delay a little bit to improve simulation performance
}
```