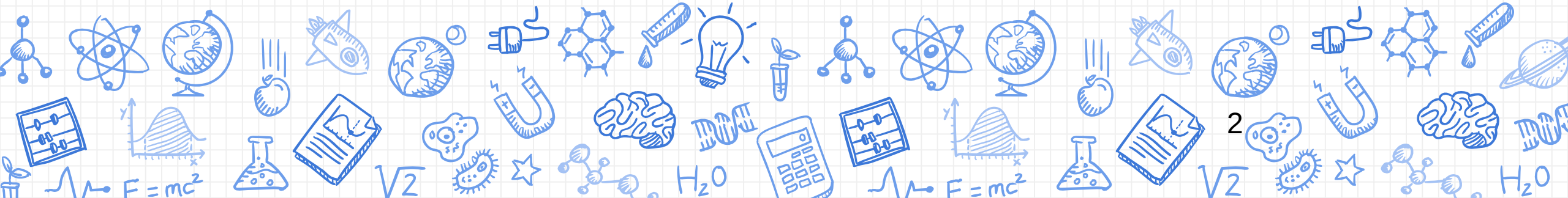
Arduino 電子元件控制練習與自走車安裝操作

Outline

- Arduino Reference
- Arduino通訊協定 - UART/I²C/SPI
- PWM - 直流馬達控制
- 伺服馬達控制
- 自走車組裝

Outline

- Arduino Reference
- Arduino通訊協定 - UART/I²C/SPI
- PWM - 直流馬達控制
- 伺服馬達控制
- 自走車組裝



Arduino Reference

- *https://www.arduino.cc/en/Reference/HomePage*

Language Reference

Arduino programs can be divided in three main parts: *structure*, *values* (variables and constants), and *functions*.

Structure

- setup()
- loop()

Control Structures

- if
- if...else
- for

Variables

Constants

- HIGH | LOW
- INPUT | OUTPUT | INPUT_PULLUP
- LED_BUILTIN
- true | false
- integer constants

Functions

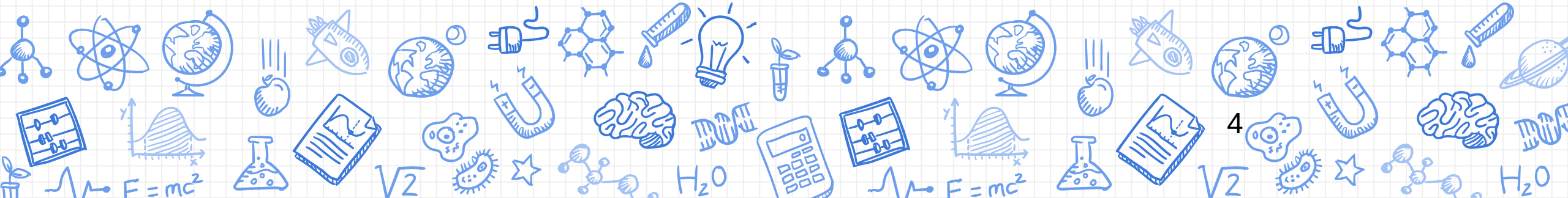
Digital I/O

- pinMode()
- digitalWrite()
- digitalRead()

Analog I/O

Outline

- Arduino Reference
- Arduino通訊協定 - UART/I²C/SPI
- PWM - 直流馬達控制
- 伺服馬達控制
- 自走車組裝



Universal Asynchronous Receiver/Transmitter

- 通用非同步收發傳輸器(Universal Asynchronous Receiver/Transmitter, UART)，例如：RS232。
- COM是電腦中異步串行通信口的簡寫，例如：COM1、COM3……等。
- Baud Rate：每秒內傳輸的訊息量，在兩個不同平台通訊時須特別注意同步。

- 補充：UART為開發版debug的好幫手。

Universal Asynchronous Receiver/Transmitter

- 在void setup()中先設定baud rate:

  Serial.begin(9600);

- 需要回傳資料時，再呼叫以下程式碼：

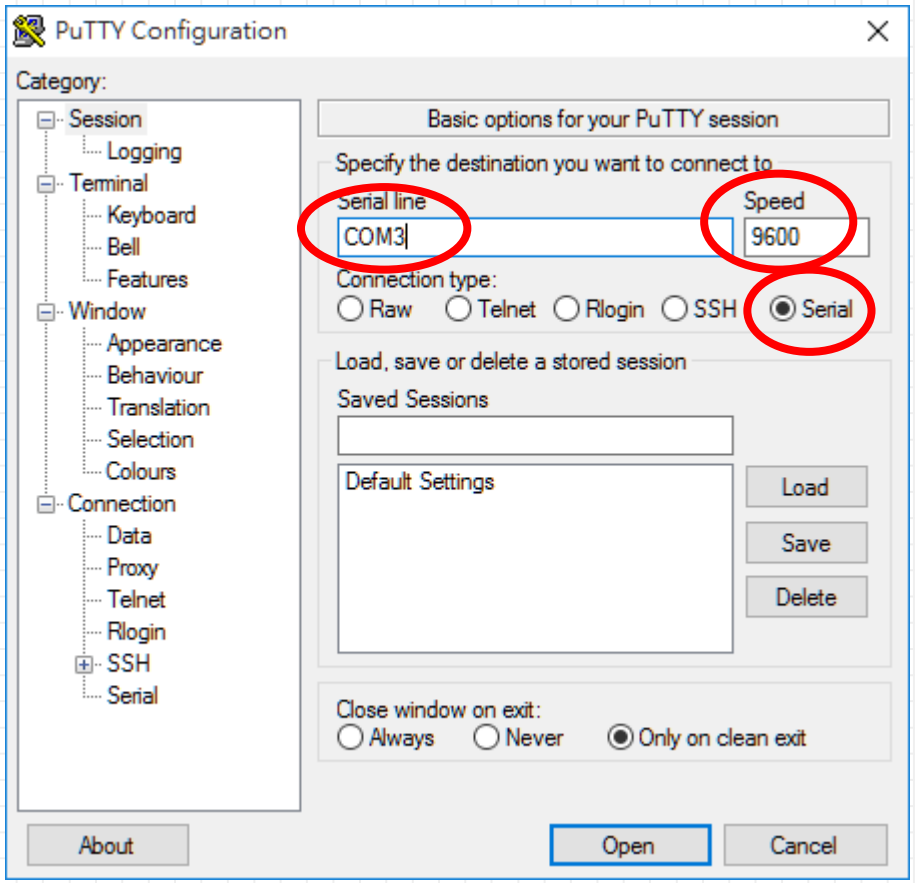
  Serial.print(“COM”);

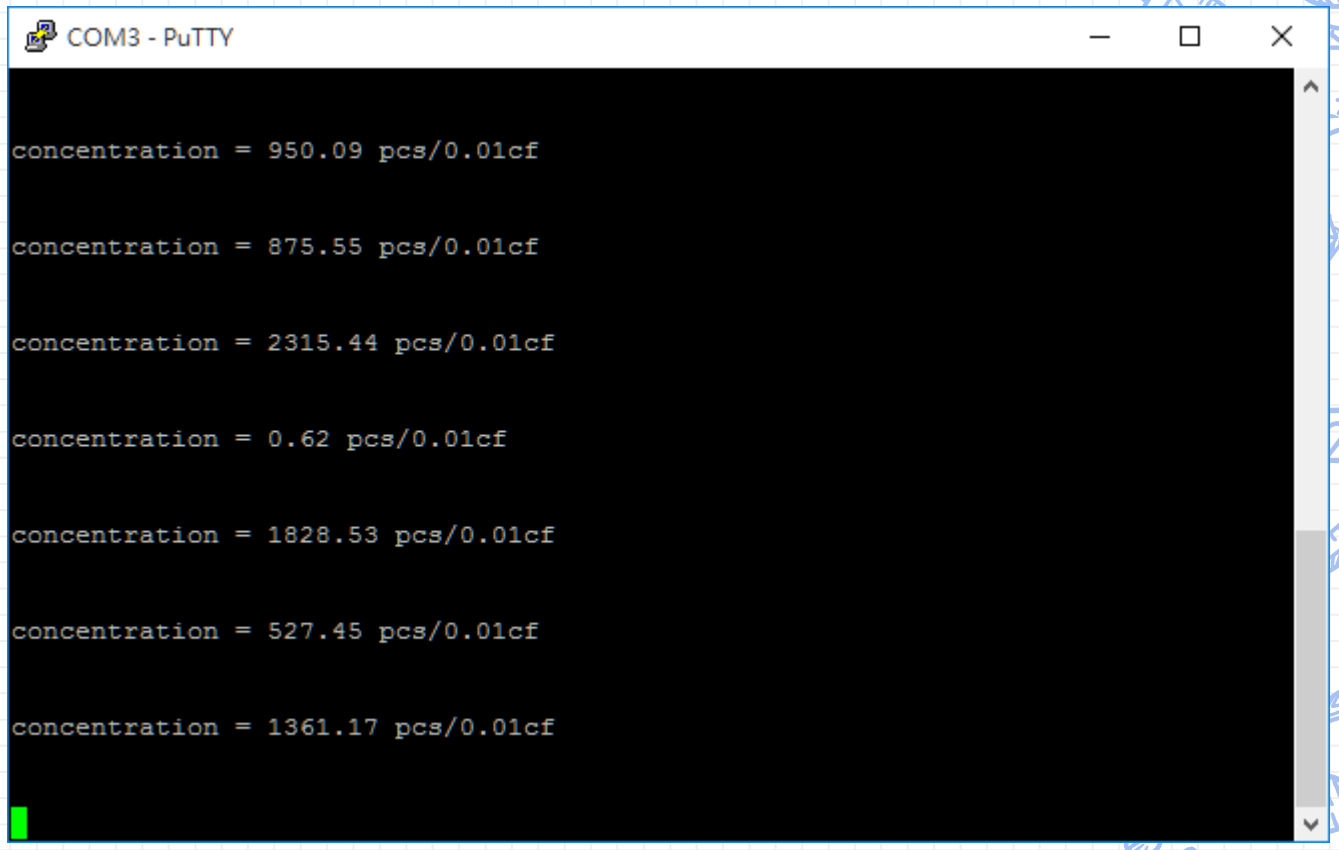
  Serial.println(“COM”);

```
void setup(){
  // 開啟 Serial port, 通訊速率為 9600 bps
  Serial.begin(9600);
}

void loop(){
  volt = analogRead(analogVoltageInput);
  volt *= (5.0 / 1023.0);

  Serial.print("Li-battery's Voltage = ");
  Serial.println(volt);

  delay(1000);
}
```

Universal Asynchronous Receiver/Transmitter

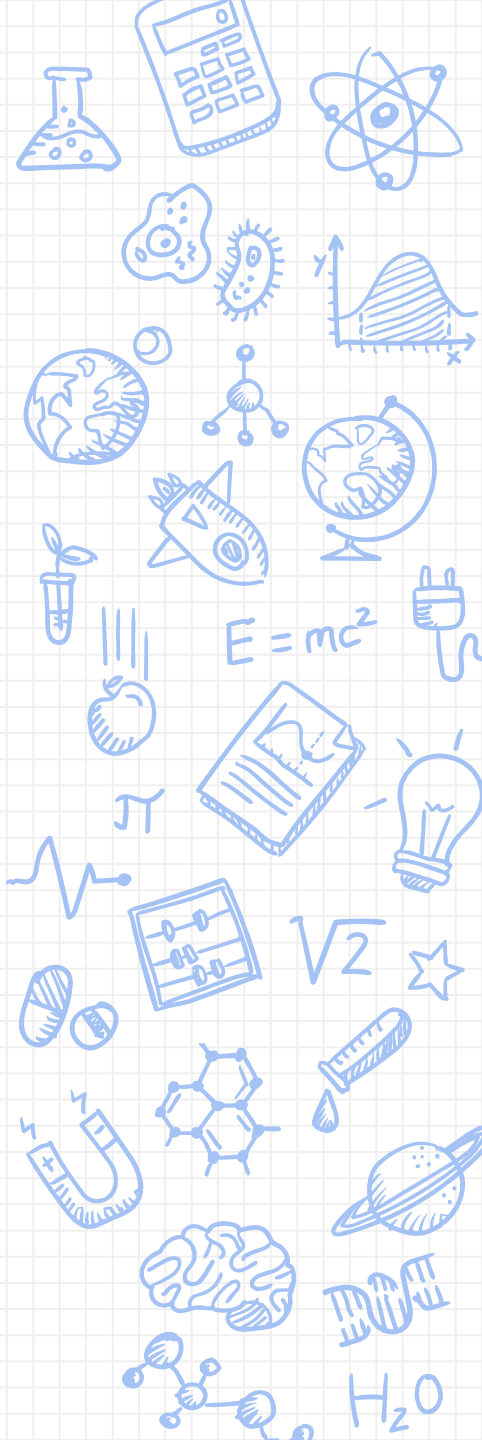
- Putty若成功設定，資料馬上可以從Arduino回傳。





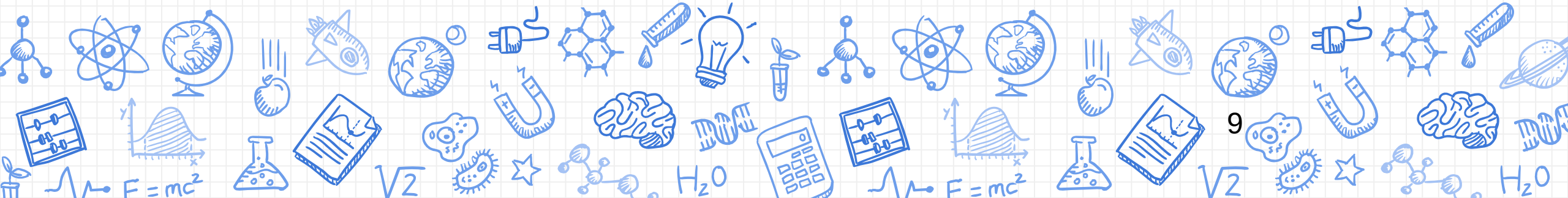
Universal Asynchronous Receiver/Transmitter

- Demo 1：使用任意starter kit提供的sensor，並以UART回傳偵測到的數值。
- 參考網址：http://www.seeedstudio.com/depot/Grove-Starter-Kit-for-LinkIt-ONE-p-2028.html



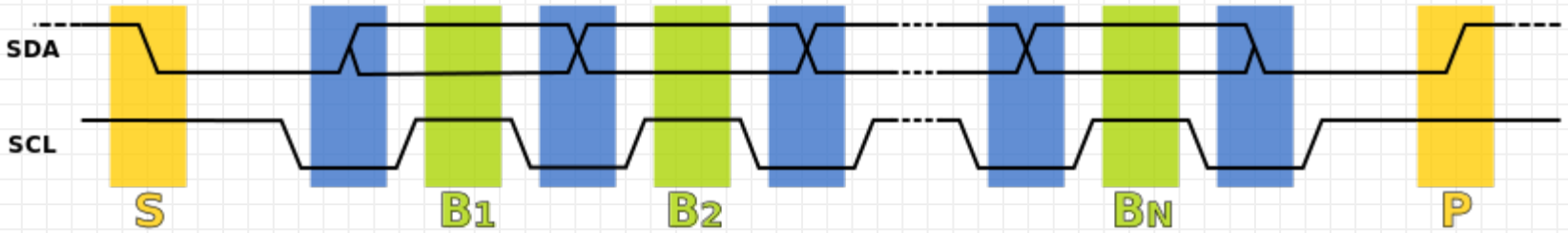
Outline

- Arduino Reference
- Arduino通訊協定 - UART/I²C/SPI
- PWM - 直流馬達控制
- 伺服馬達控制
- 自走車組裝



Inter-Integrated Circuit

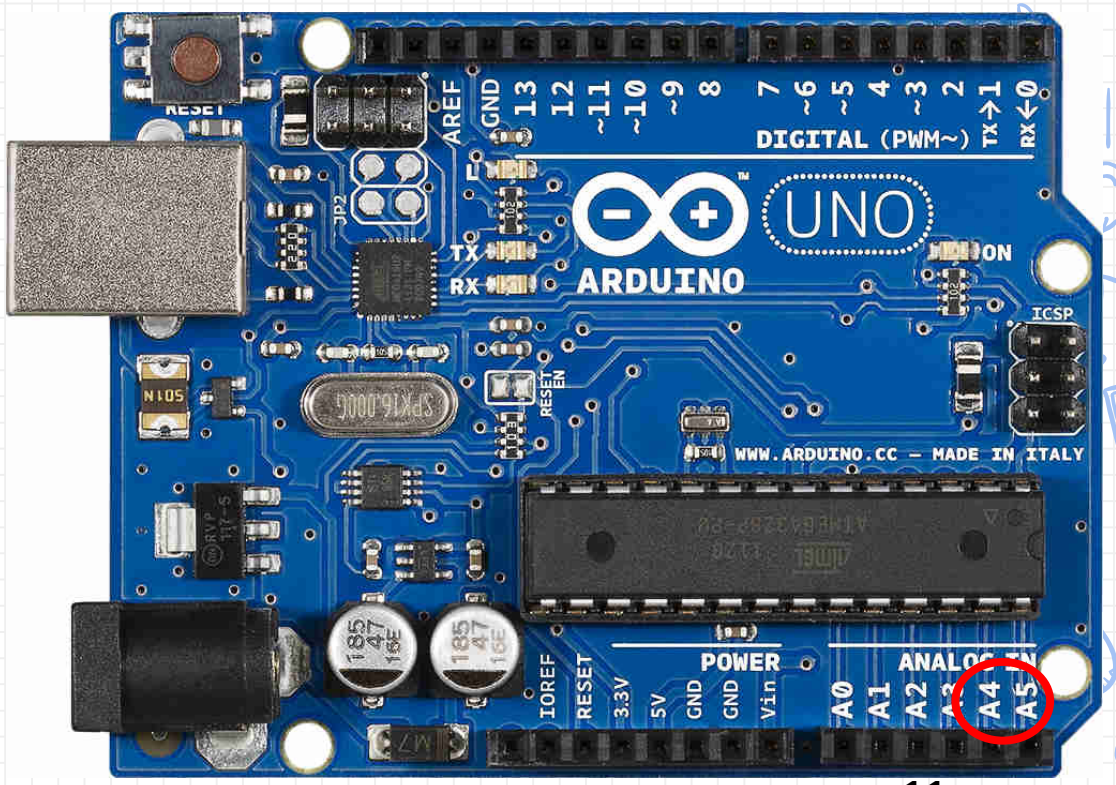
- 積體電路匯流排(Inter-Integrated Circuit, I^2C)，起先為飛利浦公司為了讓主機板、嵌入式系統或手機用以連接低速週邊裝置而發展。
- I^2C只使用兩條雙向漏極開路(Open Drain)：串列資料(SDA)、串列時脈(SCL)並利用電阻將電位上拉。



- 應用：與sensor通訊、與另一台Arduino溝通。

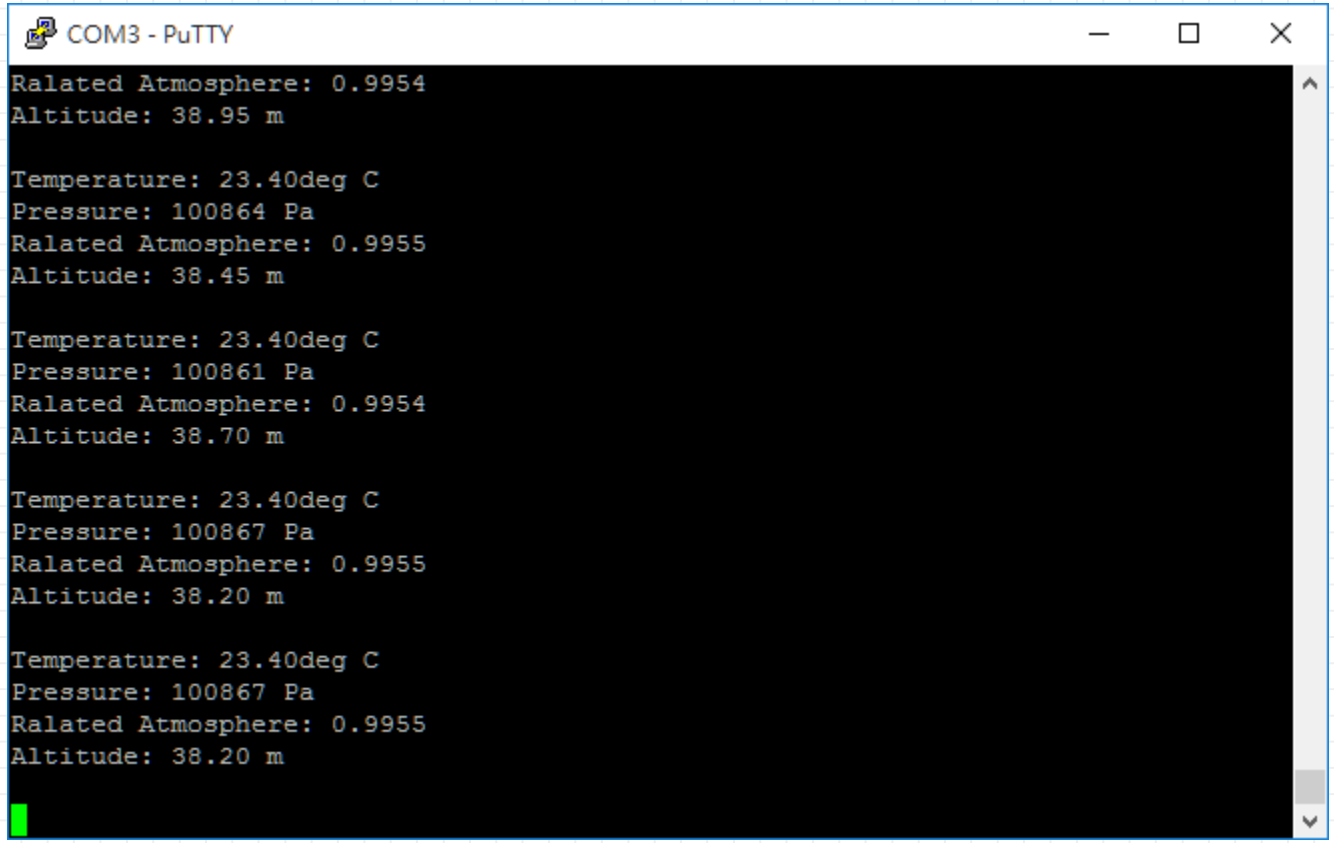
Inter-Integrated Circuit

- 例子：Barometer sensor即是以I²C的通訊管道與Arduino溝通。



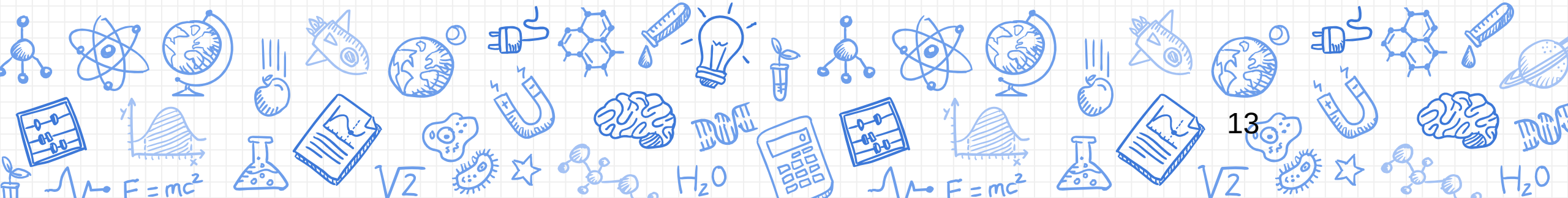
Inter-Integrated Circuit

- Barometer sensor



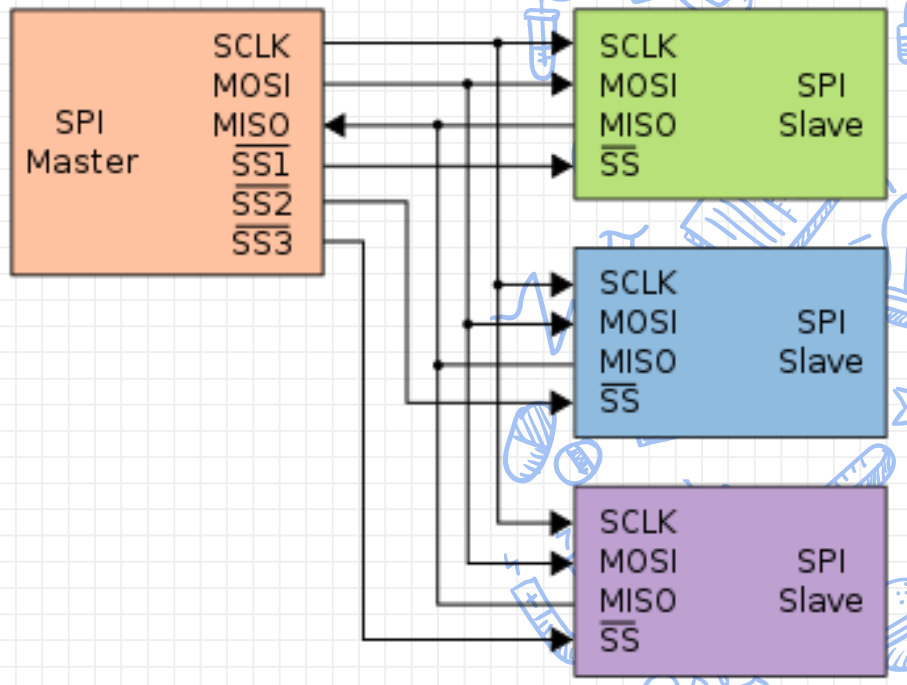
Outline

- Arduino Reference
- Arduino通訊協定 - UART/I²C/SPI
- PWM - 直流馬達控制
- 伺服馬達控制
- 自走車組裝



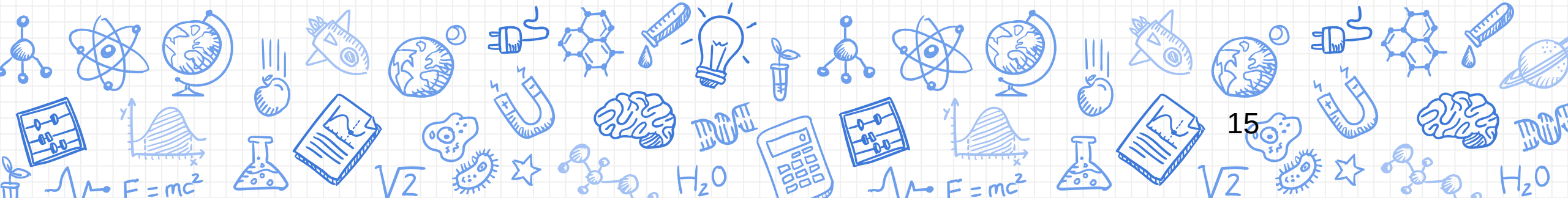
Serial Peripheral Interface Bus

- 串行外設介面(Serial Peripheral Interface Bus, SPI)，多用於一個主機(master)和一個或多個從機(slave)。
  - SCLK(Serial Clock)
  - MOSI(Master Output, Slave Input)
  - MISO(Master Input, Slave Output)
  - SS(Slave Selected)，低電壓選取



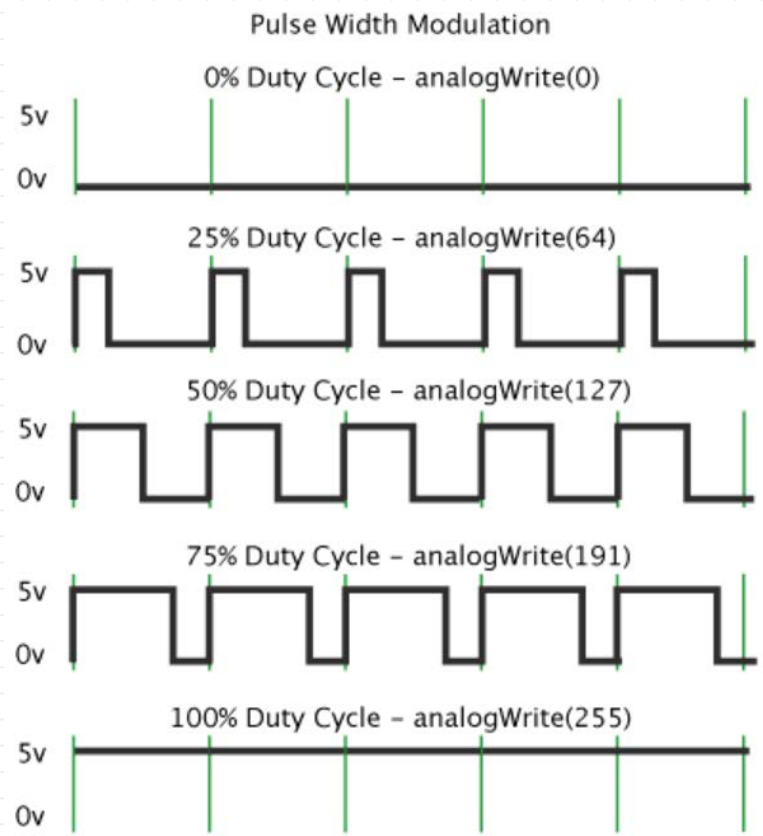
Outline

- Arduino Reference
- Arduino通訊協定 - UART/I²C/SPI
- PWM - 直流馬達控制
- 伺服馬達控制
- 自走車組裝



PWM - Pulse Width Modulation

- PWM是一種具有週期性的數位訊號，在Arduino中以analog output去對digital output做控制。



PWM - Pulse Width Modulation

- LED漸進式明暗

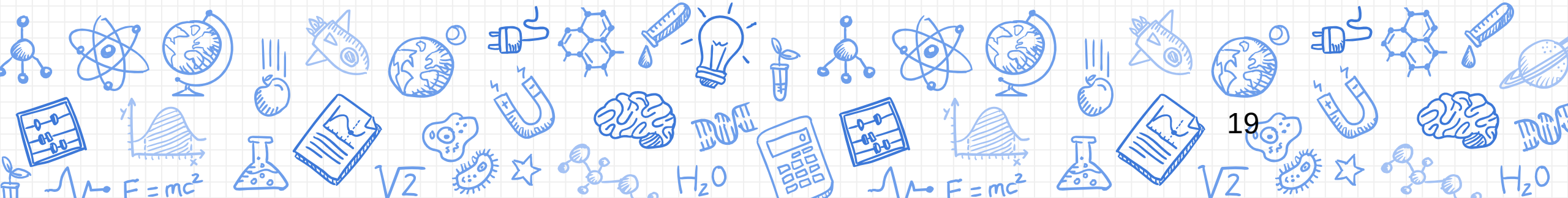
```
const int LEDPin = 13;

void setup() {
  pinMode(LEDPin, OUTPUT);
}

void loop() {
  // fade the LED from off to brightest:
  for (int brightness = 0; brightness < 255; brightness++) {
    analogWrite(LEDPin, brightness);
    delay(2);
  }
  // fade the LED from brithstest to off:
  for (int brightness = 255; brightness >= 0; brightness--) {
    analogWrite(LEDPin, brightness);
    delay(2);
  }
  // pause between LEDs:
  delay(100);
}
```

PWM - Pulse Width Modulation

- Demo 2：控制自走車直流馬達速度。
  漸進式加速
  慢速倒車
  右前方(左前方)轉彎

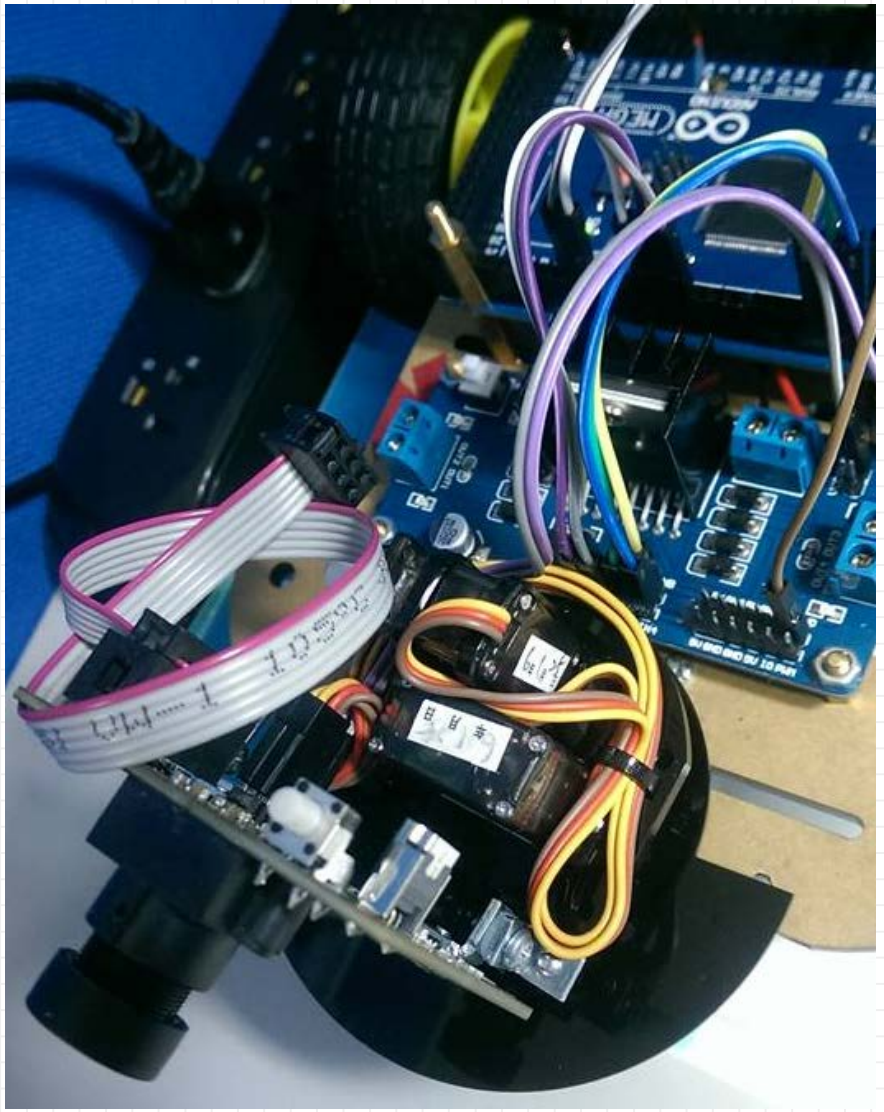
Outline

- Arduino Reference
- Arduino通訊協定 - UART/I²C/SPI
- PWM - 直流馬達控制
- 伺服馬達控制
- 自走車組裝



Servo馬達控制

- 伺服馬達比起步進馬達擁有更快的反應，由於可以急速啟動、急速停止，很精準的執行控制器給的訊號。
- 伺服馬達使用類比輸入做控制。
- 可使用範圍：
  0°~180°（此實驗）



Servo馬達控制

- myservo.attach(servoPin)，設定servo pin。
- myservo.write(45)，寫入角度，讓馬達轉到應有的位置。

```
#include <Servo.h>
Servo myservo;
const int servoPin = 51;

void setup() {
  myservo.attach(servoPin);
}

void loop() {
  myservo.write(45);
  delay(15);

  myservo.write(135);
  delay(15);
}
```

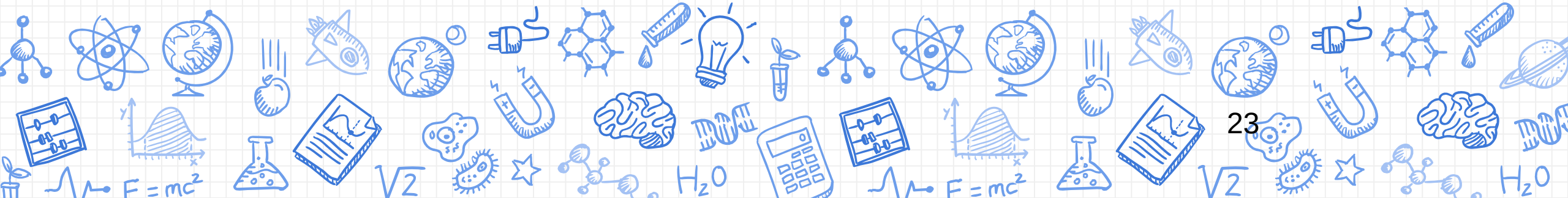
Servo馬達控制

- Demo 3-1：操作伺服馬達。
  隨機從0°~180°轉動，並記得每次隨機指定後，需要delay一段時間。

- Demo 3-2：操作伺服馬達。
  使用可變電阻，控制馬達位置。請記得不得超界，例如：電阻最小時對應到0°，電阻對大時對應到180°。

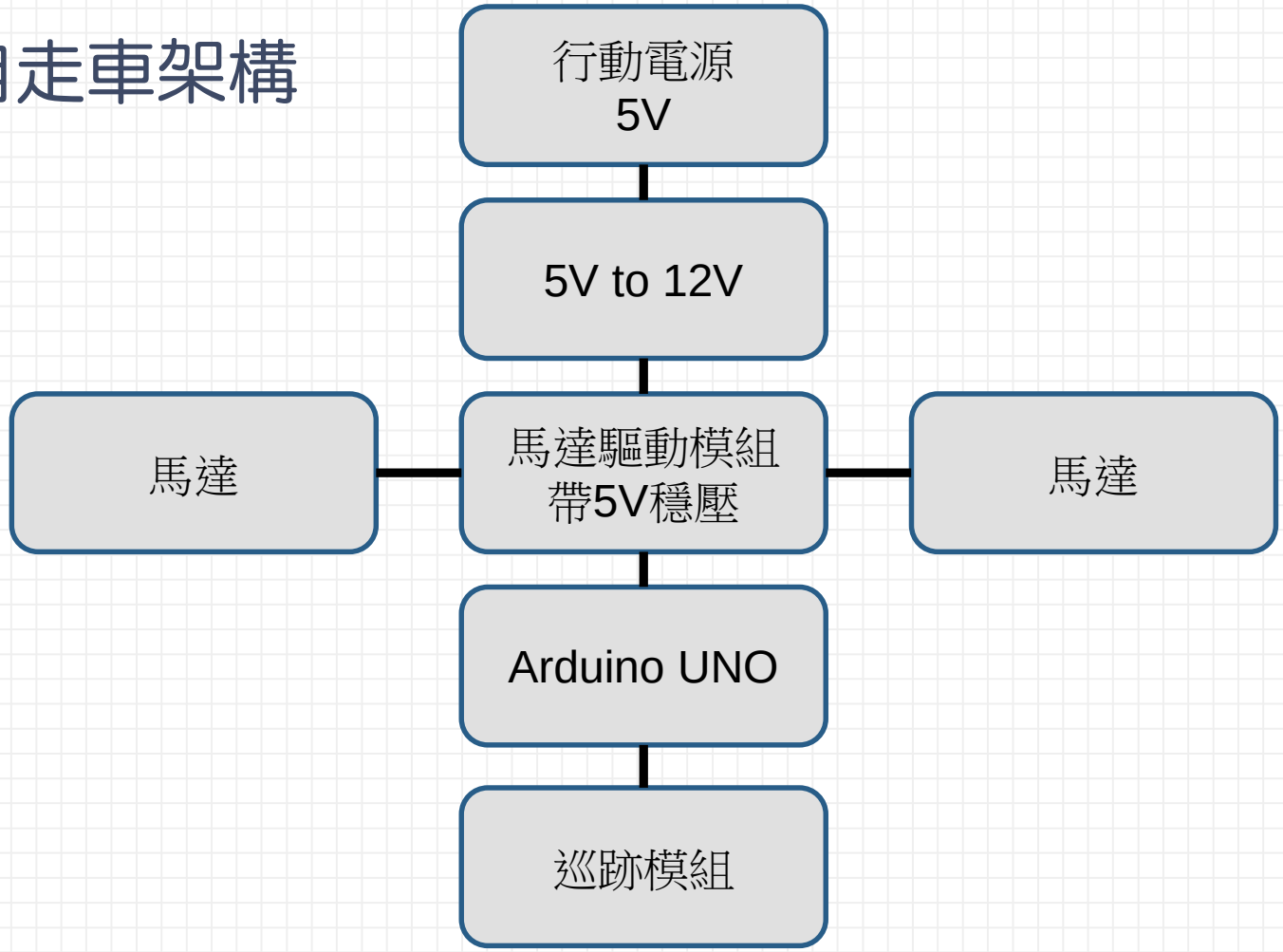
Outline

- Arduino Reference
- Arduino通訊協定 - UART/I²C/SPI
- PWM - 直流馬達控制
- 伺服馬達控制
- 自走車組裝



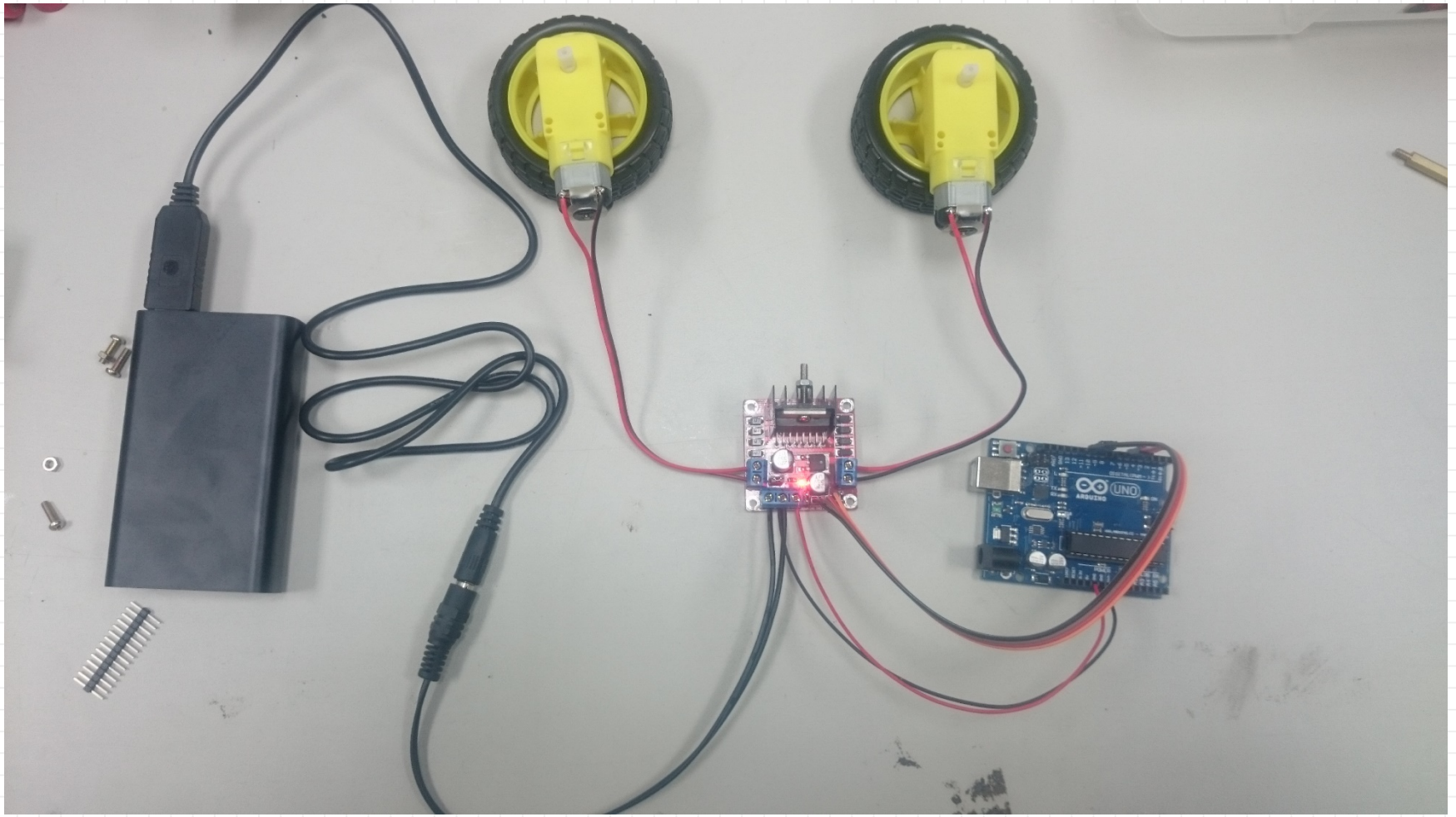
自走車組裝

- 自走車架構



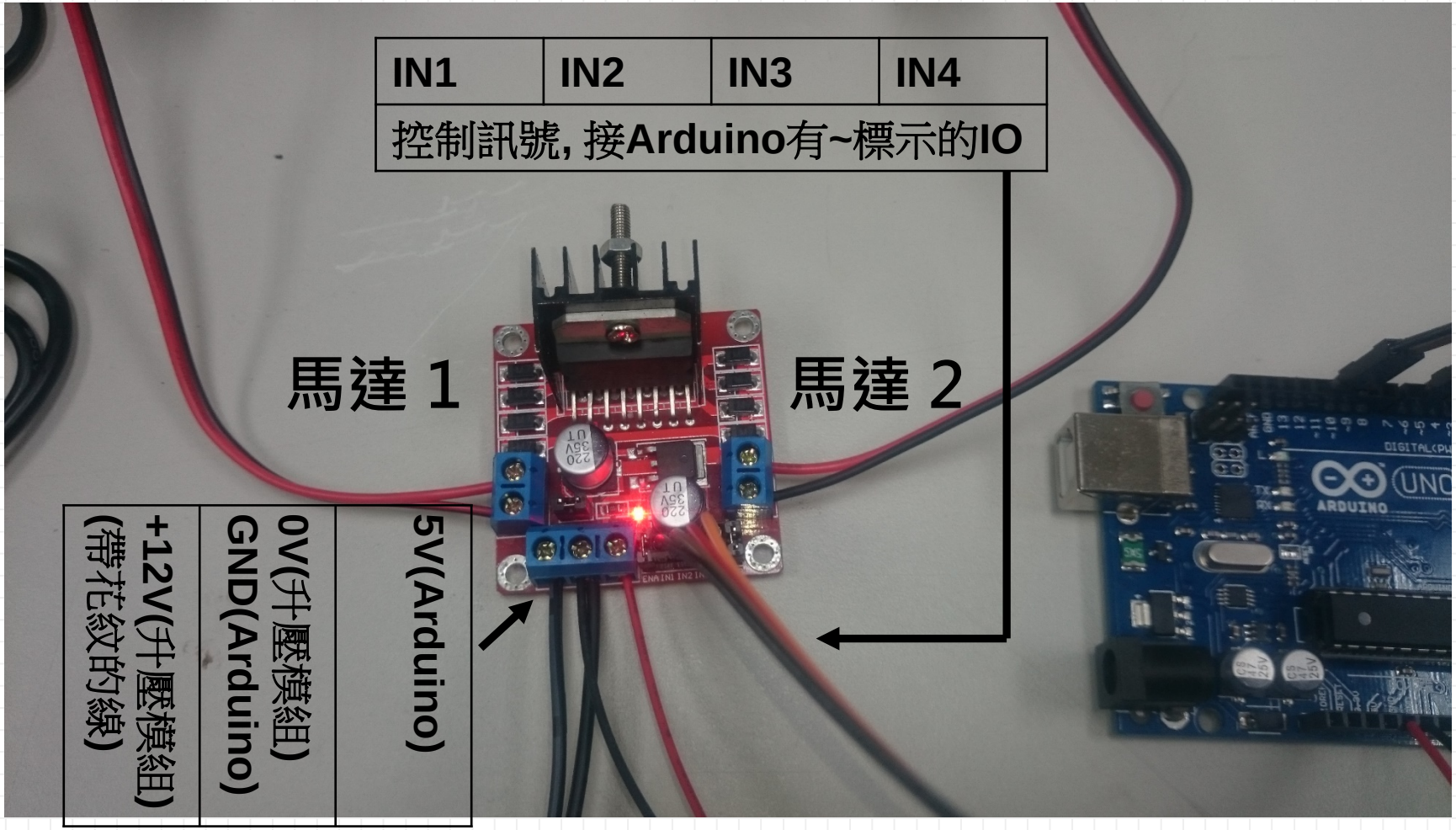
自走車組裝-零件圖

自走車組裝-配線示意圖



最後都要裝在車上！可以先組裝再接線

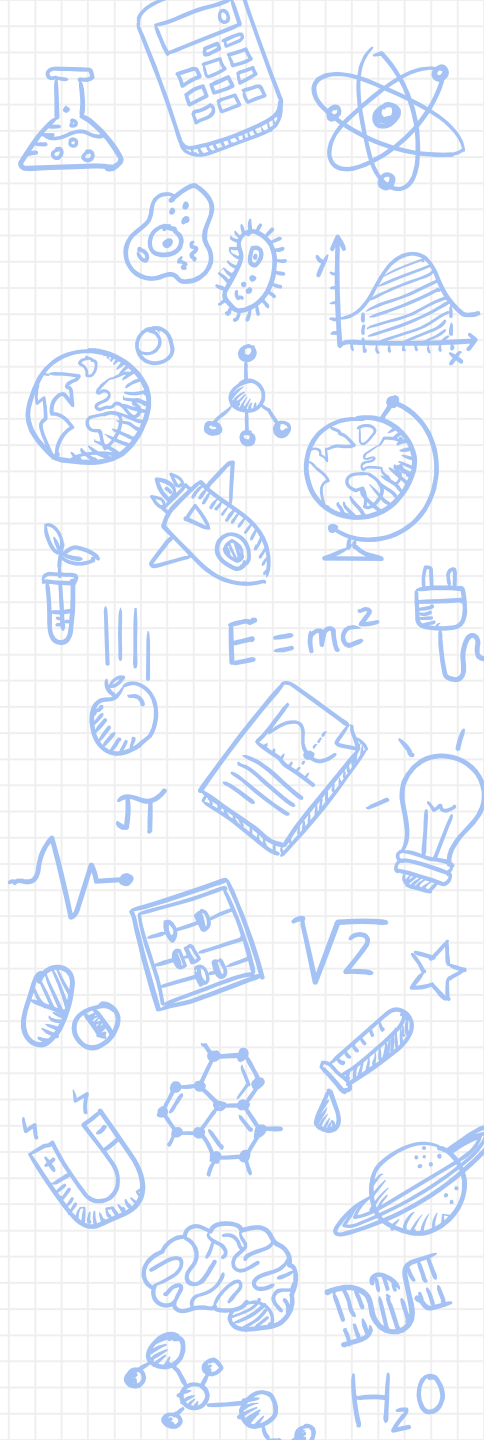
自走車組裝-接線特寫



最後都要裝在車上！可以先組裝再接線

自走車組裝

- Demo 4：熟悉巡跡模組。
  自走車超過停止線後，立即停止。



自走車專案開發

- 下一堂課(3/2)會準備基本軌道開放時間讓大家試車，以及開放補Demo。
- 下下堂課(3/9)將舉辦自走車應用競賽，各組人馬需在特定的軌道從起點走向終點。