

實驗八 Arduino 介紹與應用

8-1 課程目的與規劃

本實驗規劃的想法，是學生在進行本實驗時，可以接觸並且學習這四大領域，並針對子項目有動手做的經驗，如此在日後學習或者研究上可以有更深刻的體會。



Figure 1 本實驗所觸及到的主題

本實驗開設在機工實驗中的電子實驗，一次上課規劃為3個半小時，課程行的流程如下。

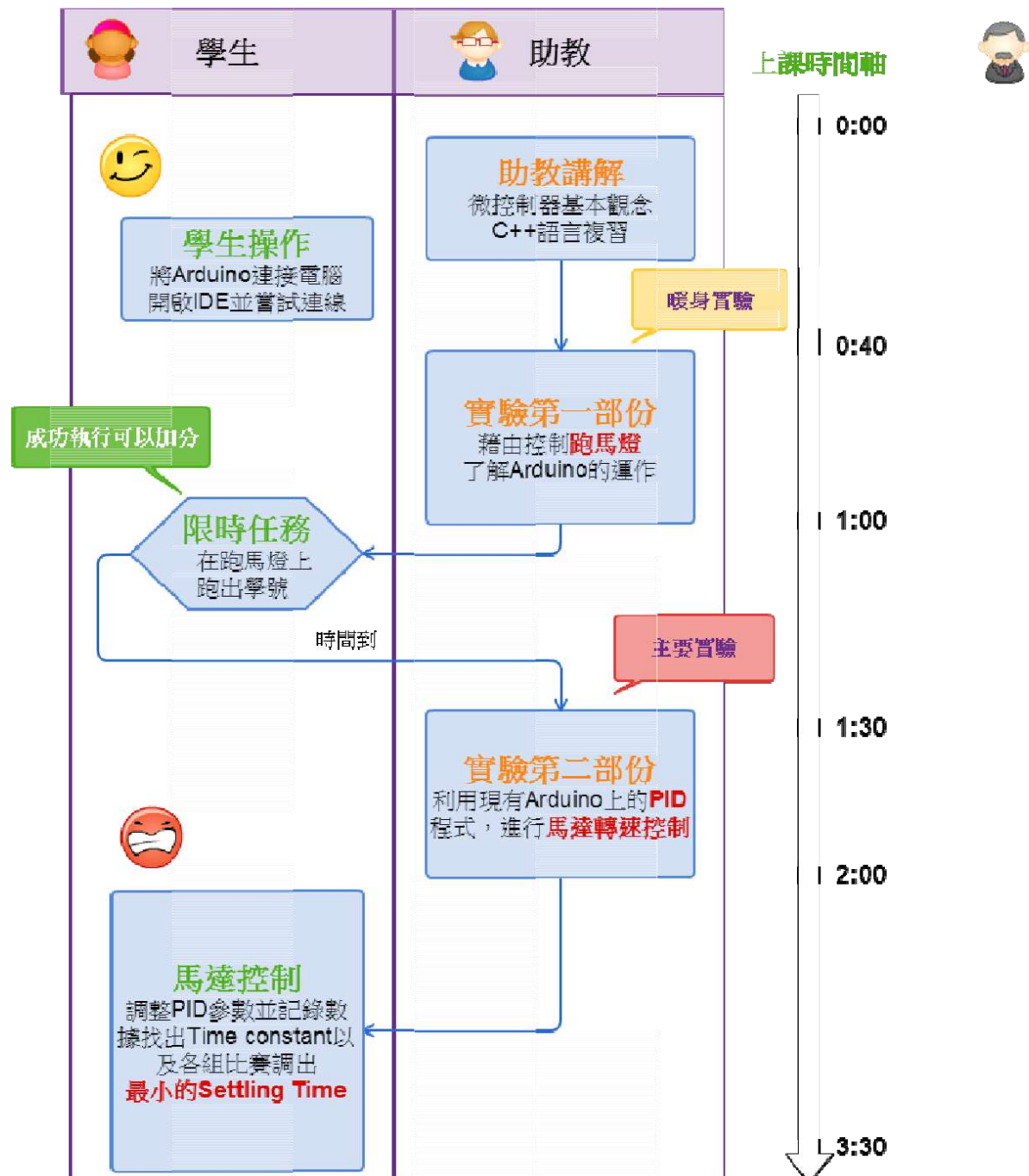


Figure 2 課程進行流程圖

8-2 基本原理

1. 什麼是Arduino 微控制器

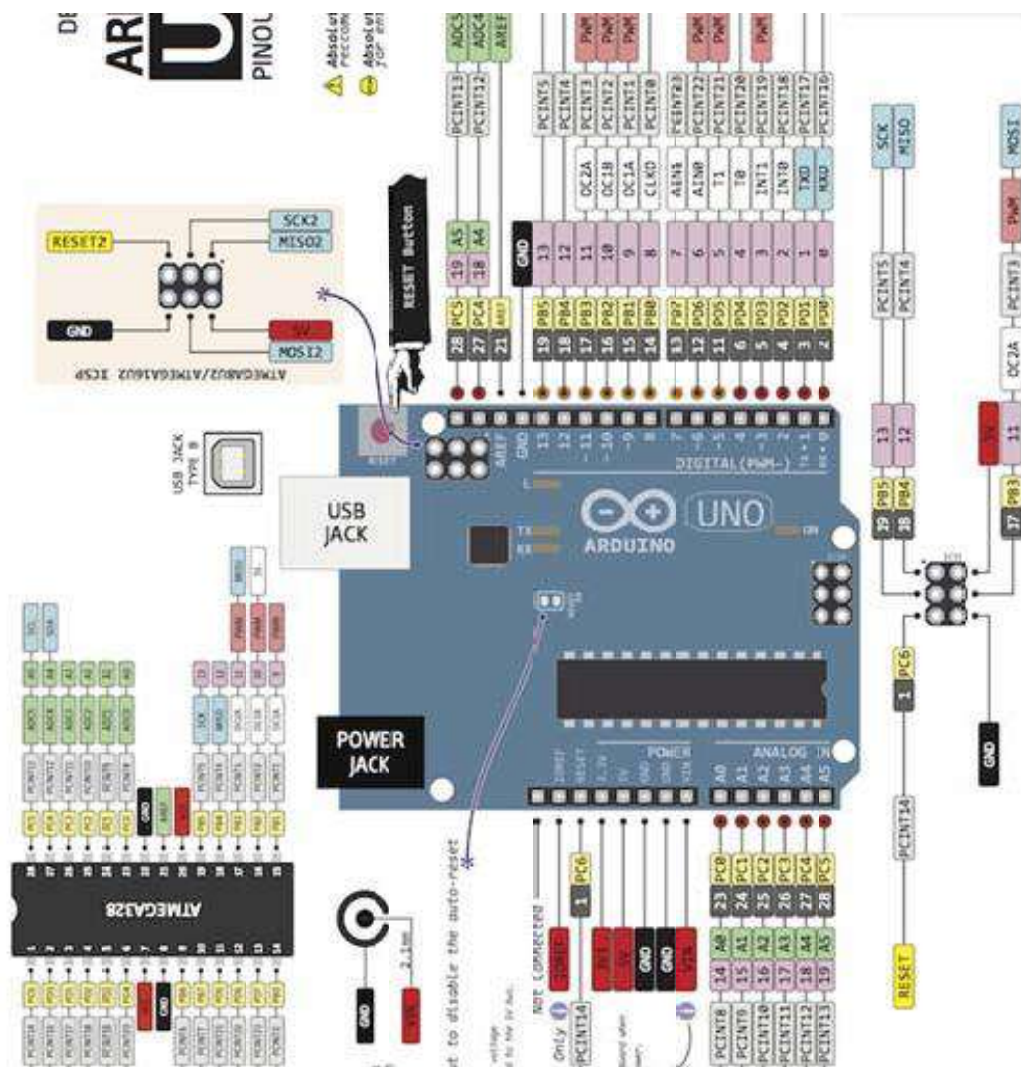


Figure 3 Arduino 基本介紹

2. 開發環境

同學可以上 Arduino 的網站下載 IDE(整合開發環境)



Figure 4 Arduino 官方網站

Arduino IDE

Arduino 1.0.5

Download

Arduino 1.0.5 (release notes), hosted by Google Code:

NOTICE: Arduino Drivers have been updated to add support for Windows 8.1, you can download the updated IDE (version 1.0.5-r2 for Windows) from the download links below.

- Windows Installer, Windows ZIP file (for non-administrator install)
- Mac OS X
- Linux: 32 bit, 64 bit
- source

Next steps:

- Getting Started
- Reference
- Environment
- Examples
- Foundations
- FAQ

Arduino 1.5.6-r2 BETA (with support for Arduino Yún and Arduino Due boards)

If you have the Arduino Yún or Due you must download the 1.5.6 version. Refer to the Yún getting started page, or Due getting started page for specific details about those boards.

WARNING: version 1.5.6 may cause problems with the serial monitor, the problem is fixed in version 1.5.6-r2 that can be downloaded below.

WARNING: This software is a beta version, you may encounter bugs or unexpected behaviours. Please discuss any issues in the Yún forum or Due forum

Download

Arduino 1.5.6-r2 (release notes):

Figure 5 Arduino IDE

由於 Arduino 是開源專案(Open Source Project)，網站上甚至可以下載到所有的原始程式碼，Arduino 所用的程式語言語法類似於 C/C++，而且 Arduino IDE 是跨平台的，有 Windows, Macintosh OSX 和 Linux 的版本。Arduino IDE 的軟體介面如下：

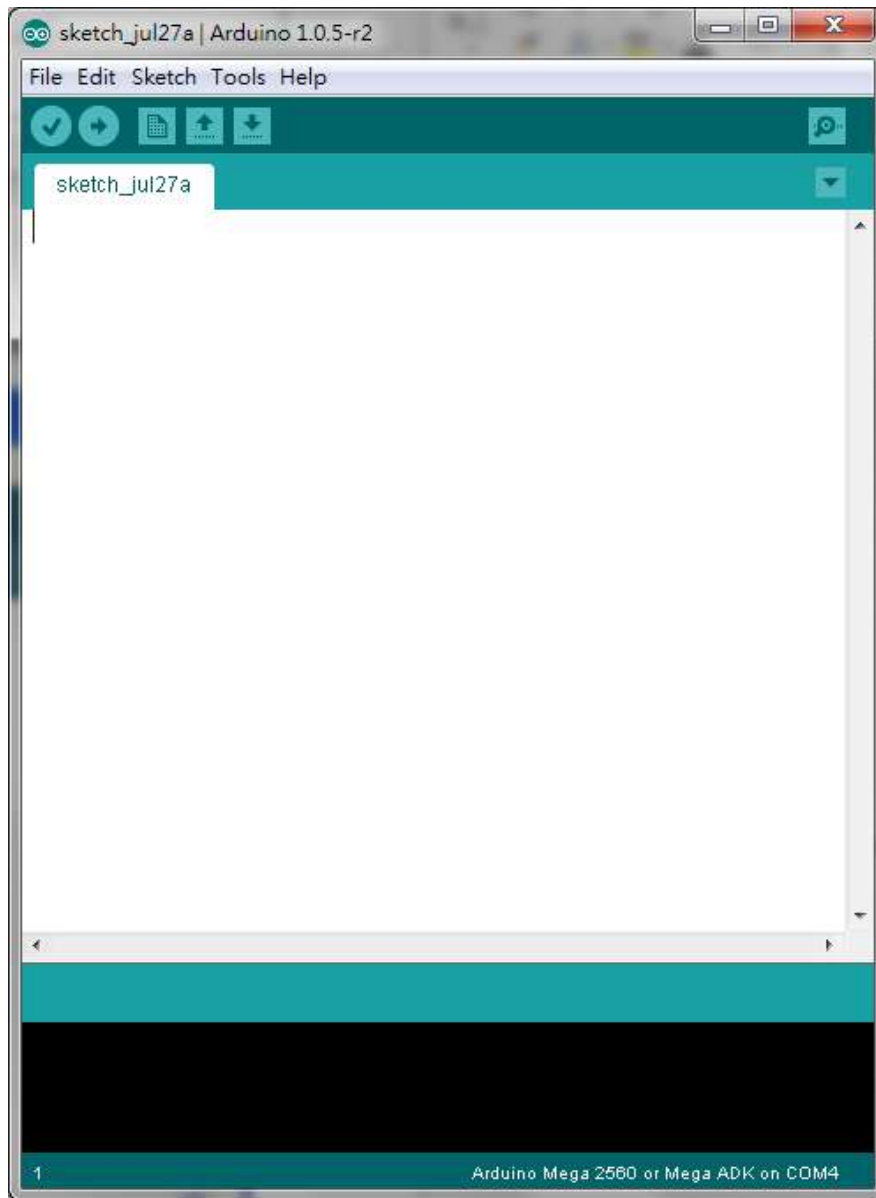
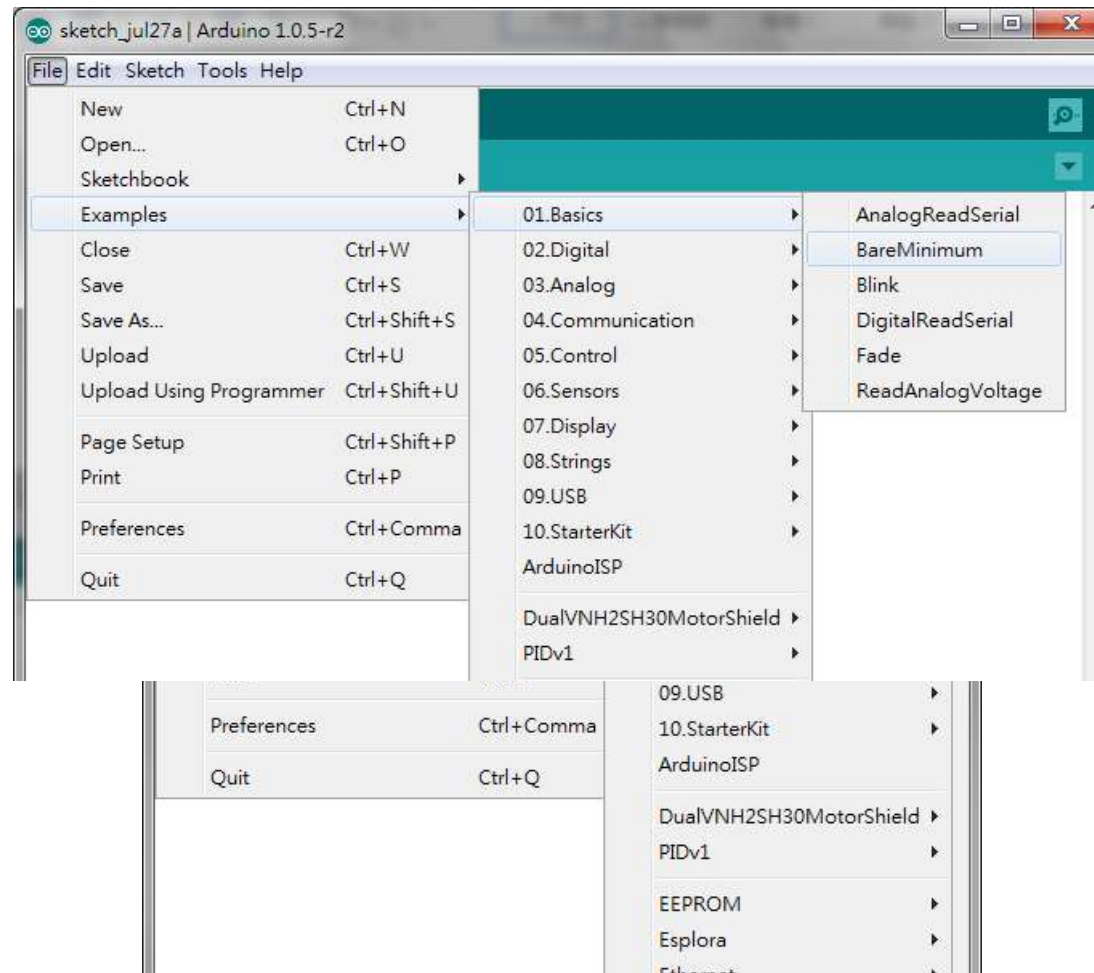


Figure 6 Arduino IDE 介面

Arduino 的程式叫作 Sketch，意為腳本，Arduino 程式主要由 setup() 和 loop() 這兩個函式組成：

```
1 void setup() {  
2   // setup 函式只會跑一次  
3 }  
4  
5 void loop() {  
6   // loop 函式會不斷的執行  
7 }
```

因為 `setup()` 和 `loop()` 是每支 Arduino 程式都會用到的兩個函式，
Arduino IDE 已經準備好程式骨架，我們在寫 Arduino 程式的時候，可以直接點選 `File > Examples > 1.Basics > BareMinimum` 這個範本程式檔。



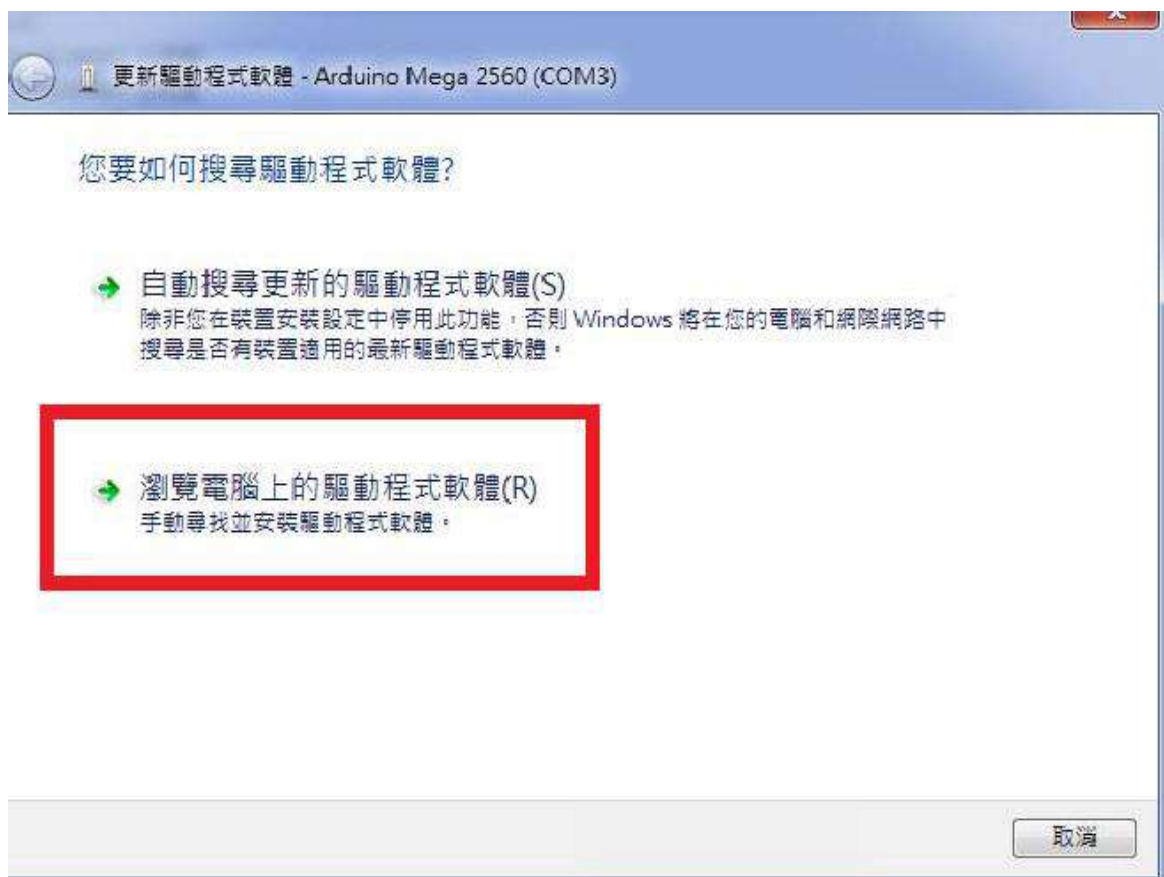
此外 Arduino IDE 已經準備多種程式範例，可以直接點選 `File > Examples`，我們在寫 Arduino 程式的時候，可先選擇所要使用功能的範例程式。

在 windows 作業系統下，須至型安裝 USB 裝置的驅動程式(Linux 下則不用)，步驟如下。

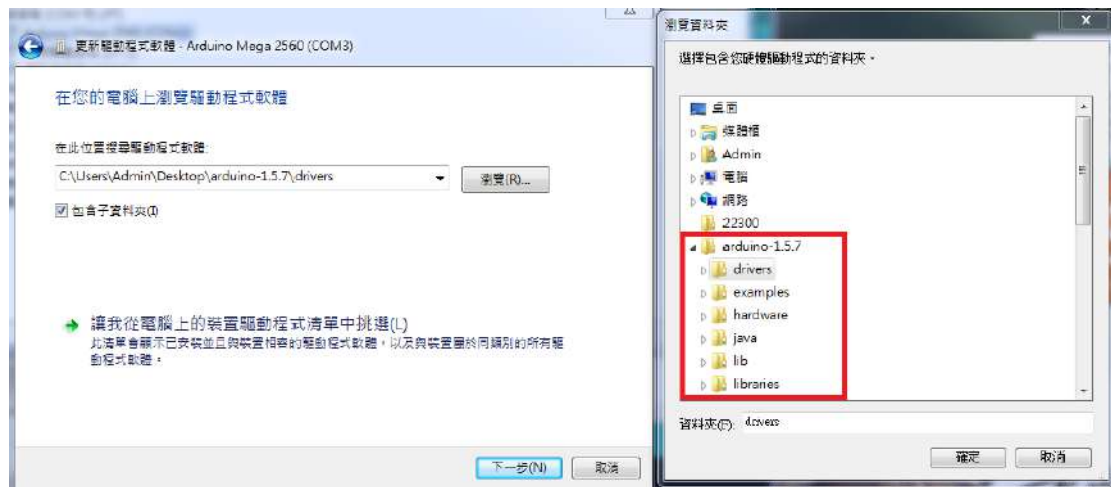
1) 在裝置管理員中找尋無法驅動的硬體



2) 利用手動搜尋，尋找驅動程式。



3) FTDI 驅動程式放置於arduino 下載的資料夾中。



8-3 實驗原理

實驗一:跑馬燈控制

1. MAX7219 顯示模組

MAX7219 是一種集成化的串列輸入/輸出共陰極顯驅動器,連接微處理器與 8 位元數位
的 7 段數位 LED 顯示,也可以連接條線圖顯示器或者 64 個獨立的 LED。其上包括一個片
上的 B 型 BCD 編碼器、多路掃描回路,段字驅動器,而且還有一個 8*8 的靜態 RAM 用來存
儲每一個資料。只有一個外部寄存器用來設置各個 LED 的段電流。

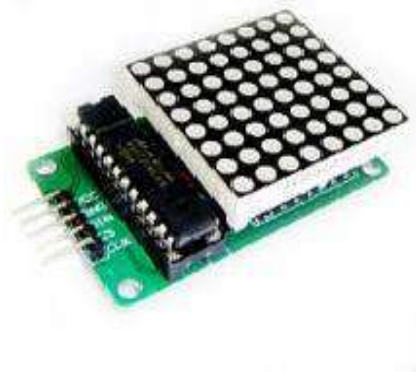


Figure 7 MAX7219 顯示模

模組參數：

1. 單個模組可以驅動一個 8*8 共陰點陣
2. 模組工作電壓：5V
3. 模組尺寸：長 5 厘米 X 寬 3.2 厘米 X 高 1.5 厘米
4. 帶 4 個固定螺絲孔，孔徑 3mm，可使用本店 M3 銅柱固定
5. 模組帶輸入輸出接口

接線說明：

1. 模組左邊為輸入端口，右邊為輸出端口。
2. 控制單個模組時，只需要將輸入端口接到 CPU
3. 多個模組級聯時，第 1 個模組的輸入端接 CPU，輸出端接第 2 個模組的輸入端，第 2 個模組的輸出端接第 3 個模組的輸入端，以此類推...

2. MAX7219顯示驅動器

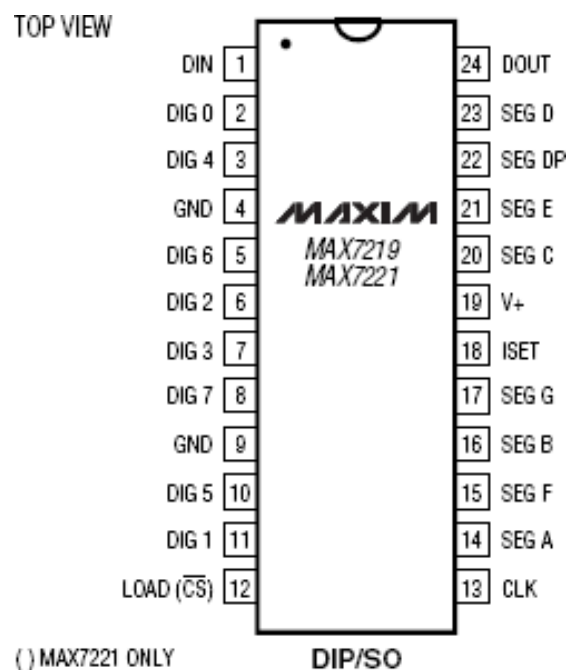


Figure8 MAX7219 腳位配置

Table 1 腳位說明

腳位	名稱	功能
1	DIN	串行數據輸入端口
4,9	GND	地線
12	CS	該端為低電壓時，串行數據被載入移位寄存器。
13	CLK	時鐘序列輸入端，最大速率為 10MHz
19	V+	正極電壓輸入，+5V

Table 2 列出 14 個可尋址的數據寄存器和控制寄存器，數據寄存器可以直接尋址，所以只要在 V+大於 2V 的情況下，每個數據皆可以獨立的修改或保存。控制寄存器包括編碼模式、顯示亮度、掃描限制、關閉模是以及顯示檢測五個寄存器。

Table 2 數據寄存器與控制寄存器

REGISTER	ADDRESS					HEX CODE
	D15–D12	D11	D10	D9	D8	
No-Op	X	0	0	0	0	0xX0
Digit 0	X	0	0	0	1	0xX1
Digit 1	X	0	0	1	0	0xX2
Digit 2	X	0	0	1	1	0xX3
Digit 3	X	0	1	0	0	0xX4
Digit 4	X	0	1	0	1	0xX5
Digit 5	X	0	1	1	0	0xX6
Digit 6	X	0	1	1	1	0xX7
Digit 7	X	1	0	0	0	0xX8
Decode Mode	X	1	0	0	1	0xX9
Intensity	X	1	0	1	0	0xXA
Scan Limit	X	1	0	1	1	0xXB
Shutdown	X	1	1	0	0	0xXC
Display Test	X	1	1	1	1	0xFF

Table 3 列出譯碼控制寄存器的格式。譯碼控制寄存器用來設置對每個數據進行 B 型 BCD 譯碼或不譯碼。

Table 3 Decode-Mode Register Examples (Address (Hex) = 0xX9)

DECODE MODE	REGISTER DATA								HEX CODE
	D7	D6	D5	D4	D3	D2	D1	D0	
No decode for digits 7–0	0	0	0	0	0	0	0	0	0x00
Code B decode for digit 0 No decode for digits 7–1	0	0	0	0	0	0	0	1	0x01
Code B decode for digits 3–0 No decode for digits 7–4	0	0	0	0	1	1	1	1	0x0F
Code B decode for digits 7–0	1	1	1	1	1	1	1	1	0xFF

Table 4 Decode-Mode Register Examples (Address (Hex) = 0xX9)

DUTY CYCLE		D7	D6	D5	D4	D3	D2	D1	D0	HEX CODE
MAX7219	MAX7221									
1/32 (min on)	1/16 (min on)	X	X	X	X	0	0	0	0	0x0
3/32	2/16	X	X	X	X	0	0	0	1	0x1
5/32	3/16	X	X	X	X	0	0	1	0	0x2
7/32	4/16	X	X	X	X	0	0	1	1	0x3
9/32	5/16	X	X	X	X	0	1	0	0	0x4
11/32	6/16	X	X	X	X	0	1	0	1	0x5
13/32	7/16	X	X	X	X	0	1	1	0	0x6
15/32	8/16	X	X	X	X	0	1	1	1	0x7
17/32	9/16	X	X	X	X	1	0	0	0	0x8
19/32	10/16	X	X	X	X	1	0	0	1	0x9
21/32	11/16	X	X	X	X	1	0	1	0	0xA
23/32	12/16	X	X	X	X	1	0	1	1	0xB
25/32	13/16	X	X	X	X	1	1	0	0	0xC
27/32	14/16	X	X	X	X	1	1	0	1	0xD
29/32	15/16	X	X	X	X	1	1	1	0	0xE
31/32	15/16 (max on)	X	X	X	X	1	1	1	1	0xF

實驗二:PID 馬達轉速控制

1. 直流馬達

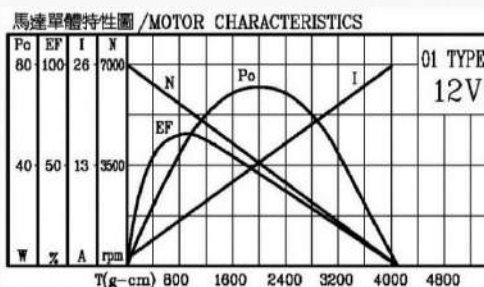
直流馬達一般是指直流有刷電動機，好處為控速簡單容易，只須控制電壓大小，便可以控制轉速。本實驗使用的馬達是 IG-42，適用於輪型機器人使用的減速馬達，提供高扭力與穩定的轉速。



Figure 9 直流馬達外觀

馬達在工作電壓 12 伏特時轉速為 120 RPM (減速過後)，馬達規格如下表格。

GEARED MOTOR TORQUE/SPEED																					
	減速比 Reduction ratio	$\frac{1}{4}$	$\frac{1}{14}$	$\frac{1}{17}$	$\frac{1}{24}$	$\frac{1}{49}$	$\frac{1}{61}$	$\frac{1}{84}$	$\frac{1}{104}$	$\frac{1}{144}$	$\frac{1}{212}$	$\frac{1}{294}$	$\frac{1}{504}$	$\frac{1}{624}$	$\frac{1}{720}$	$\frac{1}{864}$	$\frac{1}{1062}$	$\frac{1}{1470}$	$\frac{1}{2500}$	$\frac{1}{3000}$	$\frac{1}{3600}$
12V	定格扭力(Kg-cm) Rated torque	2.2	6.5	8.1	10	18	18	18	20	20	25	25	30	30	30	30	30	30	30	30	30
	定格回轉數(rpm) Rated speed	1400	405	325	248	120	98	76	63	45	31	24	13.5	10.9	9.5	8.0	6.5	4.6	2.7	2.3	1.9
馬達單體型式 / MOTOR DATA																					
定格電壓 Rated volt (V)		定格扭力 Rated torque (g-cm)		定格回轉數 Rated speed (rpm)		定格電流 Rated current (mA)		無負荷回轉數 No load speed (rpm)		無負荷電流 No load current (mA)		定格出力 Rated output (W)		重 量 Weight (g)							
12		700		5700		≤ 5500		7000		≤ 900		41.3		360							



未來的實驗可以考慮加上負載，觀察扭力與轉速的關係。

Figure 10 馬達扭力與轉速規格

2. 利用 PWM 訊號調整轉速

為控制馬達轉速，需要對直流電壓的大小和方向進行控制。目前，最常用的方式是利用 PWM 訊號來調整等效電壓。

所謂 PWM (Pulse Width Modulation - 脈波寬度調變) 信號就是一連串可以調整脈波寬度的信號。脈寬調變是一種調變或改變某個方波的簡單方法。

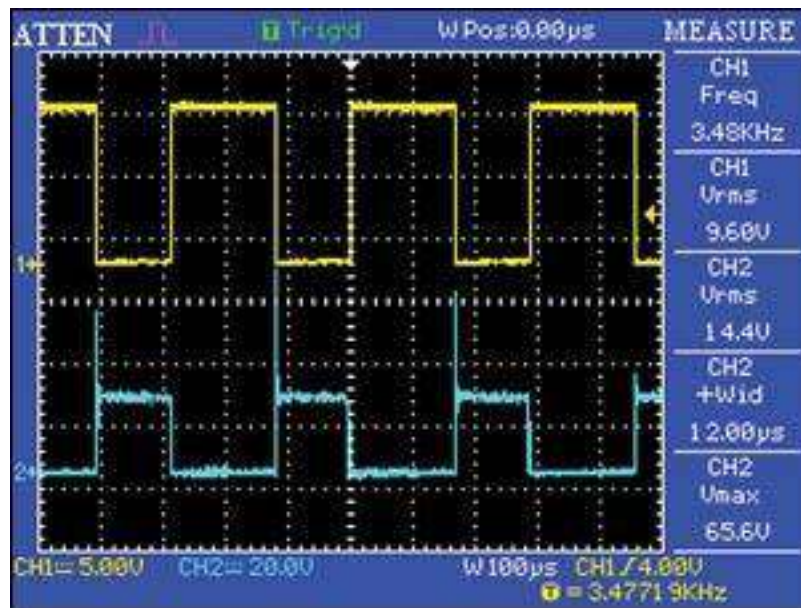
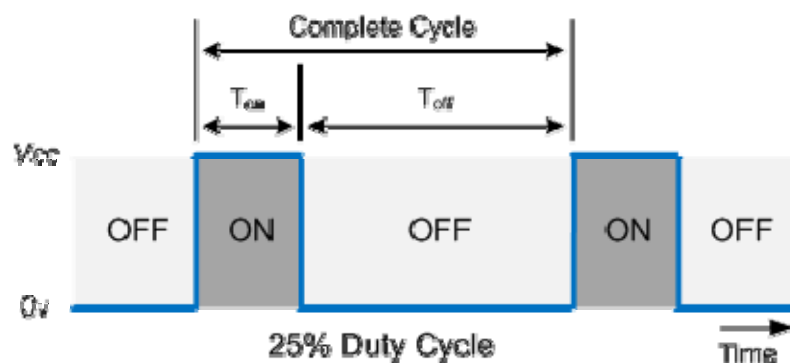


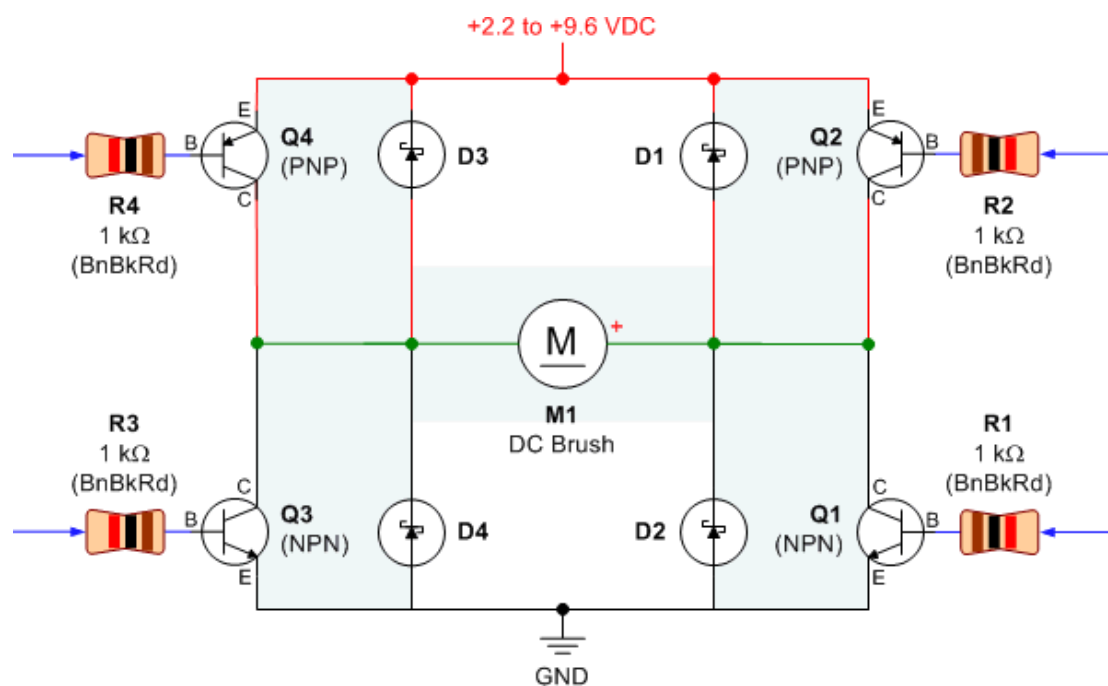
Figure 11 PWM 在示波器中的樣子

在它的基本形式上，方波的工作周期 (Duty Cycle) 是根據輸入信號的變化而變化。



3. 利用 H-Bridge 電路控制轉動方向

H-Bridge 原理



本實驗中使用可以與 Arduino 搭配的 H-Bridge 模組(Arduino Shield) ,

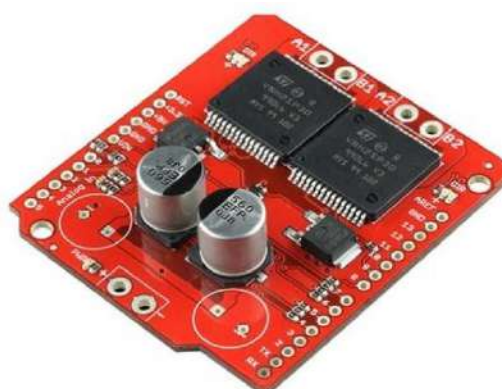
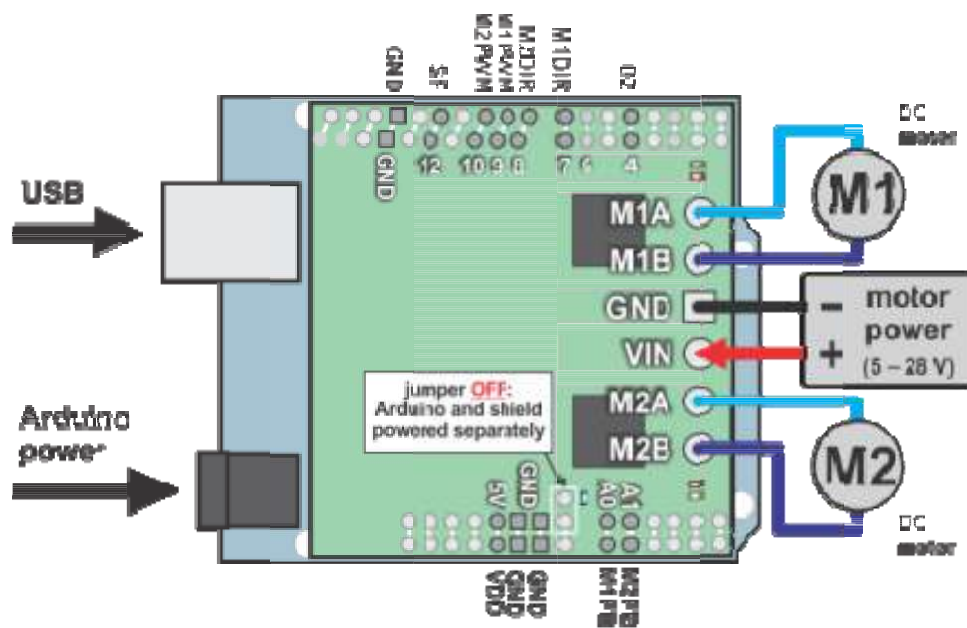


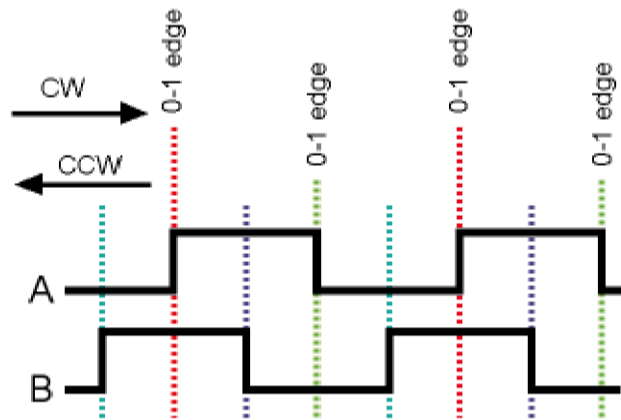
Figure 12 H-Bridge Arduino Shield



4. 霍爾編碼器



增量型的編碼器，處理兩相位霍爾感測器送出的脈波，解碼成四個相位 (Phase)，透過計數數值來計算旋轉的圈數。



5. 回授系統

在控制系統中，回授控制非常廣泛的使用在工業界中，其最基本的架構如下圖所示。

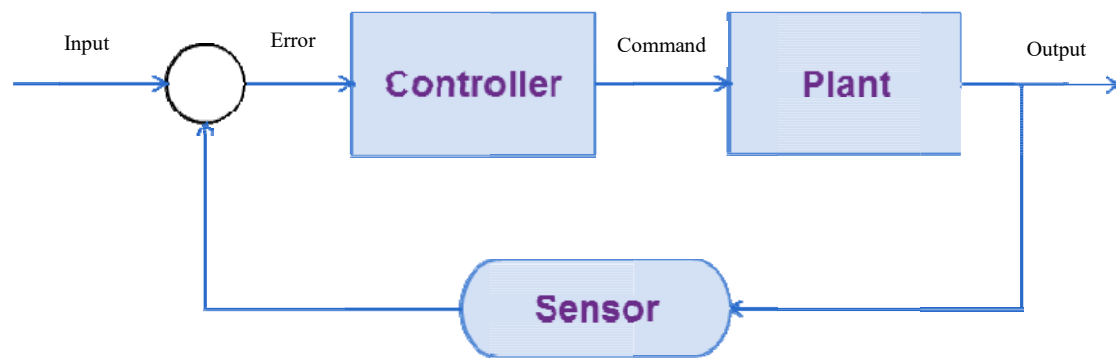


Figure 13 控制方塊圖

有了前述的元件以後，我們可以建構出一個簡單的馬達轉速回授的系統，並且利用 Arduino 微控制器，時做出 PID 控制器。

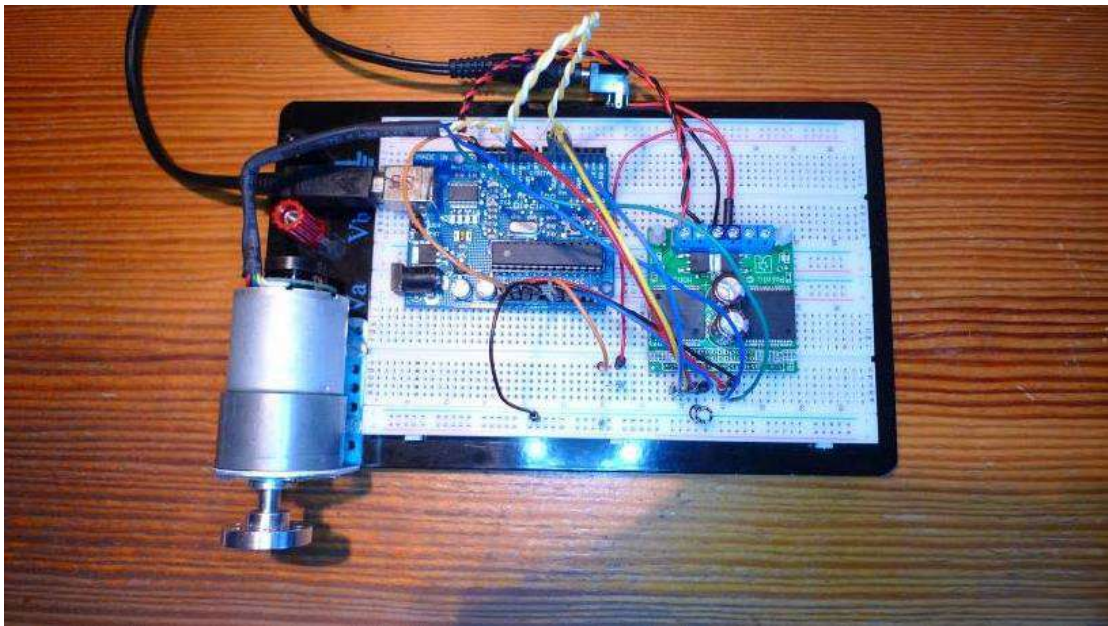


Figure 14 馬達轉速控制系統參考

簡單的系統方塊圖示意如下

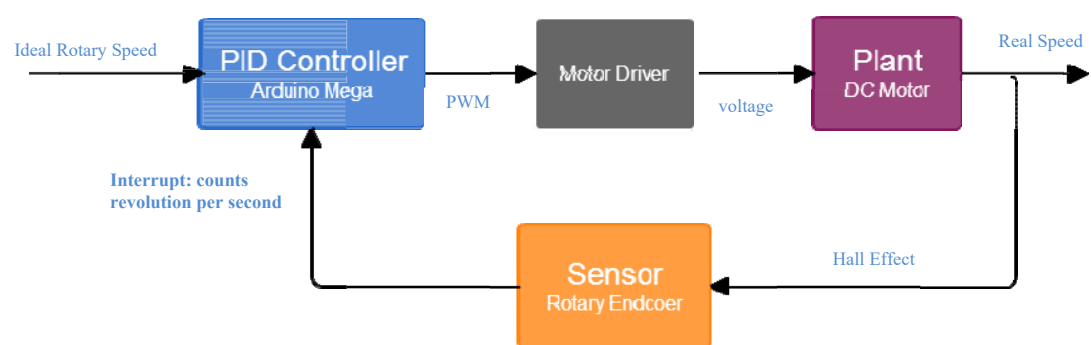


Figure 154 真實系統的方塊圖

8-4 實驗步驟

實驗一:跑馬燈控制

1. 將 MAX7219 顯示模組連接於 Arduino MEGA，並將 Arduino MEGA 連接於電腦(如 Figure 16 所示)，便可進行程式編譯。

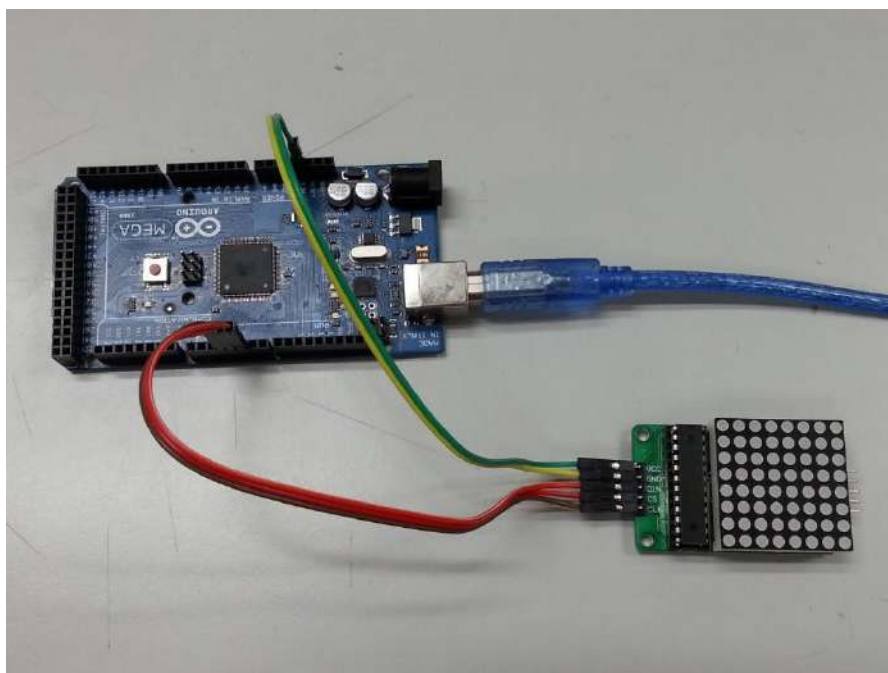


Figure 165

2. MAX7219 顯示模組連接於 Arduino MEGA 之詳細腳位說明。
(如 Table 5 及 Figure 17、18 所示)

Table 5 腳位說明

MAX7219 顯示模組之 Pin 腳	Arduino 腳位
VCC	5V
GND	GND
DIN	42
CS	40
CLK	38

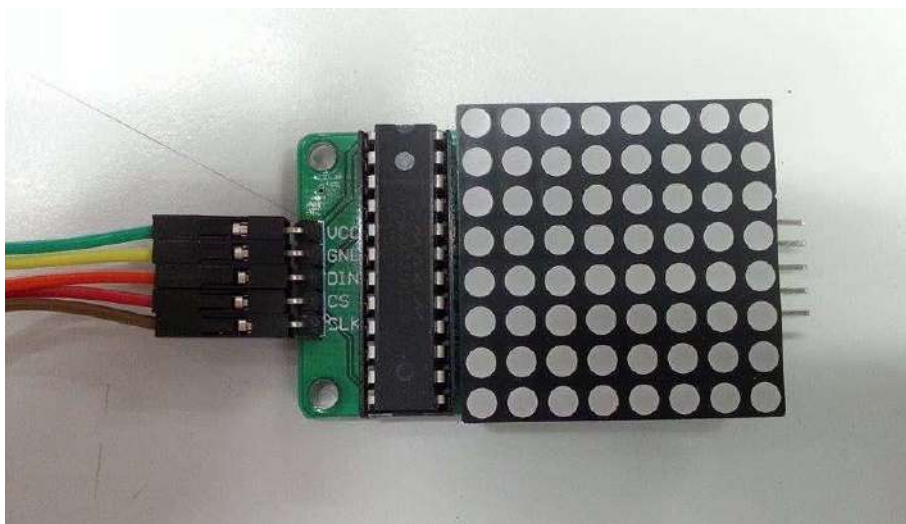


Figure 17 腳位說明

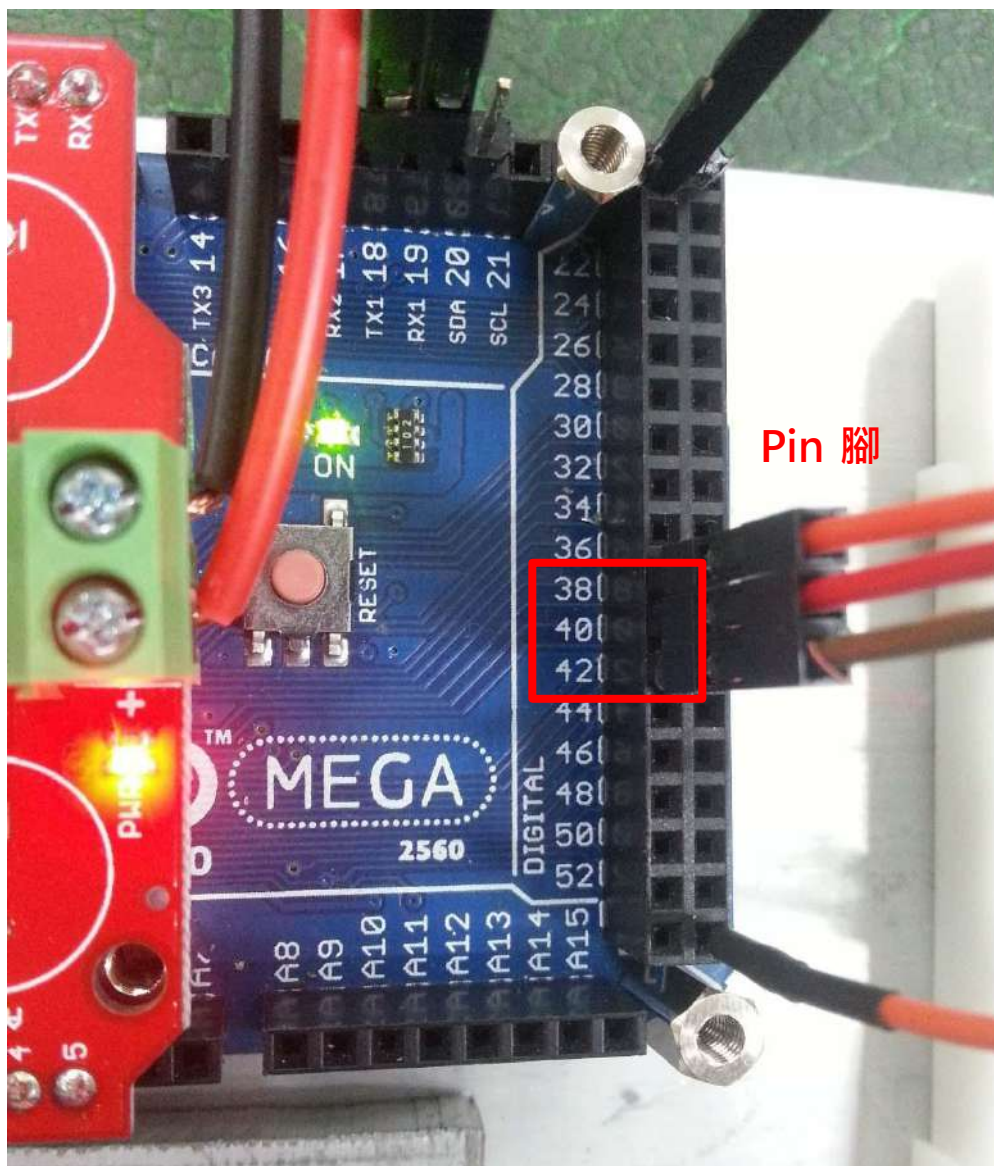


Figure 18 腳位說明

3. 開啟 Arduino IDE 軟體，並將範例程式 Upload 至 Arduino 上，即完成程式編譯。

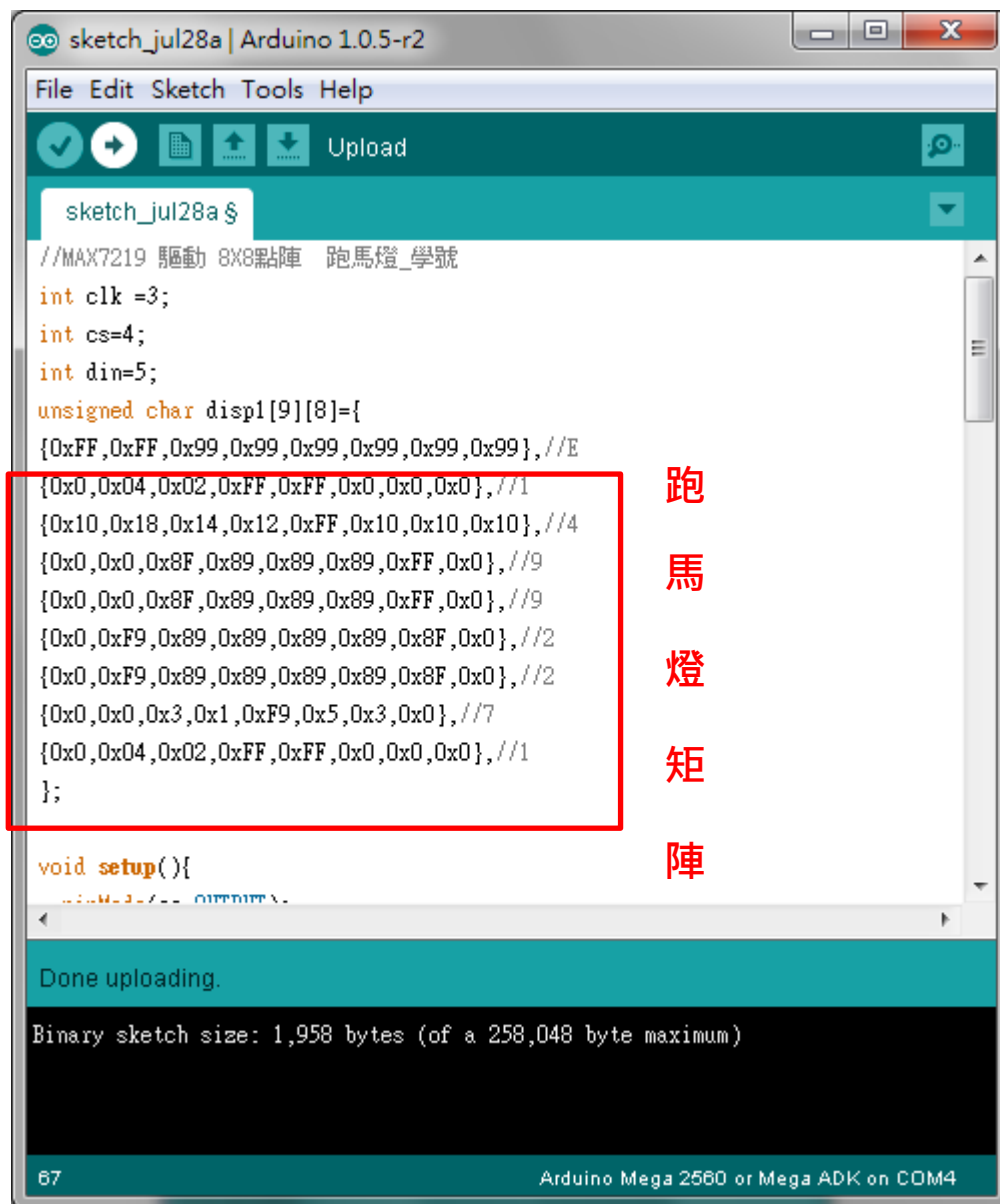


Figure 19 跑馬燈程式碼

4. 試編譯出組員學號之跑馬燈。

5. 範例程式，請上網下載

https://www.dropbox.com/s/8oz5ww7335o58gh/Newsticker_NCKUID.rar?dl=0

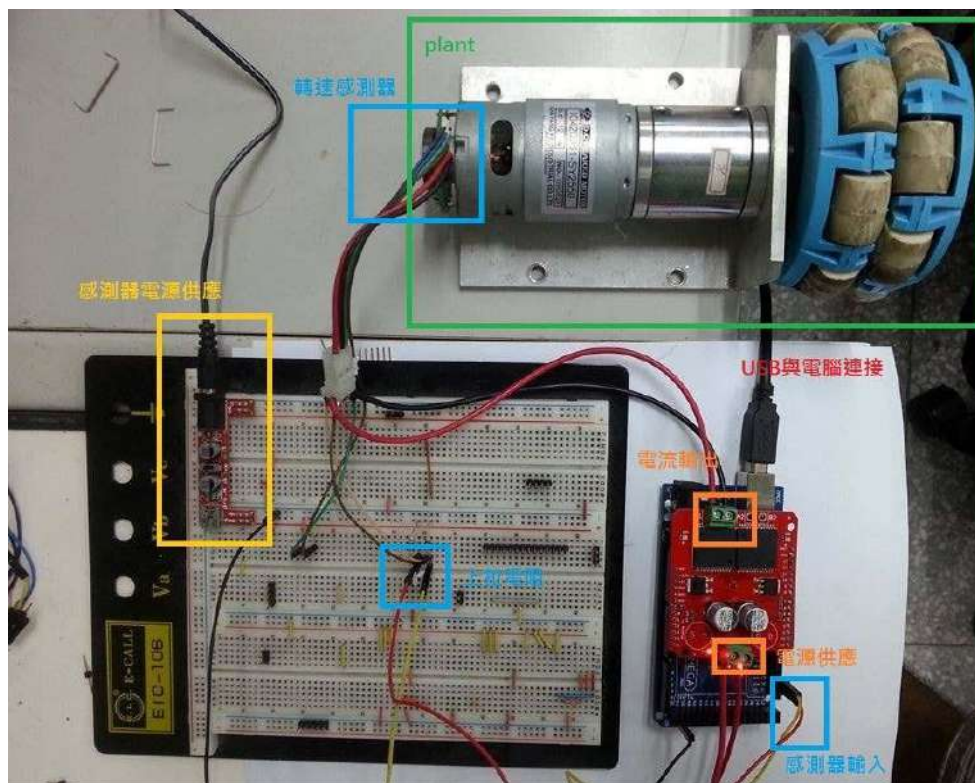
6. 參考資訊，如 Table 6 所示。

Table 6 參考資訊

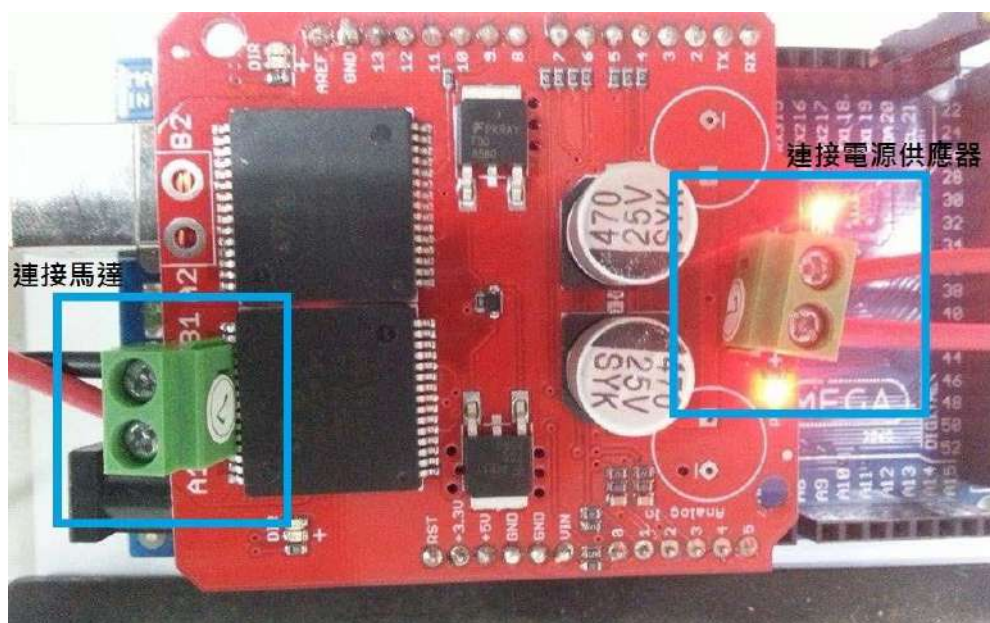
顯示	HEX CODE
E	{0x99,0x99,0x99,0x99,0x99,0x99,0xFF,0xFF}
1	{0x0,0x0,0x0,0xFF,0xFF,0x40,0x20,0x0}
2	{0x0,0xF1,0x91,0x91,0x91,0x91,0x9F,0x0}
3	{0xFF,0x89,0x89,0x89,0x89,0x89,0x89,0x89}
4	{0x8,0x8,0x8,0xFF,0x48,0x28,0x18,0x8}
5	{0x0,0x8F,0x89,0x89,0x89,0x89,0xF9,0x0}
6	{0x0,0x9F,0x91,0x91,0x91,0x91,0xFF,0x0}
7	{0x0,0xC0,0xE0,0x9F,0x9F,0x80,0xE0,0x0}
8	{0x0,0xFF,0x99,0x99,0x99,0x99,0xFF,0x0}
9	{0x0,0xFF,0x91,0x91,0x91,0x91,0xF1,0x0}
0	{0x0,0xFF,0x81,0x81,0x81,0x81,0xFF,0x0}

實驗二:PID 馬達轉速控制

1. 實驗平台架設完畢的示意圖如下

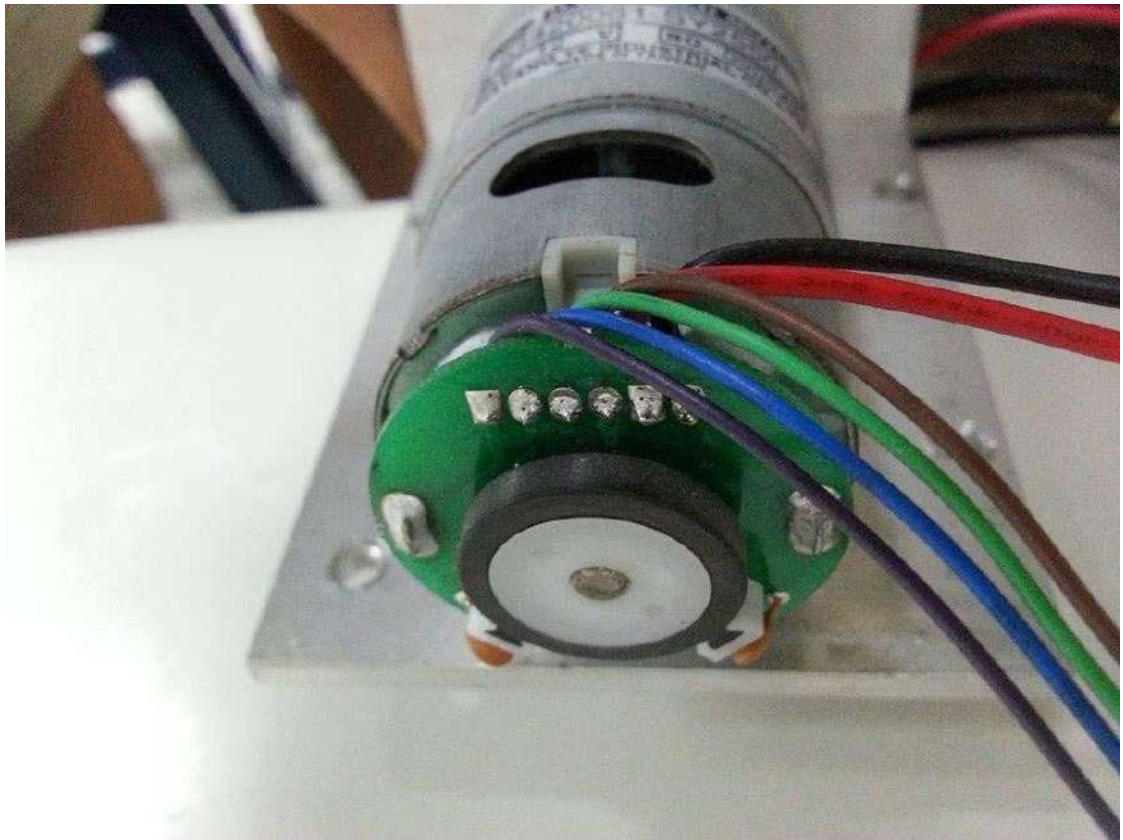


2. 將 Arduino 與 Motor Shield 扣合後，連接電腦並且接上電源



3. 連接霍爾轉速感測器

1) 感測器有四條線

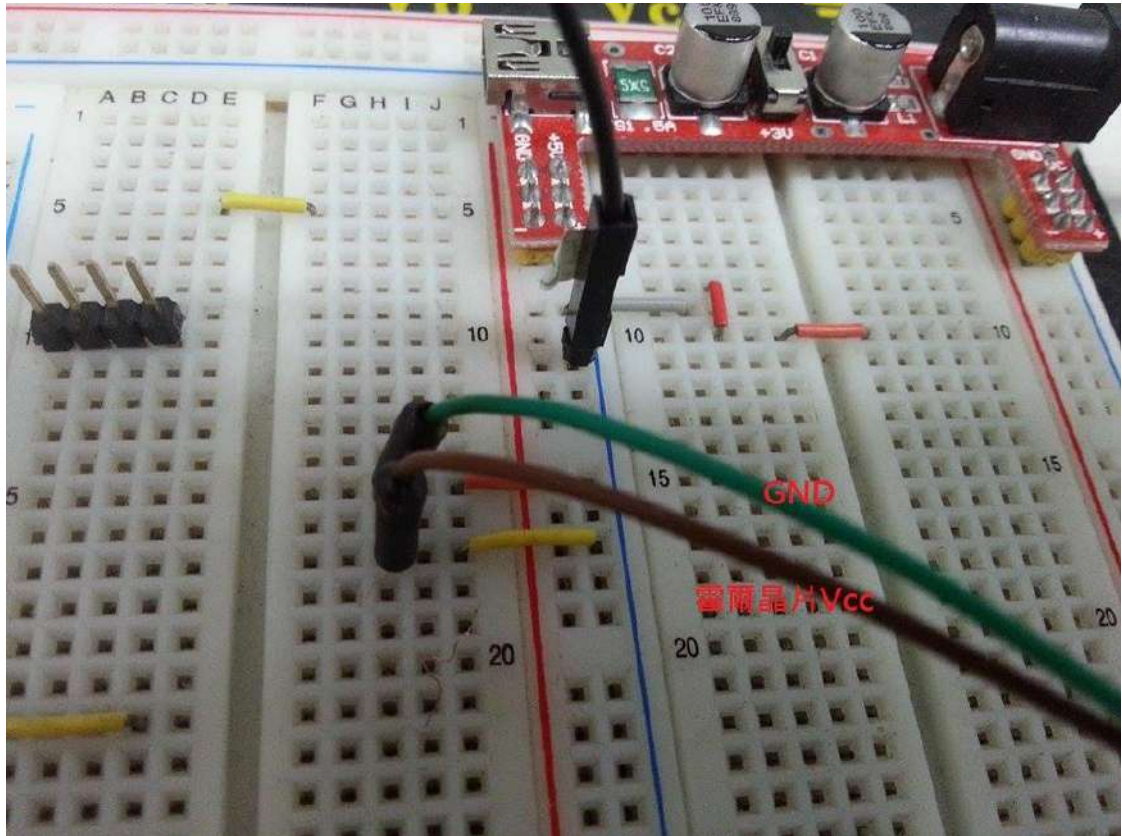


分別為下表所示

Two Channel Encoder Connections :

- | | |
|-----------|----------------------|
| 1. Black | : -MOTOR |
| 2. Red | : +MOTOR |
| 3. Brown | : HALL SENSOR Vcc |
| 4. Green | : HALL SENSOR GND |
| 5. Blue | : HALL SENSOR A Vout |
| 6. Purple | : HALL SENSOR B Vout |

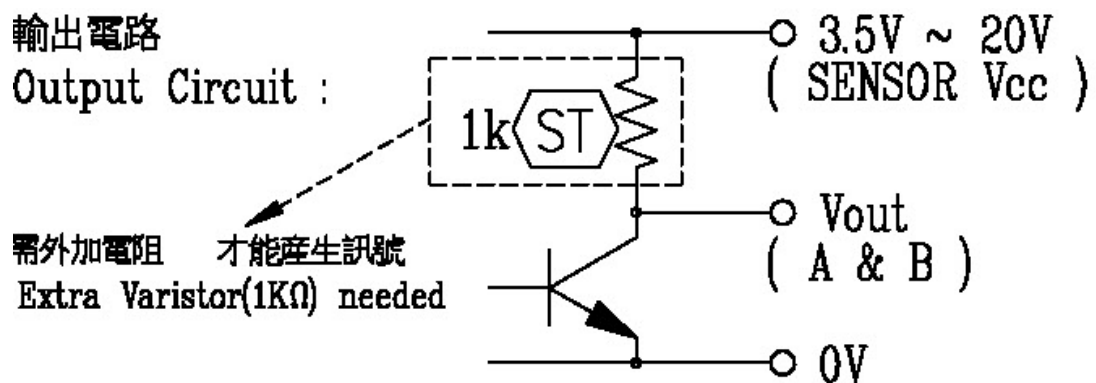
棕色與綠色線是作為霍爾晶片的工作電壓，需要外部供電(或是從 Arduino 的 5V 腳位輸出)



2) 另外兩條線為編碼後的訊號線。要讀取訊號前，需要先接 1kOhm 的上拉電阻，電路圖如下

輸出電路

Output Circuit :

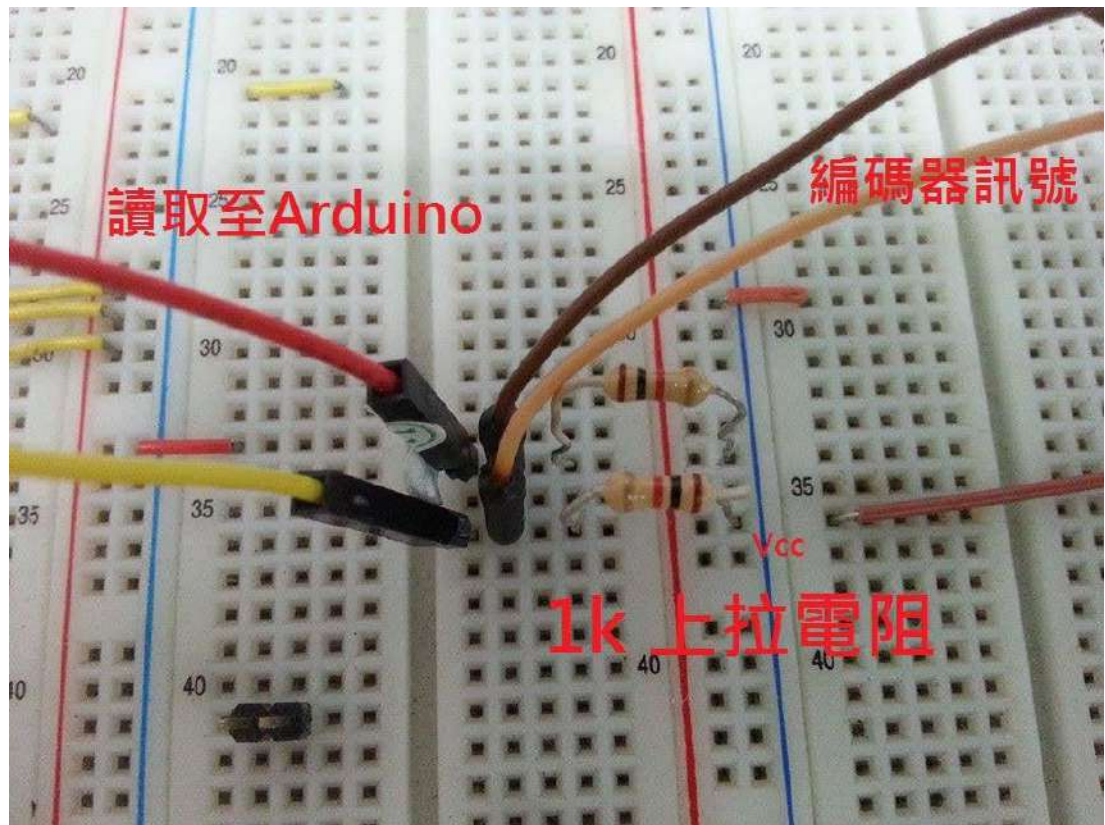


需外加電阻 才能產生訊號
Extra Varistor(1K Ω) needed

—○ Vout
(A & B)

—○ 0V

圖示說明訊號線要先接過電阻，然後將電阻連接到霍爾晶片的工作電壓，如此我們可以從 V_{out} 處量測訊號，實際電路連接如圖所示



3) 不要忘了Arduino 要與霍爾感測器晶片共用地線



4. 程式部分，請上網下載

PID Library:

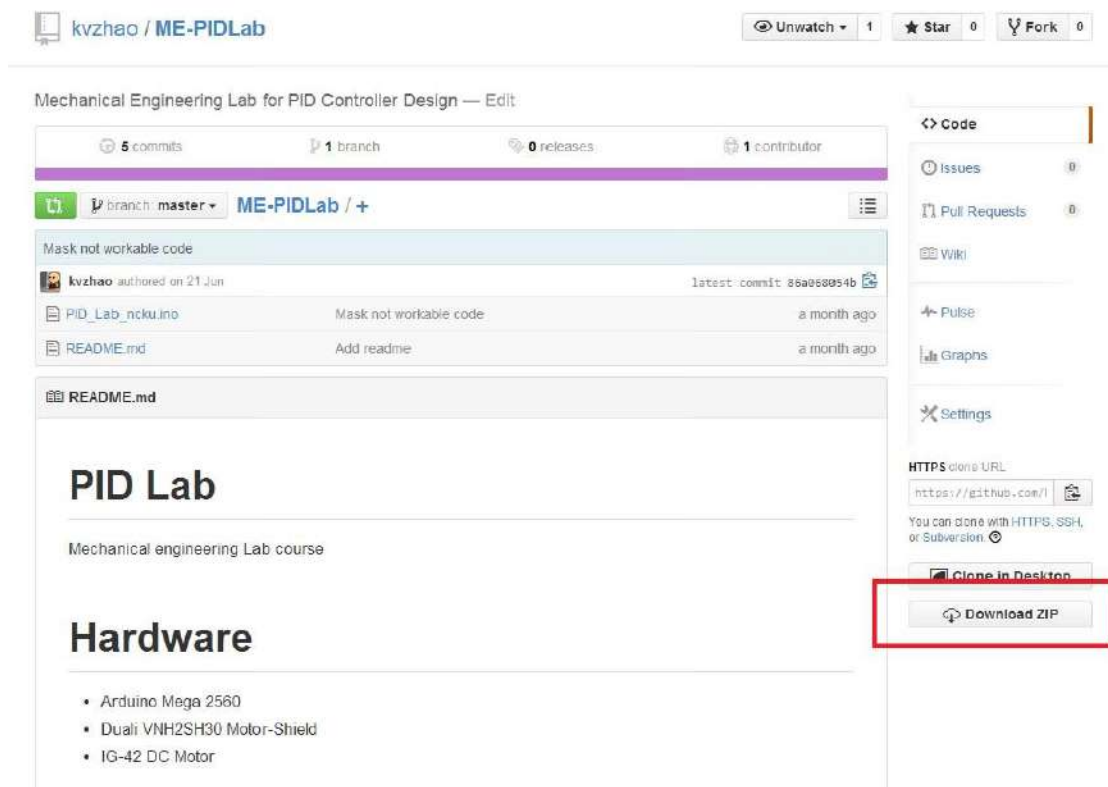
<https://github.com/br3ttb/Arduino-PID-Library>

馬達驅動程式：

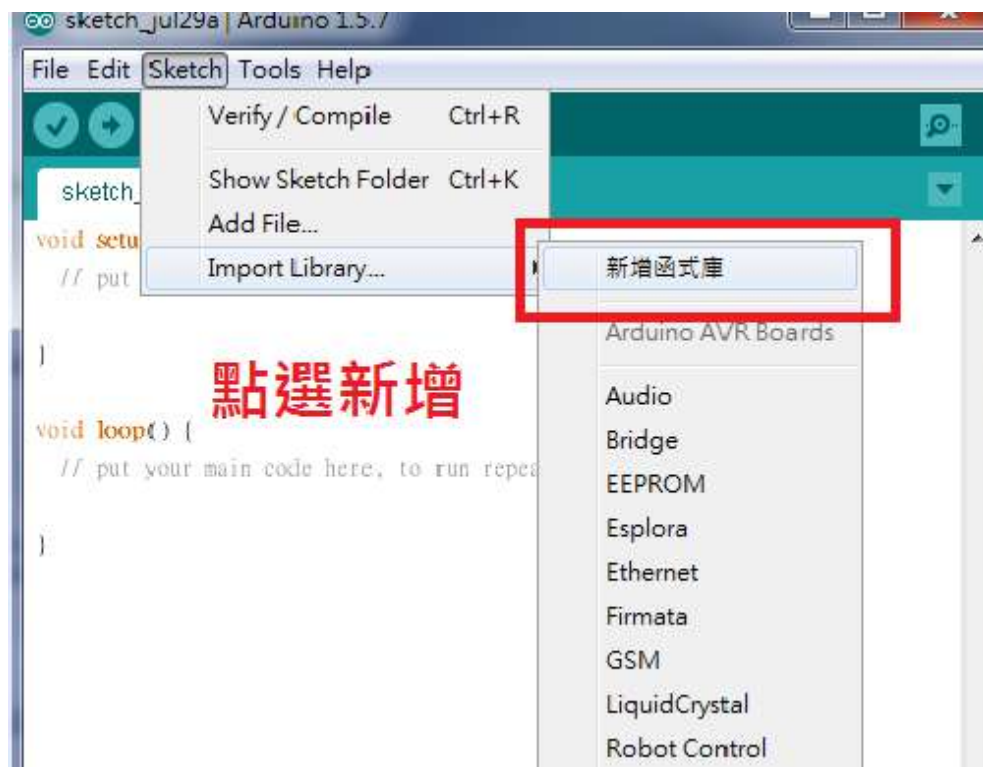
<https://github.com/kvzhao/DCMotorIG42>

本實驗程式：

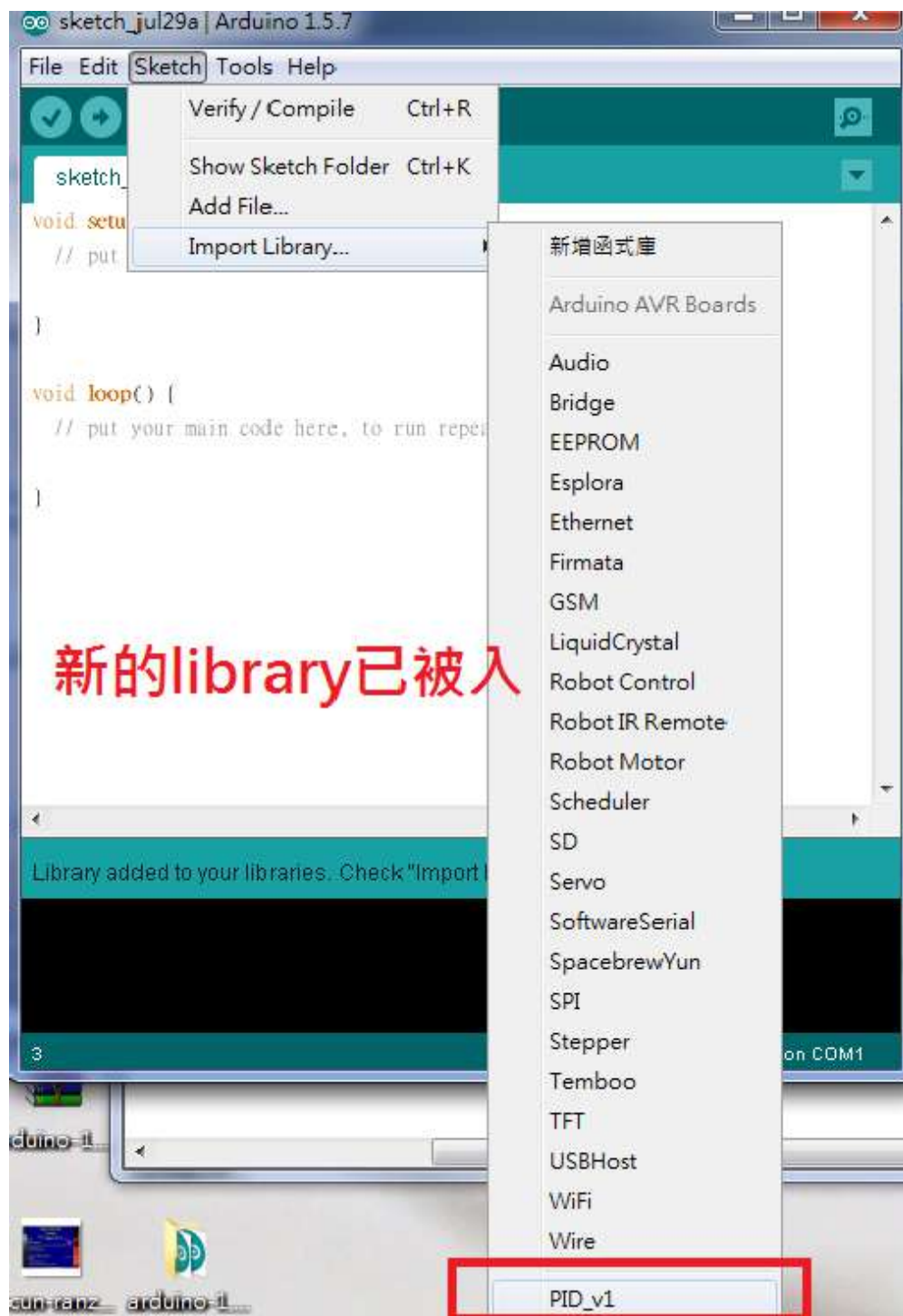
<https://github.com/kvzhao/ME-PIDLab>



5. 在 Arduino IDE 內加入函式庫



之後 library 就會出現在選單中



6.

```

sketch_aug07a $
* Parameters when we're farther away.
*****/

#include <DCMotorIG42.h>
#include <PID_v1.h>

// Controlling Target
double Setpoint, Input, Output;

// PID Parameters and Performance Index
int SamplingTime =50; // Execute
double Kp=100, Ki=20, Kd=5;
// If the error within the band, is Settling time
double errorBand = 0.02; // 2%
/*

```

呼叫函式庫↵

參數設定↵

```

sketch_aug07a $
/*
* -- Definition --
* Rise Time: time duration from 10% to 90% of final value
* Settling Time: time to approach the final value (within the error band)
* Time Constant: when output value approaches 63.2% of desired value
*/
double RiseTime, SettlingTime;
// Estimate PID
double error;
// Prevent Quick Oscillation
double prevOutput=0, oscGap = 100;

// Pins of Encoder
const uint8_t encoderPinA = 18, encoderPinB =19;

// Specify the links and initial tuning parameters
PID PID_Controller(&Input, &Output, &Setpoint, Kp, Ki, Kd, DIRECT);
// Init the DC Motor
DCMotorIG42 Motor(encoderPinA, encoderPinB, 1);

```

設定 PIN 位置↵

```

sketch_aug18a PID_Lab_ncku $

void setup()
{
  // initialize serial communication
  Serial.begin(9600);
  Serial.println("system initializing...");
  //initialize the variables we're linked to
  Input = 0;
  Output = 0;
  Setpoint = 1.5;
  Serial.println("set PID mode");
  //turn the PID on
  PID_Controller.SetMode(AUTOMATIC);
  PID_Controller.SetOutputLimits(0,400);
  PID_Controller.SetSampleTime(SamplingTime);
  Serial.println("motor driver init done");
  /* The Debug Section */
  /* Serial.println("Motor Running Test");
  Motor.RunTest();
  Motor.stop();*/
}
  
```

主程式 Setup

```

sketch_aug18a PID_Lab_ncku $

void loop()
{
  boolean settingDone = false;
  /* Head Section: Command Parse and Parameters Setting */
  Serial.print("PID parameters: Kp = "); Serial.print(Kp);
  Serial.print(", Ki = "); Serial.print(Ki);
  Serial.print(", Kd = "); Serial.print(Kd);
  Serial.println(")");
  Serial.println("Please Enter which you modify");
  /*
  while (!settingDone) {
    if(CheckSerial()) {
      if (strcmp(commandBuffer, "kp") == 0) {
        Serial.print("Kp = ");
        double input = Serial.read();
        PID_Controller.SetTunings(input, Ki, Kd);
      }
    }
  }
  */
  /* Second part: Start Running */
}
  
```

主程式 Loop

8-5 實驗記錄與觀察

實驗一:跑馬燈控制

1. 試自行設計出一組跑馬燈

實驗二:PID 馬達轉速控制

1. 以自行設計的 PID 參數 K_p 、 K_i 、 K_d 做實驗，試調出 Settling time 的最小值，並利用 Matlab 畫出 Error 與時間之曲線關係圖。

9-6 預習報告

1. 試畫出 PID 控制器之方塊圖
2. 試敘述馬達控制系統 PID 參數 K_p 、 K_i 、 K_d 之意義與功能
3. 試自行設計一組PID 參數 K_p 、 K_i 、 K_d ，可讓Settling time為最小值，請從建議參數附近找尋 PID 參數。（建議參數: $K_p = 100$ 、 $K_i = 20$ 、 $K_d = 5$ ）

8-7 問題與討論

1. 由此實驗二中可觀察到三個參數 K_p 、 K_i 、 K_d 對各項性能指標之影響，並完成下列表格。(假設 K_p 、 K_i 、 K_d 三項參數皆是增加趨勢)

Parameter	Rise time	Overshoot	Settling time	Steady-state error	Stability
$K_p \uparrow$					
$K_i \uparrow$					
$K_d \uparrow$					

2. 心得與感想

8-8 References

1. motor driver

<http://goods.ruten.com.tw/item/show?21211284593244>

<http://www.pololu.com/docs/0J55/all>

2. H-bridge

<http://www.robotroom.com/BipolarHBridge.html>

http://eshare.stust.edu.tw/EshareFile/2010_4/2010_4_ccd9befb.pdf

<http://prototyperobotics.com/tutorials/the-h-bridge>

3. dc motor

http://www.playrobot.com/cart/shop.php?id=307&factory=&header=&sub=&ctype2=&typeid=41&pagename=&Fno=&date_buy=

4. pwm

<http://arduino.cc/en/Tutorial/PWM>

<http://www.ermicro.com/blog/?p=1908>

http://www.bristolwatch.com/arduino/arduino_pwm_hb.htm

5. pid control

<http://forum.arduino.cc/index.php/topic,8652.0.html>

6. arduino

<http://www.cs.pu.edu.tw/~yctsao/mybook/book6/bk6.pdf>

<http://icerc.tnssh.tn.edu.tw/download/epaper/epaper43/20091130.pdf>

<http://blog.kenyang.net/2012/07/arduino.html>

附錄

實驗 BOM

元件名稱	規格	編號	數量
Arduino		MEGA	1
D.C GEARED MOTOR	IG-42GM	MOD0007	1
顯示模組		MAX7219	1
H-Bridge 模組	Monster Moto Shield	DEV-10182	1
杜邦線			數條
電阻	1 K Ω		2
麵包板			1