嵌入式系統總整與實作

曾煜棋、吳昆儒

National Yang Ming Chiao Tung University

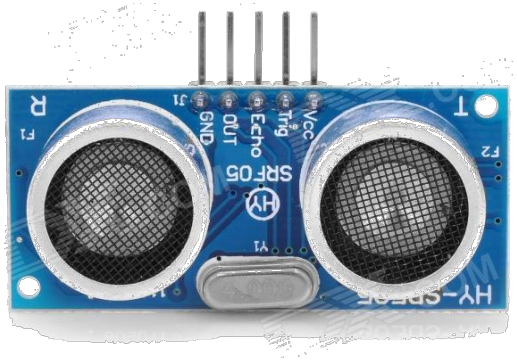
0. 課程大綱時程

本周所需零件

- 麵包版、LED燈泡、電阻 (至少拿3根)
- 杜邦線 (一人先拿20根, 未來比較方便)
- 超音波感測器
- 慣性感測器 10 DOF IMU Sensor (D)

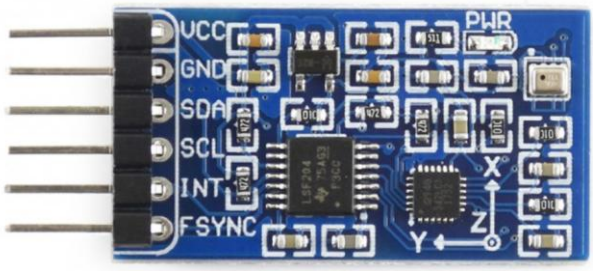
LED

HC-SR04



Ultrasonic

ICM20948



IMU
(Inertial measurement unit)

Last week

- 1. PI的環境設定
- 2. 設定遠端桌面連線, 開啟 “direct capture mode”
- 3. 介紹GPIO + Python
- 4. 傳輸檔案到PI

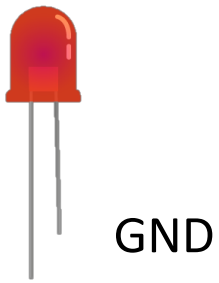
This week

- How to use wire to connect devices?
- How to write code to read sensing data on GPIO?

Outline

- 1. GPIO intro, connect LED and wires
- 2. Ultrasonic sensor
- 3. IMU (10 DOF: ICM20948 + BMP280)

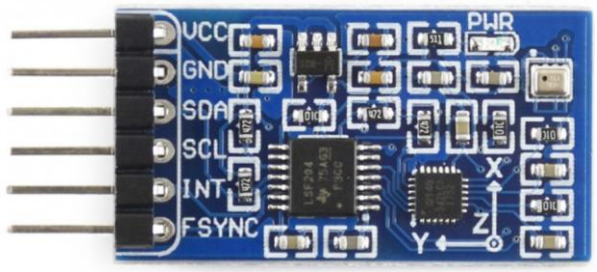
LED



HC-SR04

Ultrasonic

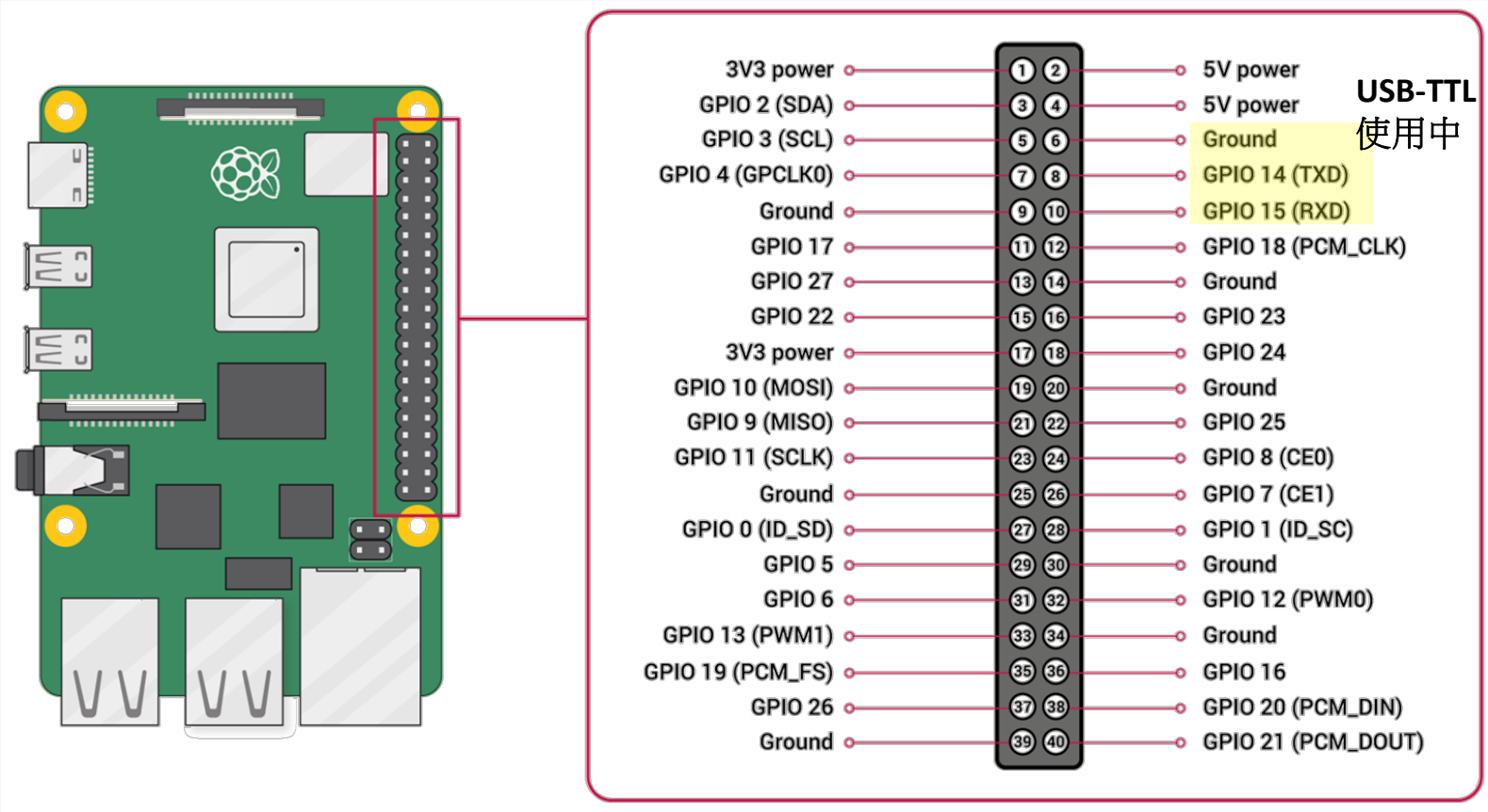
ICM20948



IMU
(Inertial measurement unit)

GPIO information

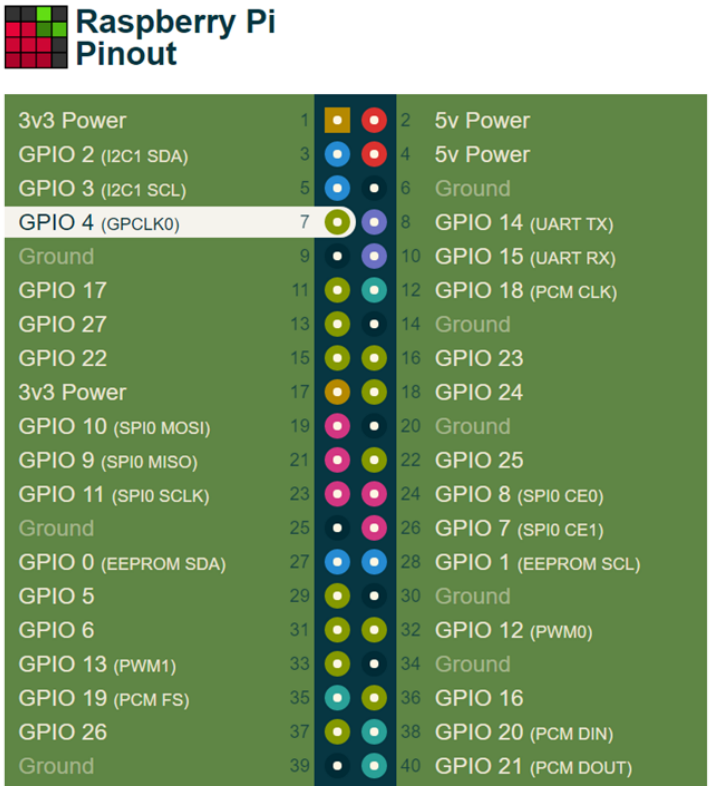
- The PIN (Physical) numbering is in Z-shape

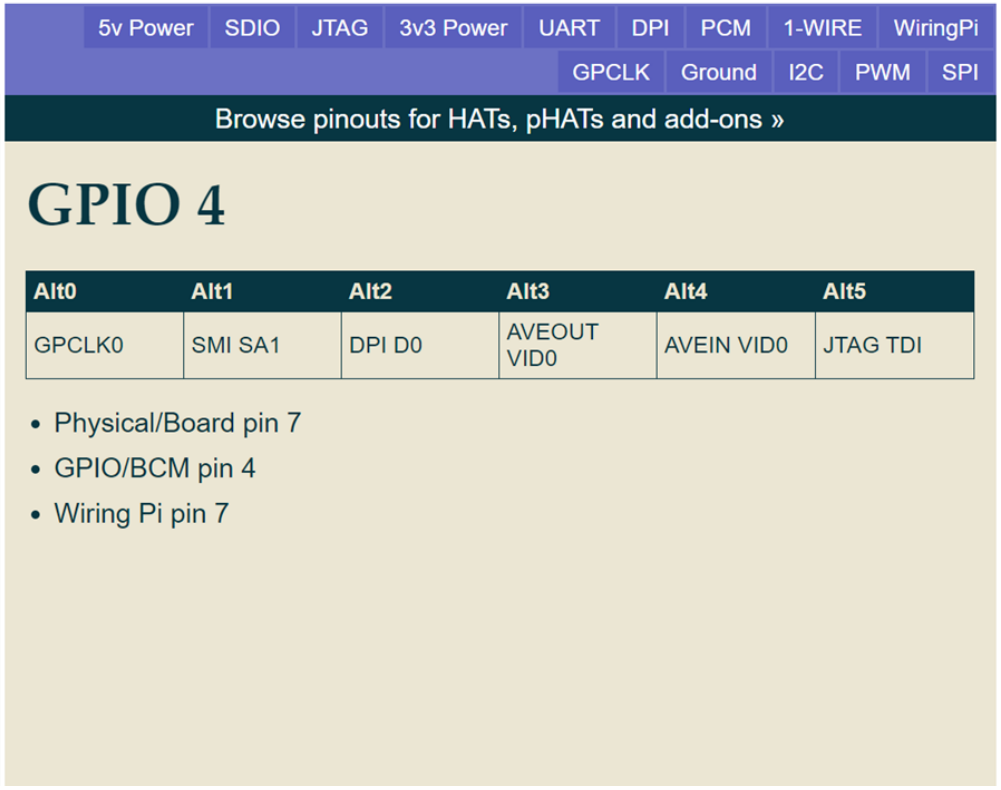


https://www.raspberrypi.org/documentation/usage/gpio/

GPIO information

- Pin number != GPIO number
  - Physical numbering vs. GPIO numbering





https://pinout.xyz/pinout/pin7_gpio4

PI4 datasheet

- Power Requirements
  - USB-C power supply: 5V at 3A
  - If attached devices consume less than 500mA, a 5V, 2.5A supply may be used.

https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf

- GPIO Limitations
  - Do not poke at the GPIO connector with a screwdriver or any metal object when the PI is powered up.
  - Do not power the PI with more than 5V.
  - Do not put more than 3.3V on any GPIO pin being used as an input.

From: Raspberry Pi Cookbook: Software and Hardware Problems and Solutions
https://books.google.com.tw/books?id=0skvDAAAQBAJ&pg=PT270&lpg=PT270#v=onepage&q&f=false

PI4 datasheet

- 4 Electrical Specification

Caution! Stresses above those listed in Table 2 may cause permanent damage to the device. This is a stress rating only; functional operation of the device under these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Symbol	Parameter	Minimum	Maximum	Unit
VIN	5V Input Voltage	-0.5	6.0	V

Table 2: Absolute Maximum Ratings

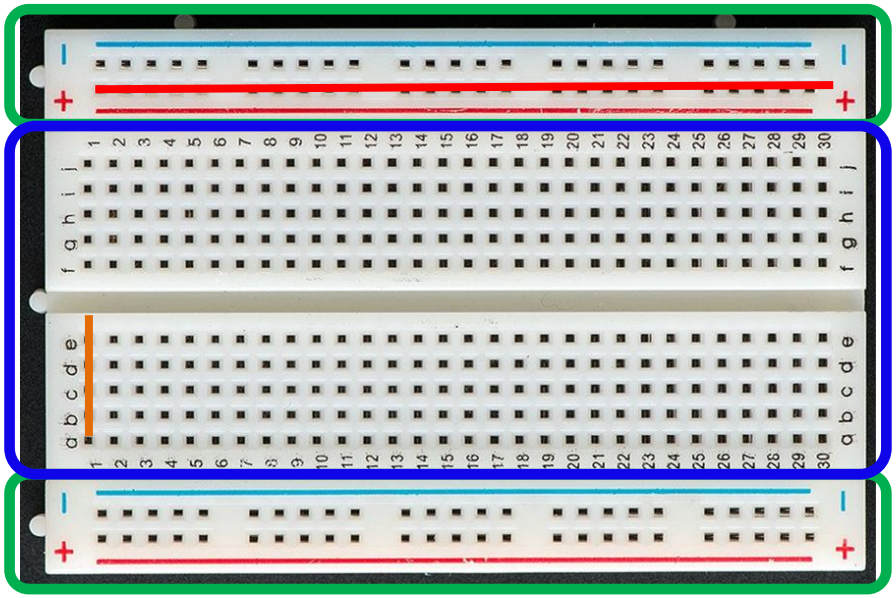
- 5.6 Temperature Range and Thermals

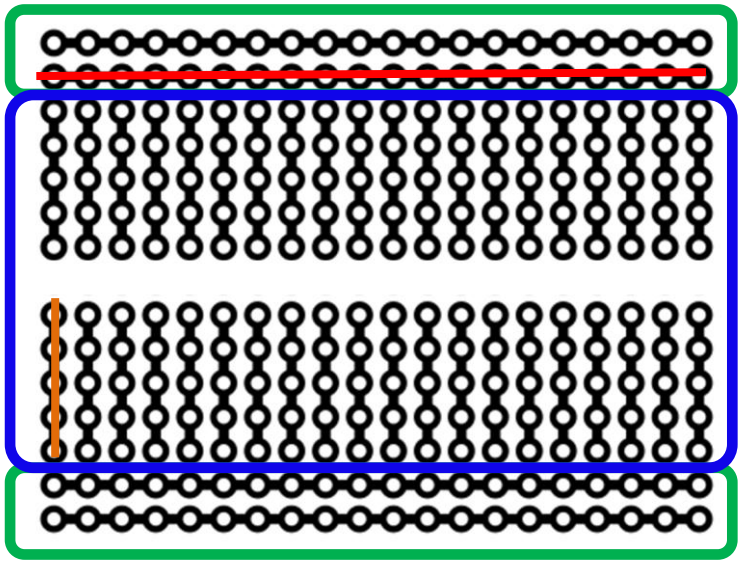
The Pi4B will operate perfectly well without any extra cooling and is designed for sprint performance - expecting a light use case on average and ramping up the CPU speed when needed (e.g. when loading a webpage). If a user wishes to load the system continually or operate it at a high temperature at full performance, further cooling may be needed.

https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf

Bread Board

Bus strips: one for ground (-) and one for a supply voltage (+)



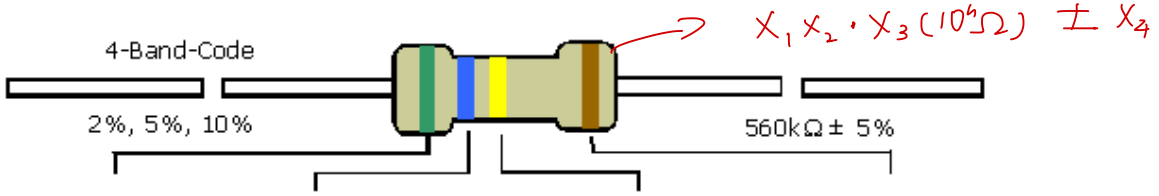


Terminal strips: (ex: a1, b1, c1, d1 and e1 are connected)

麵包版線上模擬 (需註冊)
https://www.tinkercad.com/learn/overview/OPRHCXXL20FZS3N

Resistor

use color table to determine value



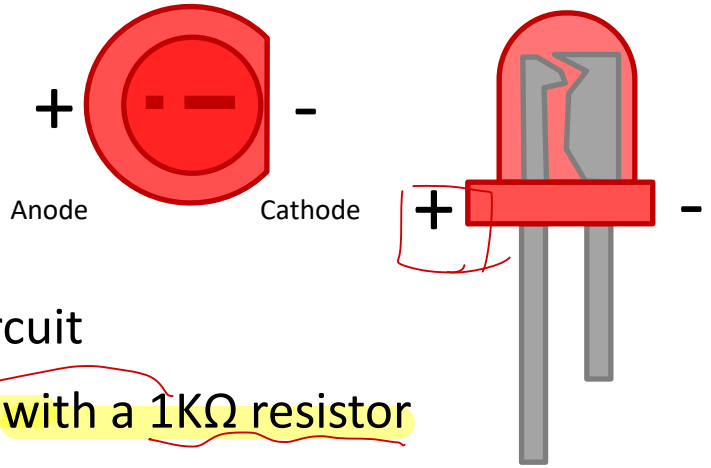
COLOR	1st BAND	2nd BAND	3rd BAND	MULTIPLIER	TOLERANCE	
Black	0	0	0	1Ω		
Brown	1	1	1	10Ω	± 1%	(F)
Red	2	2	2	100Ω	± 2%	(G)
Orange	3	3	3	1KΩ		
Yellow	4	4	4	10KΩ		
Green	5	5	5	100KΩ	±0.5%	(D)
Blue	6	6	6	1MΩ	±0.25%	(C)
Violet	7	7	7	10MΩ	±0.10%	(B)
Grey	8	8	8		±0.05%	
White	9	9	9			
Gold				0.1	± 5%	(J)
Silver				0.01	± 10%	(K)

Ex: 20KΩ = Red, Black, Orange = 20*1000 (4-band)

= Red, Black, Black, Red = 200*100 (5-band)

LED

- The LED (Light Emitting Diode) is a simple, digital actuator



- LEDs have a short leg (-) and a long leg (+) and it matters how they are oriented in a circuit
- To prevent damage, LEDs are used together with a 1KΩ resistor (or anything from 300Ω to 2KΩ)

Electrical / Optical Characteristics at TA=25°C

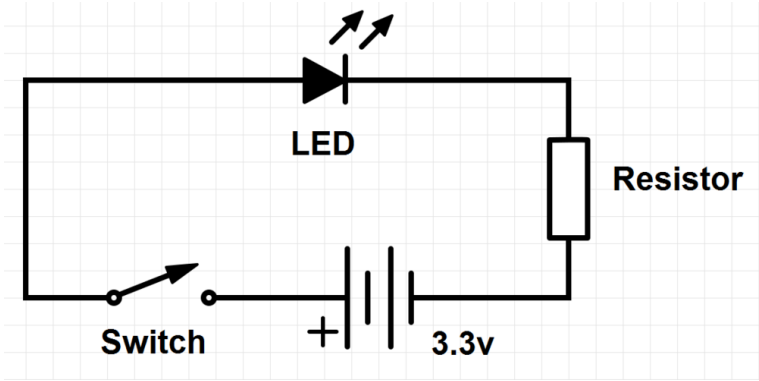
Symbol	Parameter	Device	Typ.	Max.	Units	Test Conditions
λpeak	Peak Wavelength	Super Bright Red	660		nm	IF=20mA
λD [1]	Dominant Wavelength	Super Bright Red	640		nm	IF=20mA
Δλ1/2	Spectral Line Half-width	Super Bright Red	20		nm	IF=20mA
C	Capacitance	Super Bright Red	45		pF	VF=0V;f=1MHz
VF [2]	Forward Voltage	Super Bright Red	1.85	2.5	V	IF=20mA
IR	Reverse Current	Super Bright Red		10	uA	VR = 5V

Notes:
1.Wavelength: +/-1nm.
2. Forward Voltage: +/-0.1V.

LED

$\frac{(3.3-1.85)}{0.02} = 72.5\,\Omega$

- 根據前面的參數
  - PI的GPIO腳位可以提供3.3V
  - LED的順向電壓是1.85V
  - LED的電流需要20mA



- 電阻公式: R=V/I
  - R =(3.3-1.85)/0.02=72.5 歐姆
  - 最少須接 72.5 歐姆的電阻 , 才可避免 LED燒毀

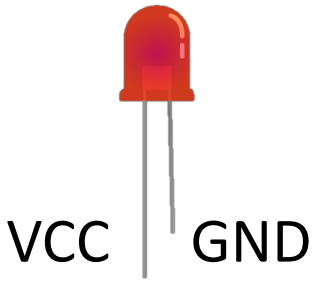
https://www.raspberrypi.org/documentation/usage/gpio/

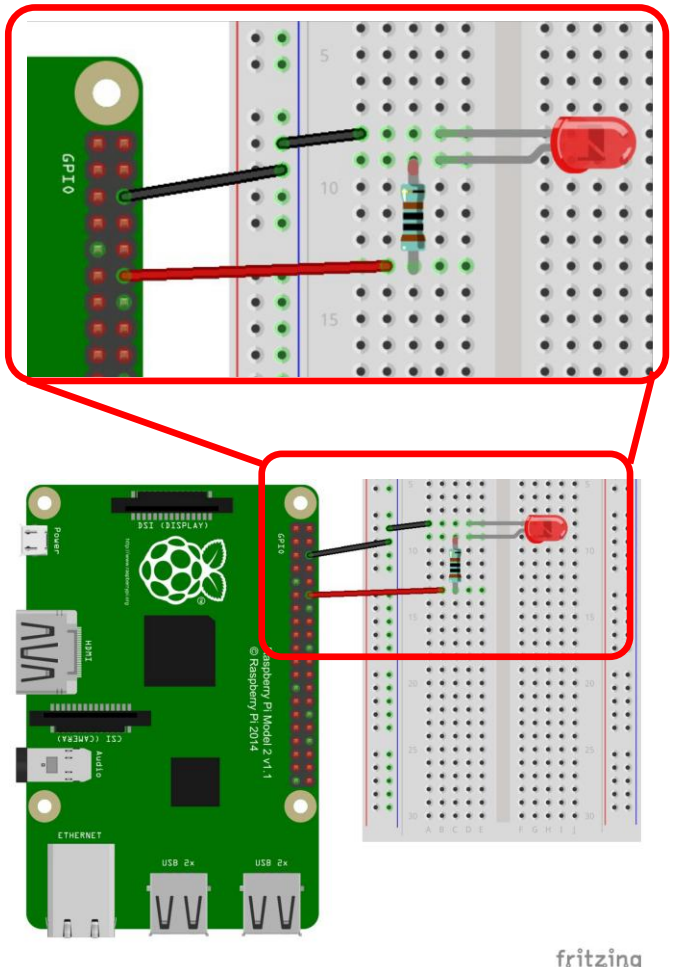
Discussion 1

- Identify the resistor from this class. (Ω)
  - Write down the color which you see
  - How to calculate the resistor value?

Control LED

- Goal: create a simplest thing to control GPIO and see physical output
- Output: it blinks an LED.
- Hardware Required
  - Raspberry PI
  - LED





Control LED by Python

```
import RPi.GPIO as GPIO
import time
```

Load GPIO library

```
LED_PIN = 12
GPIO.setmode(GPIO.BOARD)
GPIO.setup(LED_PIN, GPIO.OUT)
```

LED is on pin 12 by pin numbering (z-shape)

```
try:
 while True:
  print("LED is on")
  GPIO.output(LED_PIN, GPIO.HIGH)
  time.sleep(1)
  print("LED is off")
  GPIO.output(LED_PIN, GPIO.LOW)
  time.sleep(1)
```

The **try** clause (the statement(s) between the try and except keywords) is executed.

```
except KeyboardInterrupt:
 print "Exception: KeyboardInterrupt"
```

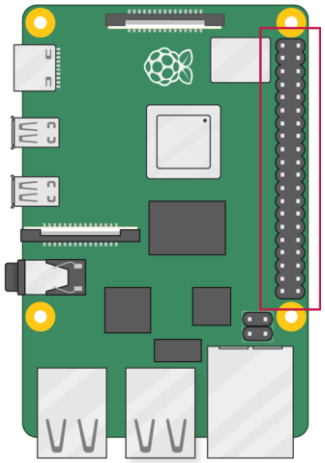
A user-generated **interruption** is signaled (ctrl + c)

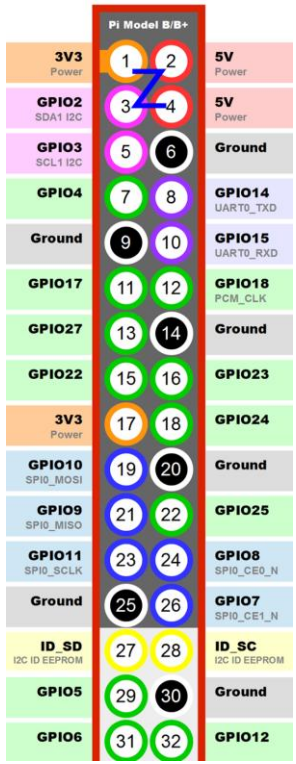
```
finally:
 GPIO.output(LED_PIN, GPIO.LOW)
 GPIO.cleanup()
```

A **finally** clause is always executed before leaving the try statement, whether an exception has occurred or not.

Syntax - GPIO

- GPIO.setmode(GPIO.BOARD)
  - GPIO.BOARD: Define by the number of the pin plug (z-shape)
  - GPIO.BCM: Define by the "Broadcom SOC channel" number





- GPIO.setup(LED_PIN, GPIO.OUT)
  - Set LED_PIN to output mode

- GPIO.cleanup()
  - clean up all the ports you've used

[Sad story] *when you have a port set HIGH as an output and you accidentally connect it to GND (LOW), which would short-circuit the port and possibly fry it.*

http://raspi.tv/2013/rpi-gpio-basics-3-how-to-exit-gpio-programs-cleanly-avoid-warnings-and-protect-your-pi

Syntax – Python error statement

- The **try** statement works as follows.
  - First, the **try** clause (the statement(s) between the try and except keywords) is executed.
  - If no exception occurs, the **except** clause is skipped and execution of the try statement is finished.
  - If an **exception** occurs during execution of the try clause, the rest of the clause is skipped. Then if its type matches the exception named after the except keyword, the except clause is executed, and then execution continues after the try statement.
  - If a **finally** clause is present, the finally clause will execute as the last task before the try statement completes. The finally clause runs whether or not the try statement produces an exception.

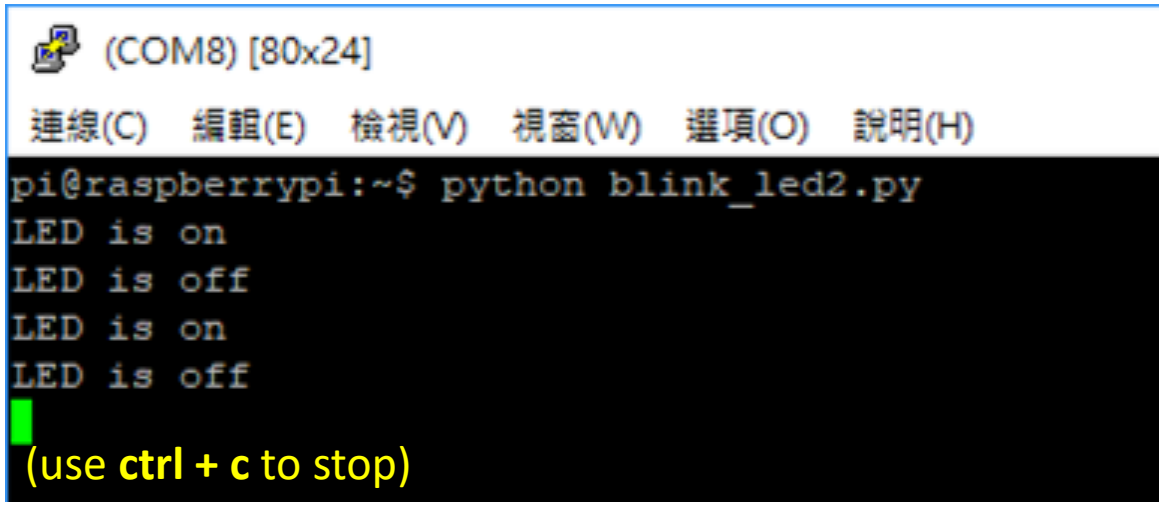
```
try:
    while True:
        print("LED is on")
        GPIO.output(LED_PIN, GPIO.HIGH)
        time.sleep(1)
        print("LED is off")
        GPIO.output(LED_PIN, GPIO.LOW)
        time.sleep(1)

except KeyboardInterrupt:
    print "Exception: KeyboardInterrupt"

finally:
    GPIO.output(LED_PIN, GPIO.LOW)
    GPIO.cleanup()
```

https://docs.python.org/3/tutorial/errors.html

How to write Python code on PI?

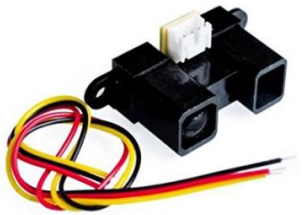
- Use text editor on PI (ex: nano, vim … etc)
  - Ex: **nano blink.py**
- Write code, then save it
  - In nano, press **ctrl + x** to exit
- Execution (use ctrl + c to stop) // 因為有控制GPIO狀態, 要加上sudo
  - **sudo python blink.py**

- Write code on PC, then send to PI
  - Ex: Using scp, mobaxterm…

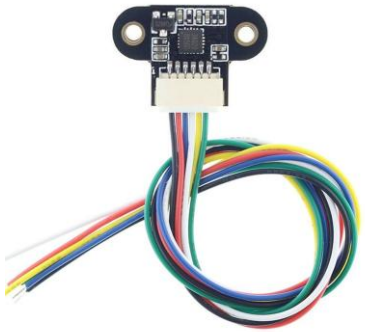


可以測量距離的sensor

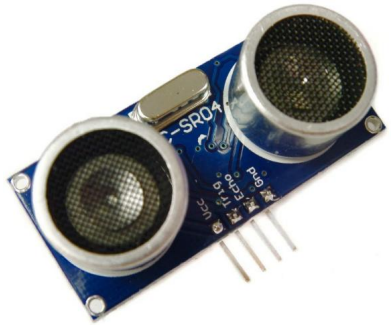
- SHARP GP2Y0A02YK0F 紅外線距離感測器 (測量範圍 20~150cm)



- TOF10120 雷射測距 距離感測器



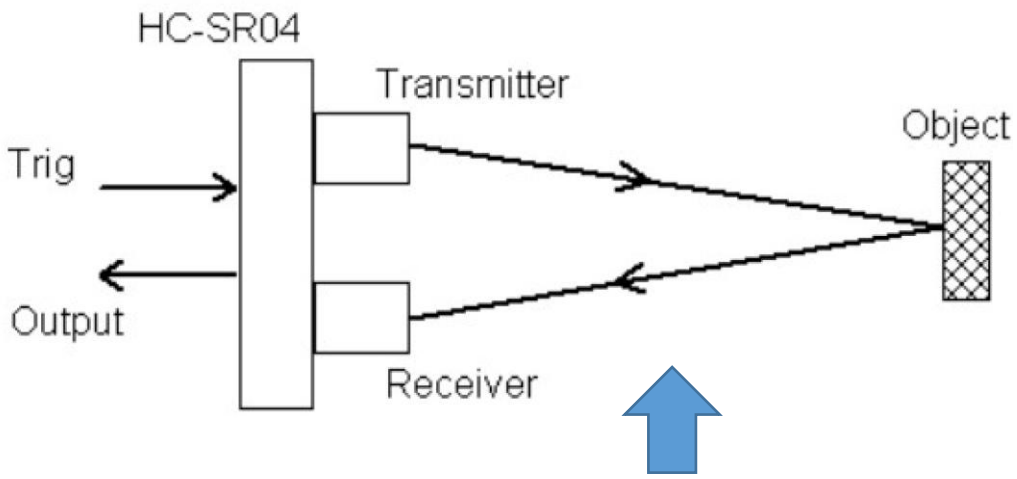
- HC-SR04 超音波測距 / 避障模組



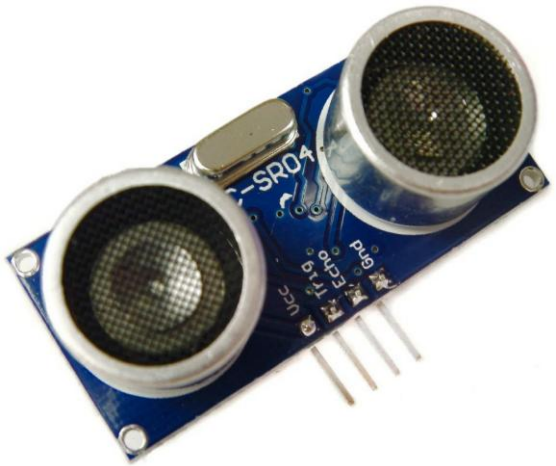
- BOSCH 30米雷射測距儀
- (其他...)

2. Ultrasonic (HC-SR04)

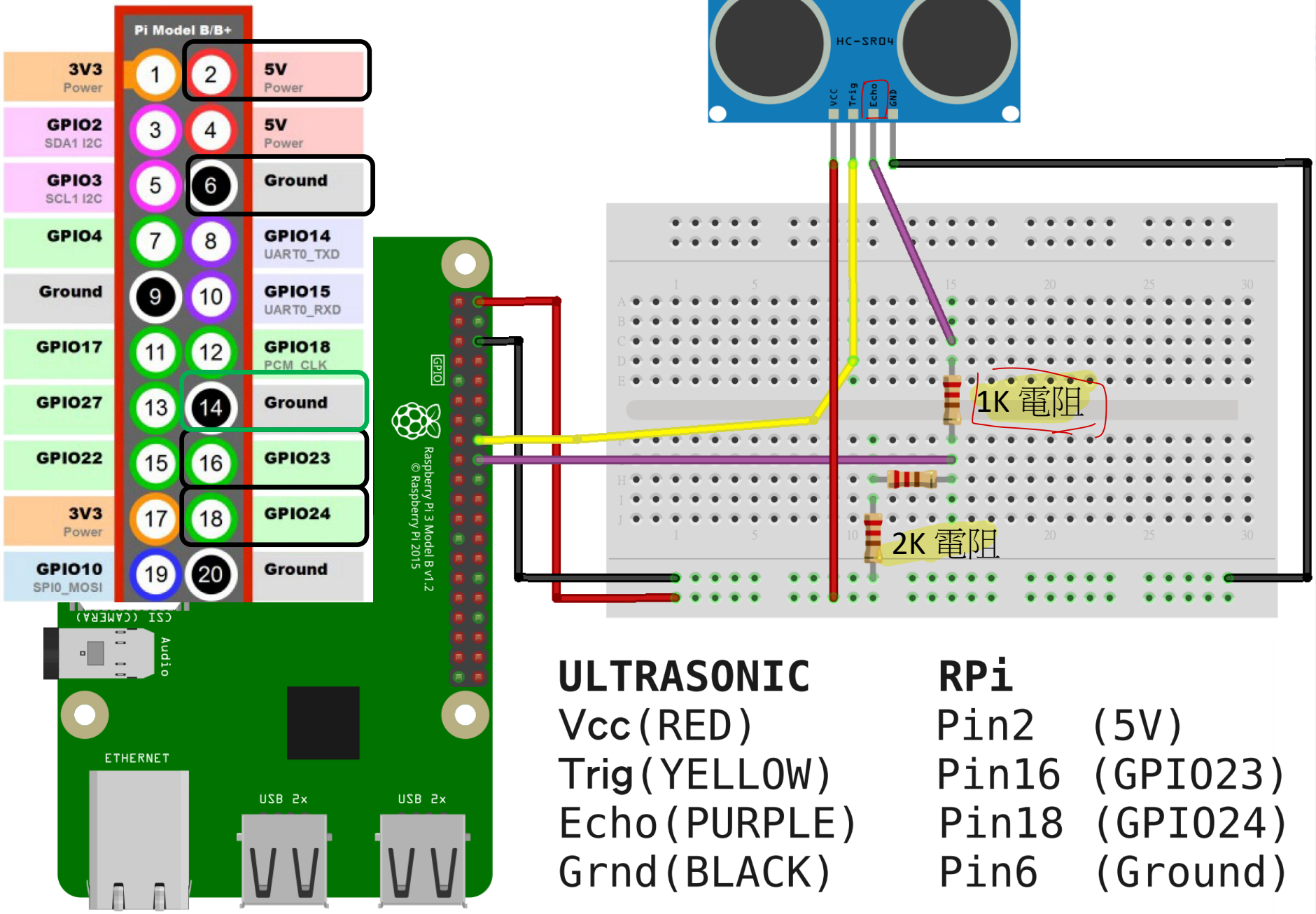
- PING sensor
- 內建發射 (40kHz) 與接收電路
- 根據發射與接收的時間差計算距離
- 特殊功能 :US-020(長距離) 、 US-100(溫度補償)



可得到時間差



VCC, Trigger, Echo, GND



ULTRASONIC	RPi
Vcc(RED)	Pin2 (5V)
Trig(YELLOW)	Pin16 (GPIO23)
Echo(PURPLE)	Pin18 (GPIO24)
Grnd(BLACK)	Pin6 (Ground)

Sound speed calculation

- Speed of sound
  - At 20°C (68°F), the speed is 343 m/s.
  - The approximate speed of sound (c) can be calculated from:

$$\mathbf{c_{air} = (331.3 + 0.606 * \theta)\ (m/s)}$$

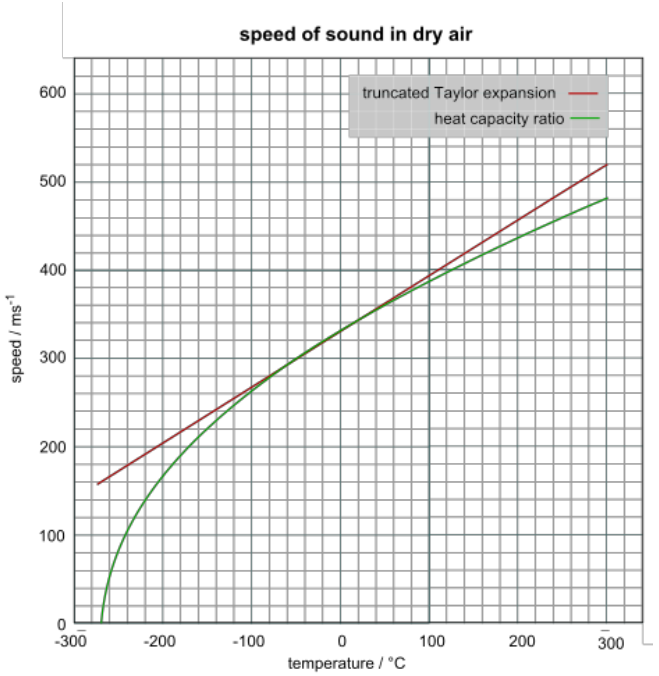
where θ is the Celsius temperature (°C).

$$Speed = \frac{Distance}{Time}$$

$$34300 = \frac{Distance}{Time/2}$$

$$17150 = \frac{Distance}{Time}$$

$$17150 \times Time = Distance$$

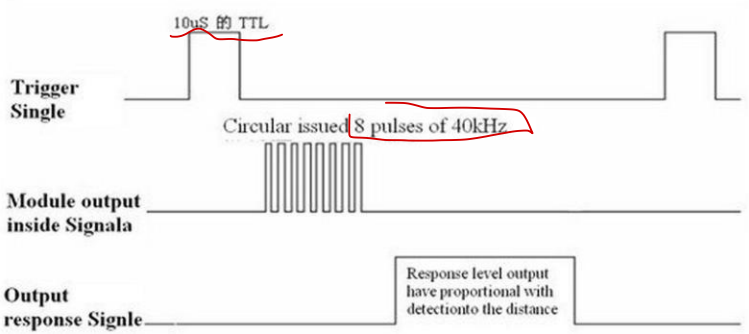


http://en.wikipedia.org/wiki/Speed_of_sound
https://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi

http://coopermaa2nd.blogspot.tw/2012/09/hc-sr04.html

2. Ultrasonic (HC-SR04)

4、Ultrasound timing diagram

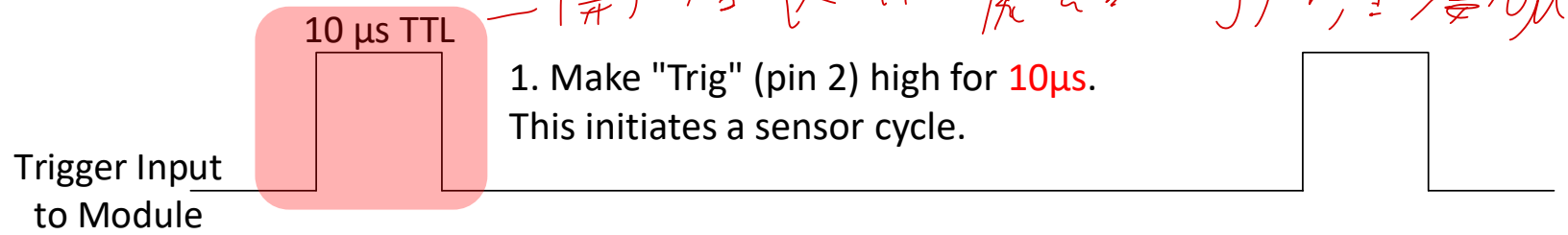


Working Voltage	DC 5 V
Working Current	15mA
Working Frequency	40Hz
Max Range	4m
Min Range	2cm
MeasuringAngle	15 degree
Trigger Input Signal	10uS TTL pulse
Echo Output Signal	Input TTL lever signal and the range in proportion
Dimension	45*20*15mm

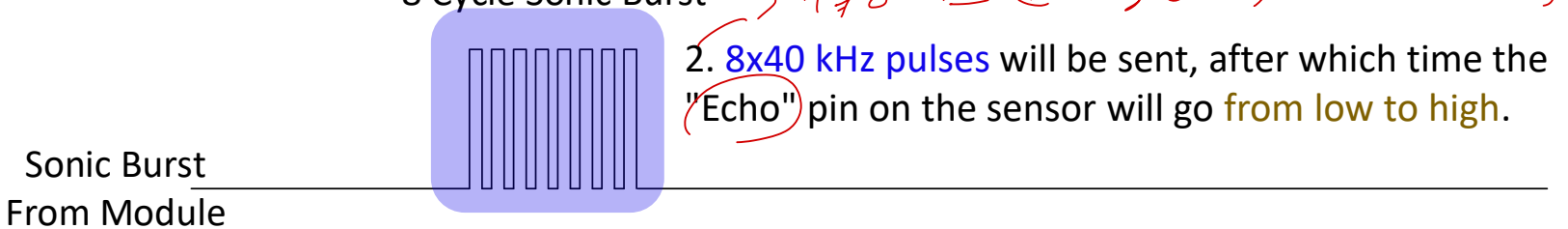
5.0 OPERATION

The timing diagram of HC-SR04 is shown. To start measurement, Trig of SR04 must receive a pulse of high (5V) for at least 10us, this will initiate the sensor will transmit out 8 cycle of ultrasonic burst at 40kHz and wait for the reflected ultrasonic burst. When the sensor detected ultrasonic from receiver, it will set the Echo pin to high (5V) and delay for a period (width) which proportion to distance. To obtain the distance, measure the width (Ton) of Echo pin.

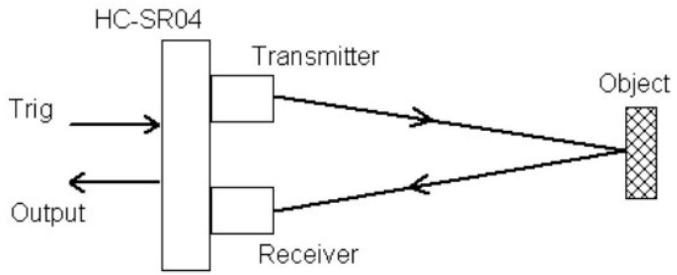
http://www.micropik.com/PDF/HCSR04.pdf



1. Make "Trig" (pin 2) high for 10µs.
This initiates a sensor cycle.

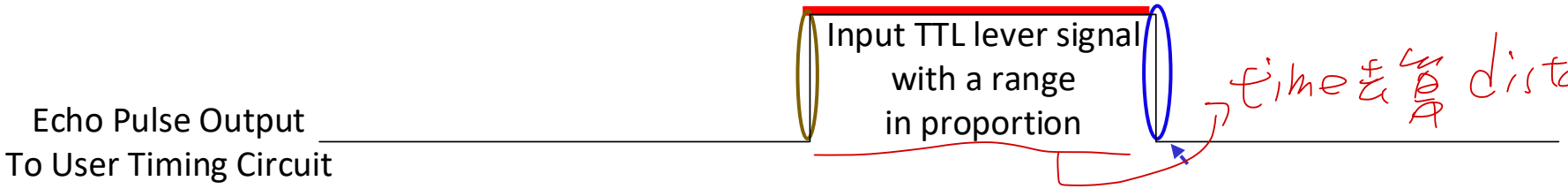


2. 8x40 kHz pulses will be sent, after which time the "Echo" pin on the sensor will go from low to high.



3. The 40kHz sound wave will bounce off the nearest object and return to the sensor.

VCC, Trigger, Echo, GND



4. When the sensor detects the reflected sound wave, the Echo pin will go low again.

5. The **distance** between the sensor and the detected object can be calculated based on **the length of time the Echo pin is high**.

2. HC-SR04 operation

1. Make "Trig" (pin 2) high for 10µs. This initiates a sensor cycle.
2. 8x40 kHz pulses will be sent, after which time the "Echo" pin on the sensor will go from low to high.
3. The 40kHz sound wave will bounce off the nearest object and return to the sensor.
4. When the sensor detects the reflected sound wave, the Echo pin will go low again.
5. The distance between the sensor and the detected object can be calculated based on the length of time the Echo pin is high.
6. If no object is detected, the Echo pin will stay high for 38ms and then go low.

Datasheet: http://www.micropik.com/PDF/HCSR04.pdf

2. Ultrasonic (HC-SR04)

- TRIG 腳位收到高電位 (3.3V) 後發送超聲波
- ECHO 腳位維持低電位 (0V), 收到回應後拉到高電位 (5V)
- Raspberry Pi 腳位的容忍電位為 3.3V, 會燒毀(QQ)
  - Sol: 將 ECHO 腳位的 5V 降壓為 3.3V 左右

$$\frac{3.3}{5} = \frac{R2}{1000 + R2}$$

$$0.66 = \frac{R2}{1000 + R2}$$

$$0.66(1000 + R2) = R2$$

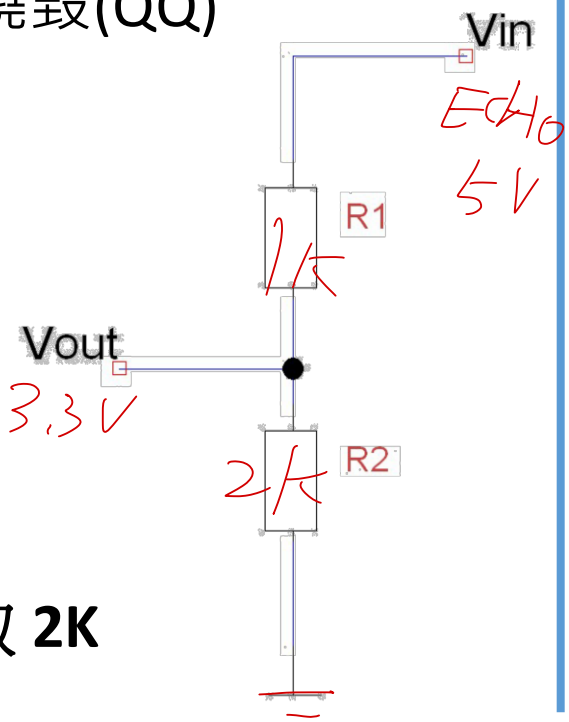
$$660 + 0.66R2 = R2$$

$$660 = 0.34R2$$

$$1941 = R2$$

$$Vout = Vin \times \frac{R2}{R1 + R2}$$

$$\frac{Vout}{Vin} = \frac{R2}{R1 + R2}$$



計算結果: R1=1K, R2 取 2K

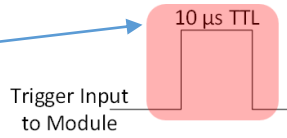
https://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi

2. HC-SR04 sample code

```
1  import RPi.GPIO as GPIO
2  import time
3  #GPIO.cleanup()
4  v = 343
5  TRIGGER_PIN = 16
6  ECHO_PIN = 18
7  GPIO.setmode(GPIO.BOARD)
8  GPIO.setup(TRIGGER_PIN,GPIO.OUT)
9  GPIO.setup(ECHO_PIN,GPIO.IN)
10
11 def measure():
12     GPIO.output(TRIGGER_PIN, GPIO.HIGH)
13     time.sleep(0.00001)      # 10uS
14     GPIO.output(TRIGGER_PIN, GPIO.LOW)
15     pulse_start = time.time()
16     while GPIO.input(ECHO_PIN) == GPIO.LOW:
17         pulse_start = time.time()
18     while GPIO.input(ECHO_PIN) == GPIO.HIGH:
19         pulse_end = time.time()
20     t = pulse_end - pulse_start
21     d = t * v
22     d = d/2
23     return d*100
24
25 print(measure())
26 GPIO.cleanup()
```

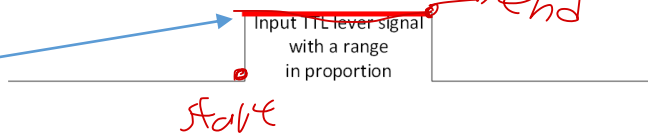
Load library

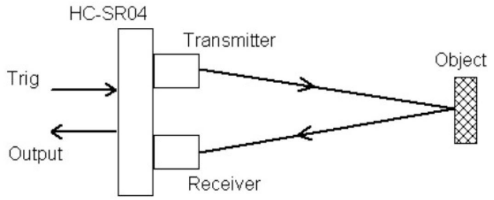
ULTRASONIC	RPi
Vcc(RED)	Pin2 (5V)
Trig(YELLOW)	Pin16 (GPIO23)
Echo(PURPLE)	Pin18 (GPIO24)
Grnd(BLACK)	Pin6 (Ground)



1. Make "Trig" (pin 2) high for 10µs. This initiates a sensor cycle.

5. The **distance** between the sensor and the detected object can be calculated based on **the length of time the Echo pin is high**





2. Ultrasonic (HC-SR04)

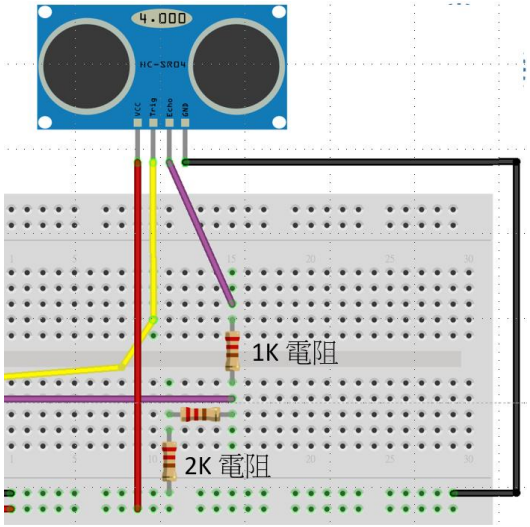
- Create a new Python code
  - nano ultrasonic_distance.py
- Run the code
  - sudo python ultrasonic_distance.py

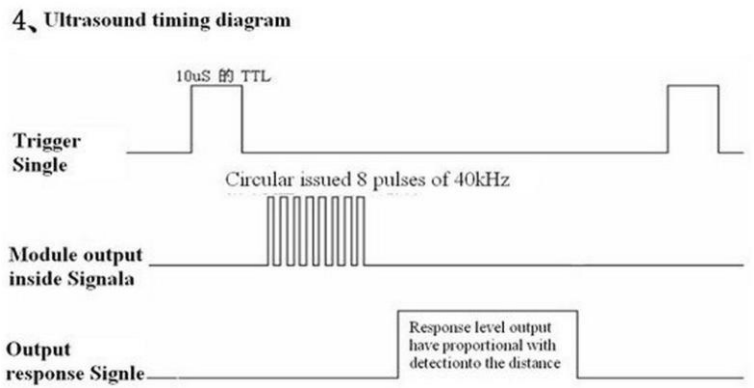
```
1  import RPi.GPIO as GPIO
2  import time
3  #GPIO.cleanup()
4  v = 343
5  TRIGGER_PIN = 16
6  ECHO_PIN = 18
7  GPIO.setmode(GPIO.BOARD)
8  GPIO.setup(TRIGGER_PIN,GPIO.OUT)
9  GPIO.setup(ECHO_PIN,GPIO.IN)
10
11 def measure():
12     GPIO.output(TRIGGER_PIN, GPIO.HIGH)
13     time.sleep(0.00001)      # 10uS
14     GPIO.output(TRIGGER_PIN, GPIO.LOW)
15     pulse_start = time.time()
16     while GPIO.input(ECHO_PIN) == GPIO.LOW:
```

This is picture!
Try to write code by yourself.
(you can also use Google Lens)

Discussion 2

- Why do we need to put resistors in the circuit?
- According to datasheet, what is the minimum value for measurement period?





Quiz 1

- Distance notification: Combine LED and ultrasonic
- The blinking frequency of the LED changes with distance
  - Short distance: high (distance < 50 cm)
  - Long distance: low (50 ≤ distance < 100 cm)
  - Ignore: none (distance ≥ 100 cm)

LED

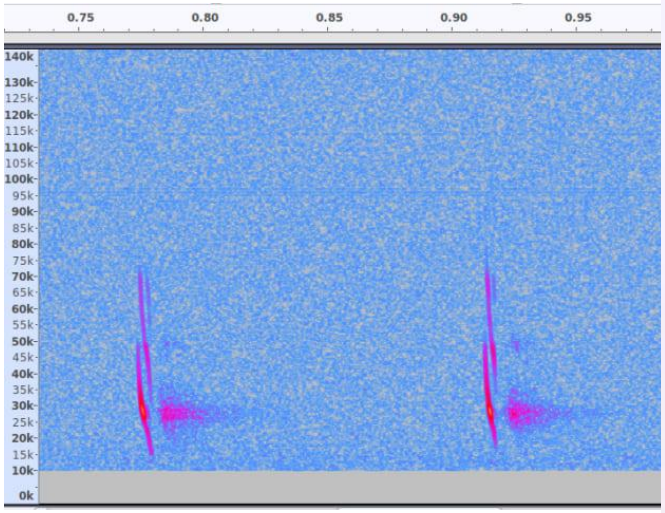
HC-SR04

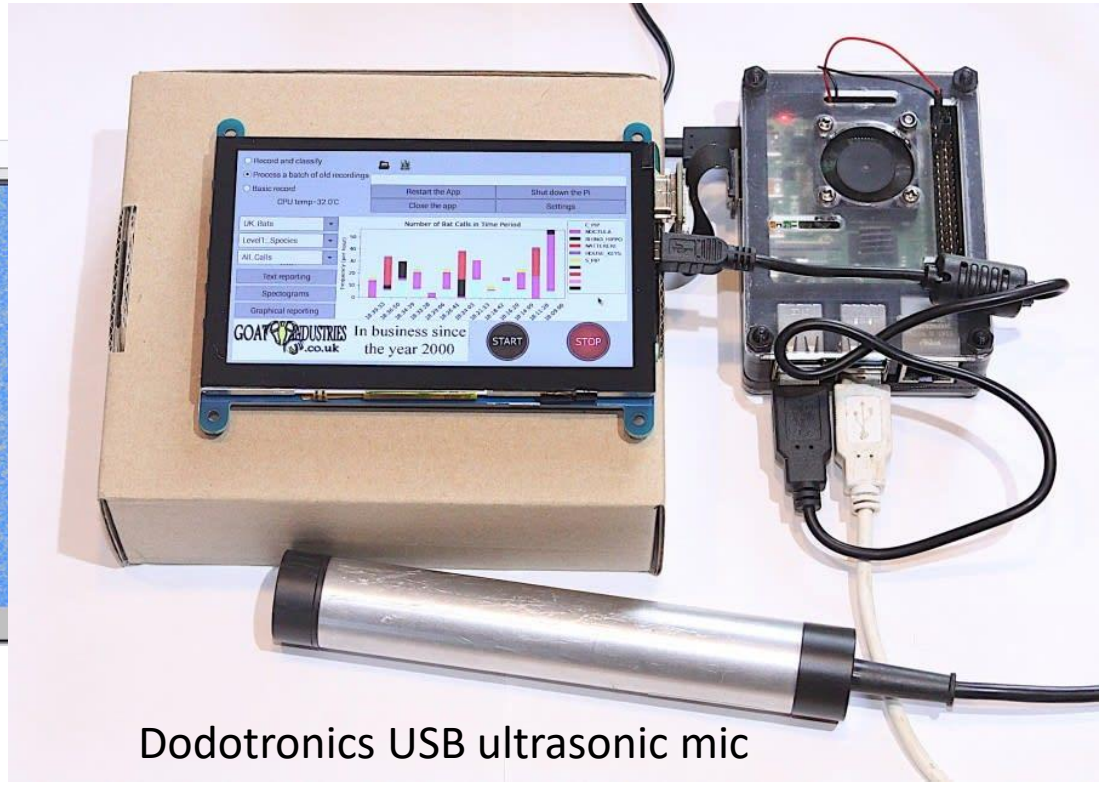
Ultrasonic

https://www.rs-online.com/designspark/designing-a-raspberry-pi-based-intelligent-ultrasonic-bat-detector-app

Another Ultrasonic APP

- Designing a Raspberry Pi Based Intelligent Ultrasonic Bat Detector App





Dodotronics USB ultrasonic mic

IMU

- 慣性測量單元（英文：Inertial measurement unit，簡稱 IMU），結合高可靠度的MEMS 6DOF慣性感測器，由三軸的陀螺儀(gyroscopes)和三個方向的加速度計(accelerometers)結合成慣性感測器
- 測量物體在三維空間中的角速度和加速度，以此解算出物體的姿態
  - IMU通常會安裝在被測物體的重心上
- IMU大多應用在需要進行運動控制的設備，如載具、機器人及移轉動設備上。也被使用在需要用姿態進行精密位移推算的場合，如軍事防禦: 潛艇、飛機、飛彈和太空飛行器的慣性導航設備等。

From 集研科技: http://www.systemaccess.com.tw/page/product/p02.aspx?kind=228&root=127&rootnum=167

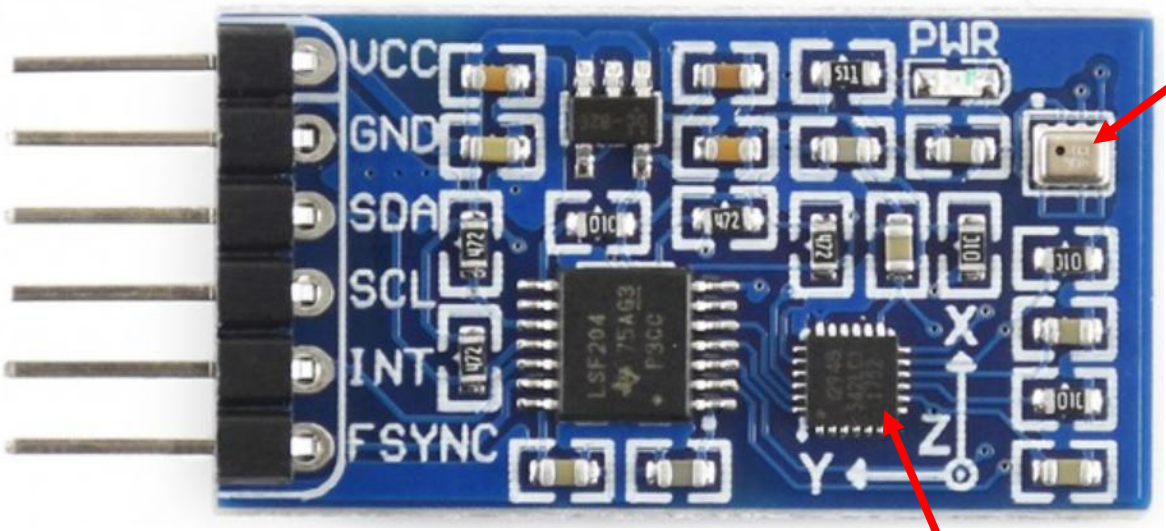
https://www.waveshare.com/wiki/10_DOF_IMU_Sensor_(D)

10 DOF IMU Sensor (D)

- 10 DOF:
  - 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer, and barometric altimeter
- Chip module
  - ICM20948 (加速度+陀螺儀+AK09916磁力計), BMP280 (氣壓計)
- Power supply: 3.3V~5V
- Accelerometer Features:
  - Resolution: 16 bit
  - Measurement range (configurable): ±2, ±4, ±8, ±16g
- Gyroscope Features:
  - Measurement range (configurable): ±250, ±500, ±1000, ±2000°/sec
- Compass/Magnetometer Features:
  - Measurement range: ±4900μT
- Barometric pressure sensor Features:
  - Measurement range: 300~1100hPa (Altitude: +9000m ~ -500m)

The ICM-20948 is a multi-chip module (MCM) consisting of two dies integrated into a single QFN package. One die houses a 3-axis gyroscope, a 3-axis accelerometer, and a Digital Motion Processor™ (DMP). The other die houses the AK09916 3-axis magnetometer from Asahi Kasei Microdevices Corporation.

10 DOF IMU

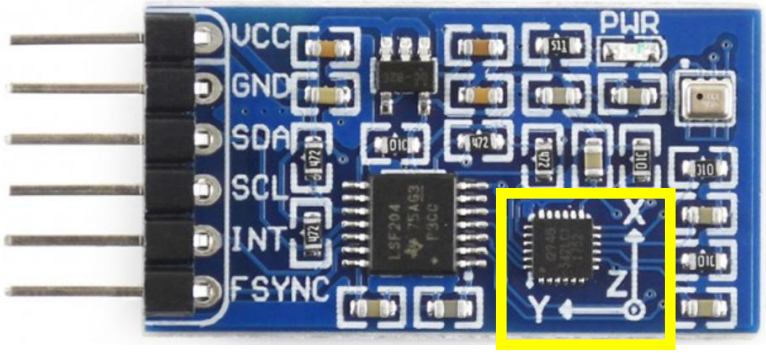


BMP280

TI LSF0204
(電壓電平轉換器)

ICM-20948

Orientation



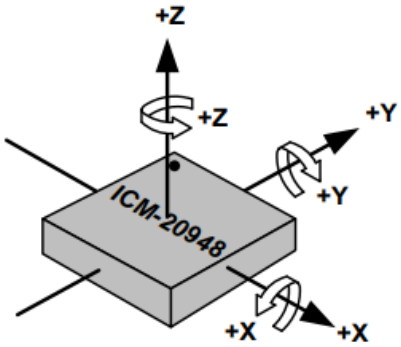
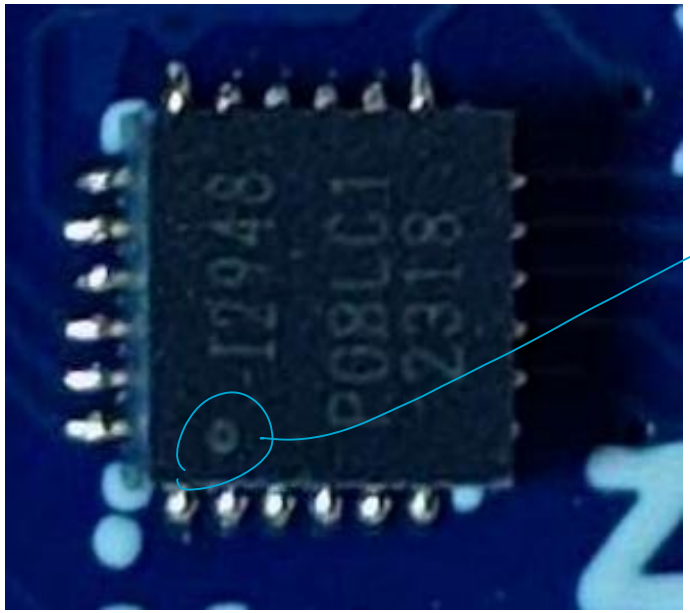


Figure 12. Orientation of Axes of Sensitivity and Polarity of Rotation



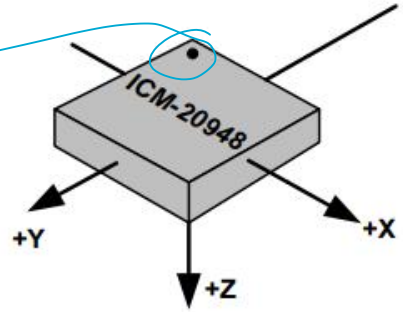
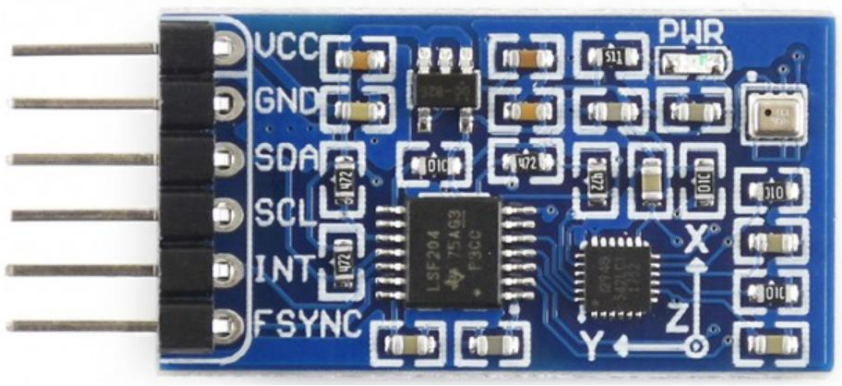
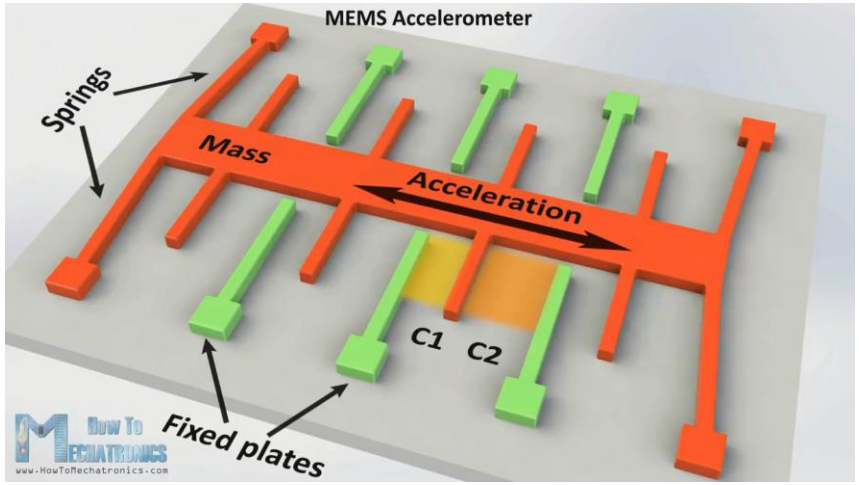


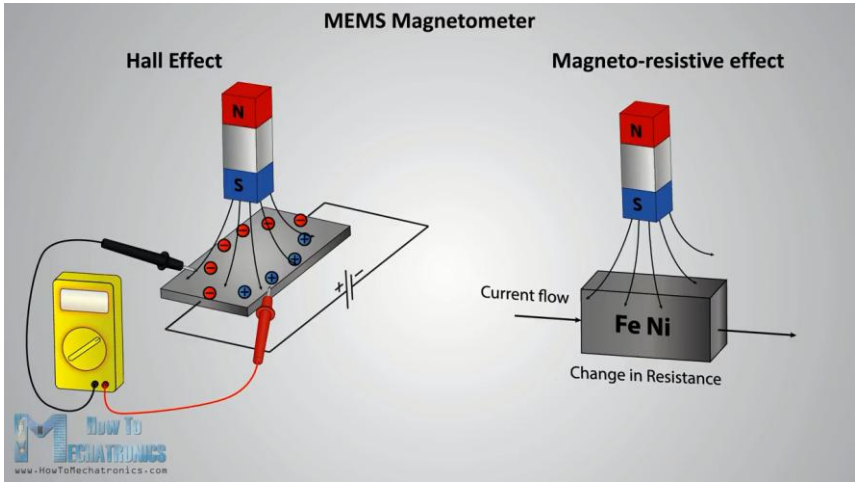
Figure 13. Orientation of Axes of Sensitivity for Magnetometer

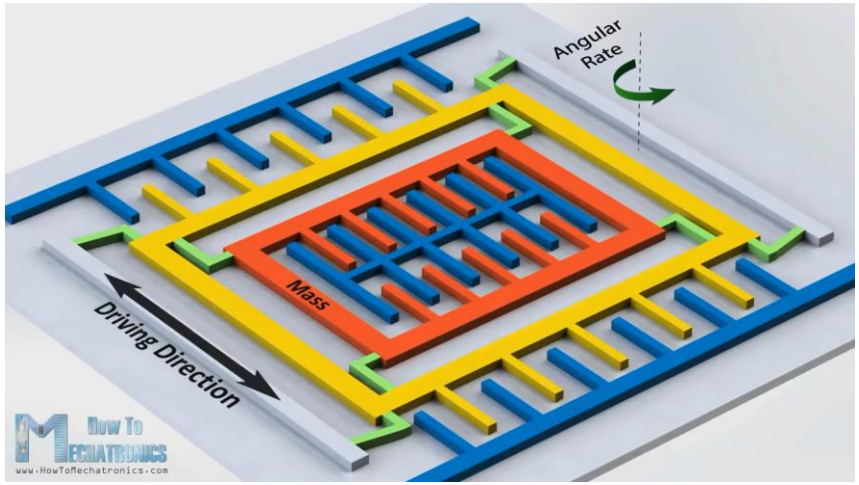
https://www.youtube.com/watch?v=eqZgxR6eRjo

How MEMS Work (Accelerometer, Gyroscope, Magnetometer)



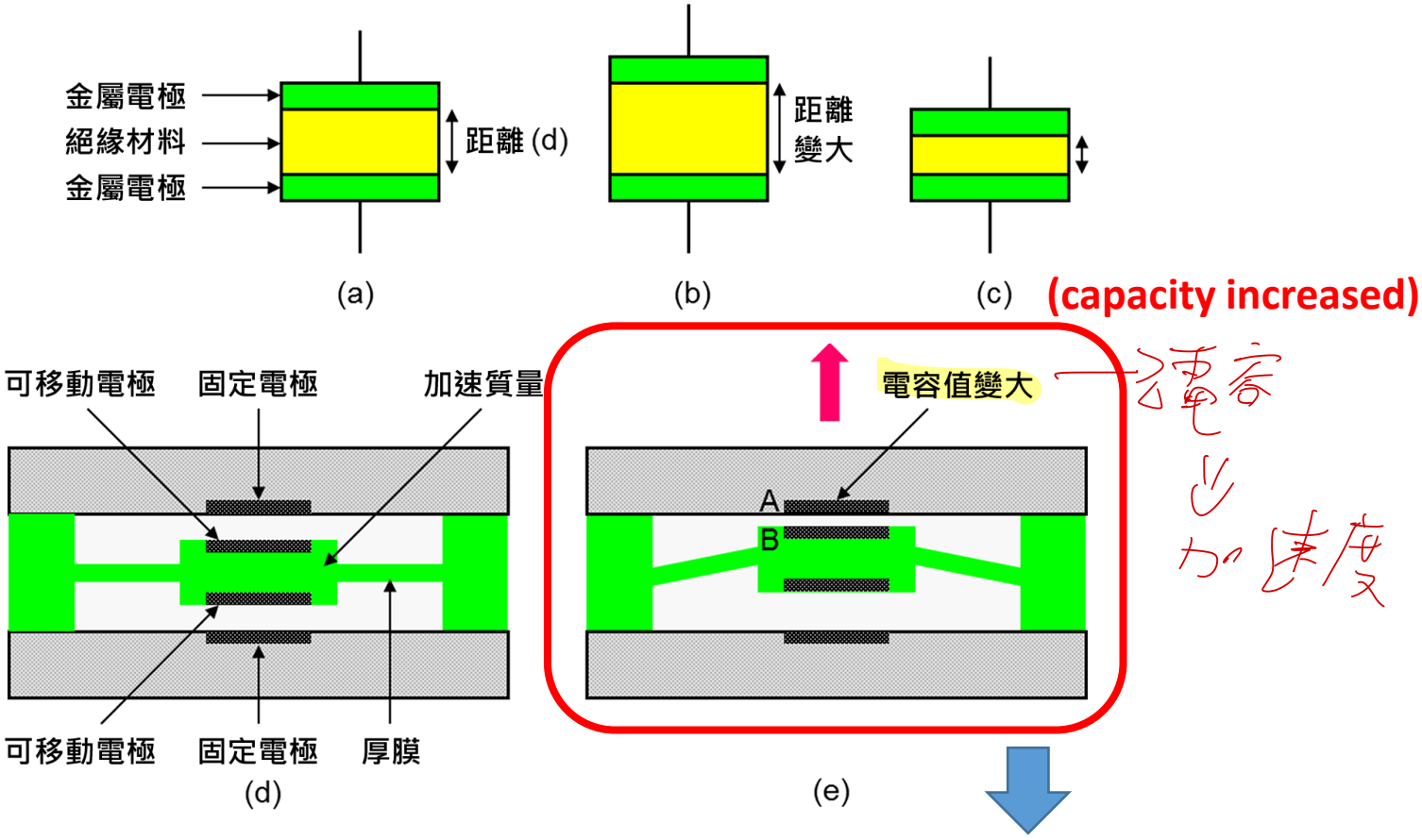






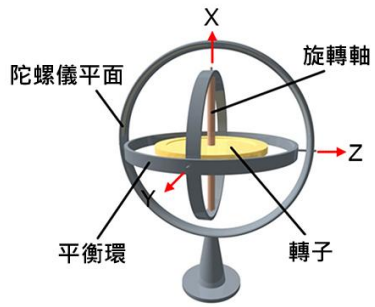
IMU. How MEMS Accelerometer Gyroscope Magnetometer Work & Arduino Tutorial

Sensor - Accelerometer

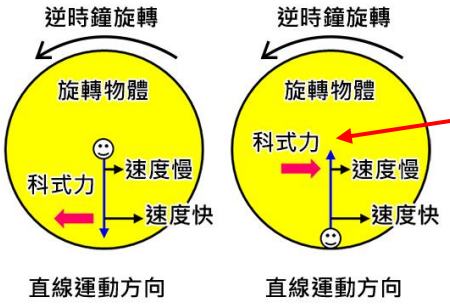


When you move the sensor, the distance (**capacity**) is changed.
We **use capacity to calculate acceleration**.

Sensor - Gyroscope

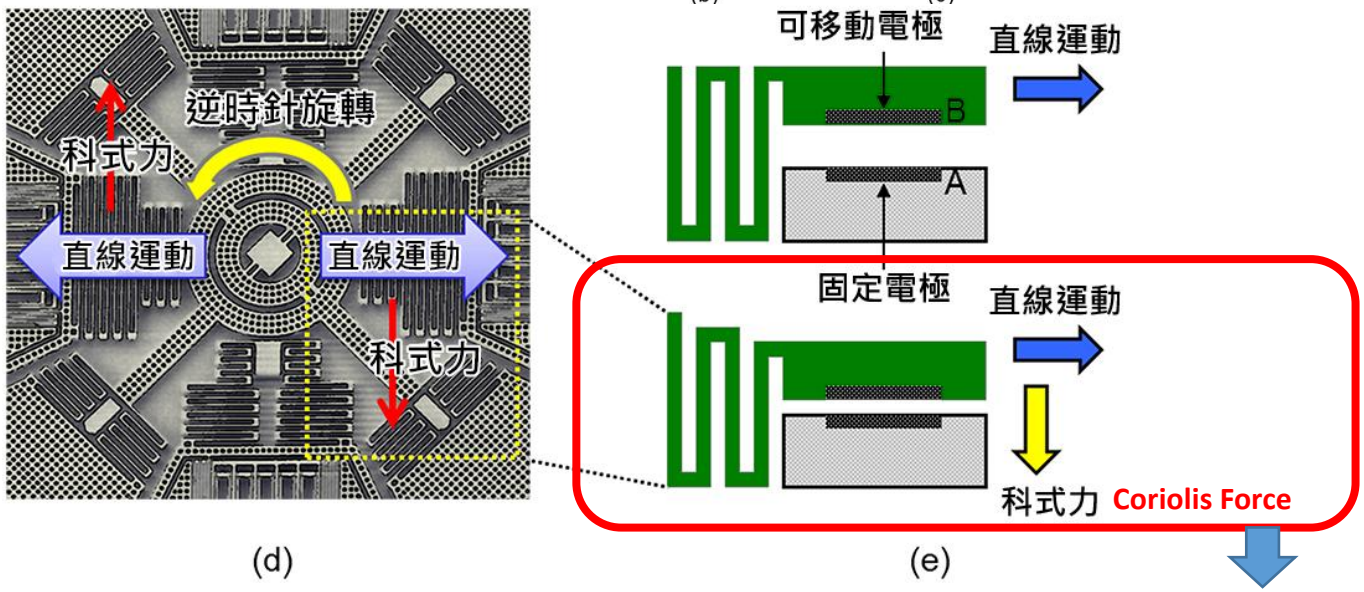


(a)



(b) (c)

Coriolis Force (科氏力)

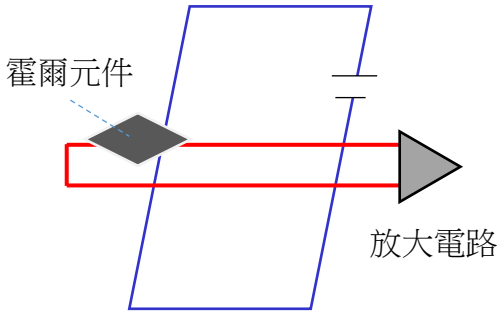


(d) (e)

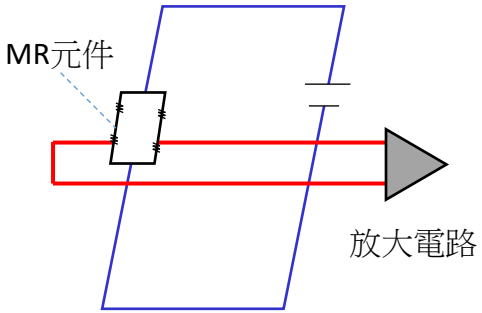
When you rotate the sensor, Coriolis Force change the distance (**capacity**).

We **use capacity to calculate Angular velocity (角速度).**

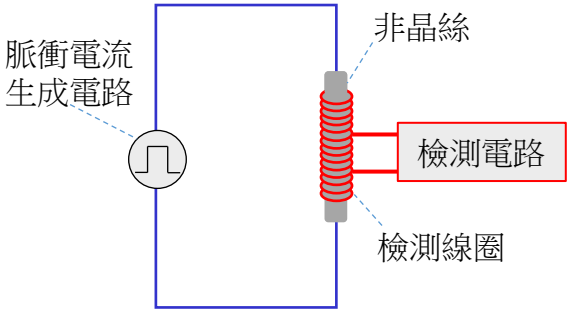
Sensor - Magnetometer



基於磁場變化
電壓隨著霍爾效應發生變化



基於磁場變化
MR元件的電阻發生變化



基於磁場變化
利用脈衝電流檢測線圈電壓

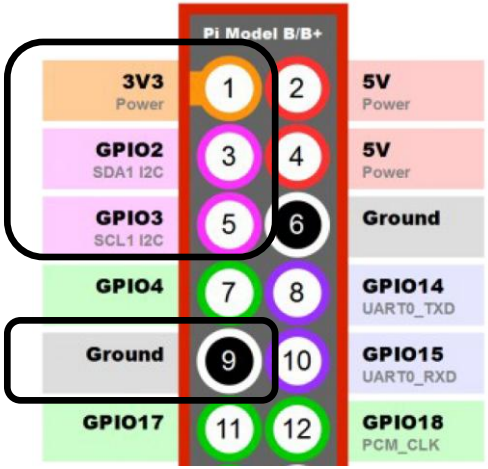
Hall sensor

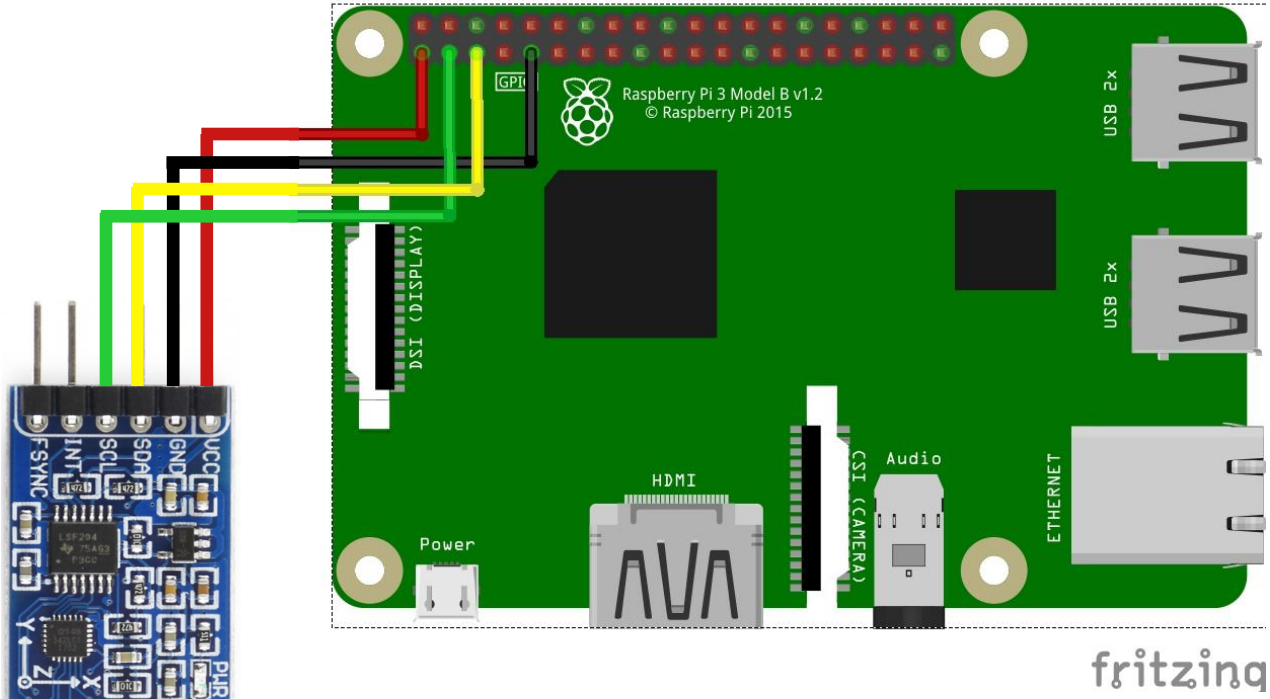
4.9 THREE-AXIS MEMS MAGNETOMETER WITH 16-BIT ADCS AND SIGNAL CONDITIONING

The 3-axis magnetometer uses highly sensitive Hall sensor technology. The magnetometer portion of the IC incorporates magnetic sensors for detecting terrestrial magnetism in the X-, Y-, and Z-Axes, a sensor driving circuit, a signal amplifier chain, and an arithmetic circuit for processing the signal from each sensor. Each ADC has a 16-bit resolution and a full scale range of ±4900 µT.

Connect IMU

VCC	**Pin 1 (3.3V), Red line**
GND	**Pin 9 (Ground), Black line**
SCL	**Pin 5 (SCL), Yellow line**
SDA	**Pin 3 (SDA), Green line**

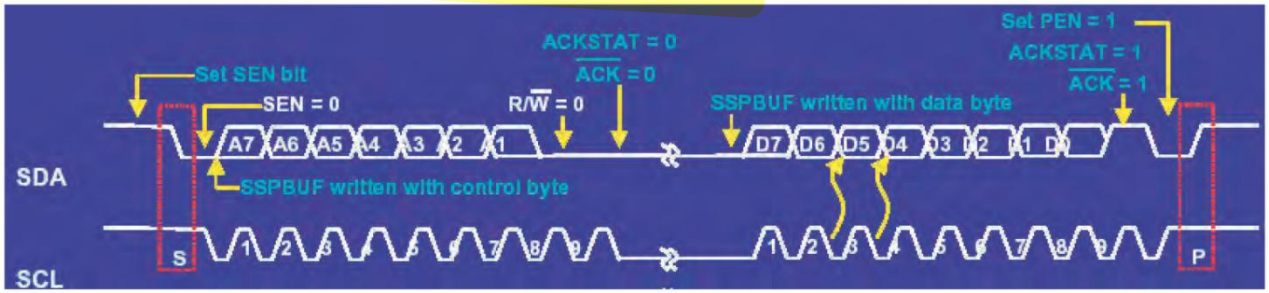




這個念square喔!!

What is I2C?

- I2C全名為Inter-IC，它是一種半雙工同步多組設備匯流排，只需要兩條信號線：串列資料線 (SDA) 及串列時脈線 (SCL)。



- 在傳送資料過程中共有三種類型信號，分別是：開始信號、結束信號和應答信號。
  - 開始信號：SCL為高電位時，SDA由高電位降為低電位，開始傳送資料。
  - 結束信號：SCL為高電位時，SDA由低電位升為高電位，結束傳送資料。
  - 應答信號：收到 8bit 資料後，向發送資料的IC發出特定的低電位脈衝
- 資料讀取方式：
  - 當 SCL 由低電位升為高電位時，讀取 SDA 的資料。

Before connecting IMU

- Install I2C tool to check the state:
  - **sudo apt install i2c-tools**
- No I2C device attached
  - List devices: **sudo ls -al /dev/*i2c***

```
pi@raspberrypi:~$ ls -al /dev/*i2c*
ls: cannot access /dev/*i2c*: No such file or directory
```

  - Show I2C address: **sudo i2cdetect -y 1** => no I2C

```
pi@raspberrypi:~/gy801$ sudo i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

After connecting IMU

- I2C device attached
  - List devices: **ls -al /dev/*i2c***

```
pi@raspberrypi:~$ ls -al /dev/*i2c*
crw-rw---- 1 root i2c 89, 1 Mar 12 11:17 /dev/i2c-1
```

  - Show I2C address: **sudo i2cdetect -y 1** => IMUが生存

```
pi@raspberrypi:~ $ sudo i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- 68 -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- 77
```

Features

- Gyroscope Features:
  - Digital **X, Y, Z-axis angular rate sensors** with programmable full-scale ranges of ±250, ±500, ±1000, ±2000 dps.
  - Integrated 16-bit ADC with configurable ODR and low-pass filters.
- Accelerometer Features:
  - Digital **X, Y, Z-axis accelerometer** with programmable full-scale ranges of ±2g, ±4g, ±8g, ±16g.
  - Integrated 16-bit ADC, configurable ODR, and low-pass filters, with motion wake-up interrupt for low-power operation.
- Magnetometer Features:
  - **3-axis Hall-effect sensor** with magnetic concentrator.
  - Wide dynamic range, high resolution, low power consumption, 16-bit output, full-scale range of ±4900 µT.

GYROSCOPE

- The ICM-20948 consists of **three independent vibratory MEMS rate gyroscopes**, which detect rotation about the X-, Y-, and Z-Axes.
- When the gyros are rotated about any of the sense axes, the **Coriolis Effect** causes a vibration that is detected by a capacitive pickoff.
- The resulting signal is amplified, demodulated, and filtered to **produce a voltage that is proportional to the angular rate**.
- This **voltage is digitized** using individual on-chip 16-bit Analog-to-Digital Converters (ADCs) to sample each axis.
- The full-scale range of the gyro sensors may be digitally programmed to ±250, ±500, ±1000, or ±2000 degrees per second (dps).

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
GYROSCOPE SENSITIVITY						
Full-Scale Range	GYRO_FS_SEL=0		±250		dps	1
	GYRO_FS_SEL=1		±500		dps	1
	GYRO_FS_SEL=2		±1000		dps	1
	GYRO_FS_SEL=3		±2000		dps	1
Gyroscope ADC Word Length			16		bits	1
Sensitivity Scale Factor	GYRO_FS_SEL=0		131		LSB/(dps)	1
	GYRO_FS_SEL=1		65.5		LSB/(dps)	1
	GYRO_FS_SEL=2		32.8		LSB/(dps)	1
	GYRO_FS_SEL=3		16.4		LSB/(dps)	1

3.1 GYROSCOPE SPECIFICATIONS

ACCELEROMETER

- The ICM-20948's 3-Axis accelerometer uses **separate proof masses for each axis**.
- Acceleration along a particular axis induces displacement on the corresponding proof mass, and **capacitive sensors detect the displacement differentially**.
- When the device is placed on a **flat surface**, it will measure 0g on the X- and Y-axes and **+1g on the Z-axis**. -> beange g↓
- Each sensor has a dedicated sigma-delta ADC for providing **digital outputs**.
- The full scale range of the digital output can be ±2g, ±4g, ±8g, or ±16g.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
ACCELEROMETER SENSITIVITY						
Full-Scale Range	ACCEL_FS=0		±2		*G*	1
	ACCEL_FS=1		±4		*G*	1
	ACCEL_FS=2		±8		*G*	1
	ACCEL_FS=3		±16		*G*	1
ADC Word Length	Output in two's complement format		16		Bits	1
Sensitivity Scale Factor	ACCEL_FS=0		16,384		LSB/*g*	1
	ACCEL_FS=1		8,192		LSB/*g*	1
	ACCEL_FS=2		4,096		LSB/*g*	1
	ACCEL_FS=3		2,048		LSB/*g*	1

3.2 ACCELEROMETER SPECIFICATIONS

MAGNETOMETER

- The 3-axis magnetometer uses highly **sensitive Hall sensor** technology.
- The magnetometer portion of the IC incorporates magnetic sensors for **detecting terrestrial magnetism** in the X-, Y-, and Z-Axes, a sensor driving circuit, a signal amplifier chain, and an arithmetic circuit for processing the signal from each sensor.
- Each ADC has a 16-bit resolution and a full scale range of ±4900 µT.

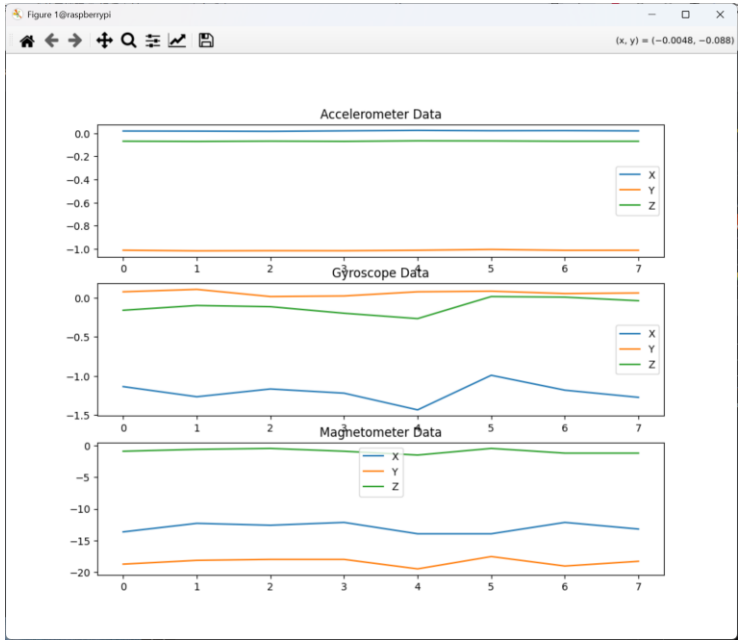
PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
MAGNETOMETER SENSITIVITY						
Full-Scale Range			±4900		µT	1
Output Resolution			16		bits	1
Sensitivity Scale Factor			0.15		µT / LSB	1

3.3 MAGNETOMETER SPECIFICATIONS

Sample code

- 10dof_imu_reading.py
  - 在終端機連續測量, 顯示目前的數值
- 10dof_imu_save_pic.py
  - 同上, 不過結束時會將數值儲存成圖檔
- 10dof_imu_plot.py → note
  - 使用 ssh -X 進行 X11 轉發
  - 可在ssh client顯示圖形化程式

```
pi@raspberrypi:~/lab2 $ python 10dof_imu_reading.py

Accel: -0.01 -1.02 -0.05
Gyro:  -2.08 -0.33 02.53
Mag:   -15.30 -18.15 -6.75

Accel: -0.00 -1.02 -0.06
Gyro:  -1.45 -0.11 -0.84
Mag:   -15.30 -18.15 -6.00
```



Troubleshoot

- IMU搖晃時容易中斷?
  - Sol: 換杜邦線 or 麵包版的位置試試
- 讀不到IMU?
  - Sol: 用sudo i2cdetect -y 1檢查是否有偵測到

```
pi@raspberrypi:~/lab2 $ python 10dof_imu_reading.py
Traceback (most recent call last):
  File "/home/pi/lab2/10dof_imu_reading.py", line 290, in <module>
    imu = ICM20948()
  File "/home/pi/lab2/10dof_imu_reading.py", line 254, in __init__
    self.bank(0)
  File "/home/pi/lab2/10dof_imu_reading.py", line 92, in bank
    self.write(ICM20948_BANK_SEL, value << 4)
  File "/home/pi/lab2/10dof_imu_reading.py", line 71, in write
    self._bus.write_byte_data(self._addr, reg, value)
  File "/home/pi/.local/lib/python3.9/site-packages/smbus2/smbus2.py", line 457, in write_byte_data
    ioctl(self.fd, I2C_SMBUS, msg)
OSError: [Errno 5] Input/output error
```

```
pi@raspberrypi:~/lab2 $ sudo i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

```
pi@raspberrypi:~/lab2 $ sudo i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- 68 -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- 77
```

Code and datasheet

```
7   # 定義ICM-20948和AK09916的暫存器地址和參數
8   CHIP_ID = 0xEA
9   I2C_ADDR = 0x68
10  I2C_ADDR_ALT = 0x69
11  ICM20948_BANK_SEL = 0x7f
12
13  # Bank 3
14  ICM20948_I2C_MST_ODR_CONFIG = 0x00
15  ICM20948_I2C_MST_CTRL = 0x01
16  ICM20948_I2C_MST_DELAY_CTRL = 0x02
17  ICM20948_I2C_SLV0_ADDR = 0x03
18  ICM20948_I2C_SLV0_REG = 0x04
19  ICM20948_I2C_SLV0_CTRL = 0x05
20  ICM20948_I2C_SLV0_DO = 0x06
21
22  # Bank 0
23  ICM20948_WHO_AM_I = 0x00
24  ICM20948_USER_CTRL = 0x03
25  ICM20948_PWR_MGMT_1 = 0x06
26  ICM20948_PWR_MGMT_2 = 0x07
27  ICM20948_INT_PIN_CFG = 0x0F
```

TDK InvenSense

7 REGISTER MAP FOR GYROSCOPE AND ACCELE

The following table lists the register map for the ICM-20948, for user banks 0, 1,

7.1 USER BANK 0 REGISTER MAP

ADDR (HEX)	ADDR (DEC.)	REGISTER NAME	SERIAL I/F	BIT7	BIT6	BIT5	BIT4
00	0	WHO_AM_I	R	WHO_AM_I[7:0]			
03	3	USER_CTRL	R/W	DMP_EN	FIFO_EN	I2C_MST_EN	I2C_IF_DIS
05	5	LP_CONFIG	R/W		I2C_MST_CYCLE	ACCEL_CYCLE	GYRO_CYCL
06	6	PWR_MGMT_1	R/W	DEVICE_RESET	SLEEP	LP_EN	-
07	7	PWR_MGMT_2	R/W	-		DISABLE_ACCEL	
0F	15	INT_PIN_CFG	R/W	INT1_ACTL	INT1_OPEN	INT1_LATCH_INT_EN	INT_ANYRD_2CLEAR
10	16	INT_ENABLE	R/W	REG_WOF_EN	-		
11	17	INT_ENABLE_1	R/W	-			
12	18	INT_ENABLE_2	R/W	-			FIFO_OVER
13	19	INT_ENABLE_3	R/W	-			FIFO_WM_
17	23	I2C_MST_STATUS	R/C	PASS_THROUGH	I2C_SLV4_DONE	I2C_LOST_ARB	I2C_SLV4_NCK
19	25	INT_STATUS	R/C	-			
1A	26	INT_STATUS_1	R/C	-			
1B	27	INT_STATUS_2	R/C	-			FIFO_OVER
1C	28	INT_STATUS_3	R/C	-			FIFO_WM_
28	40	DELAY_TIMEH	R	DELAY_TIMEH[7:0]			
29	41	DELAY_TIMEL	R	DELAY_TIMEL[7:0]			

Measurement data

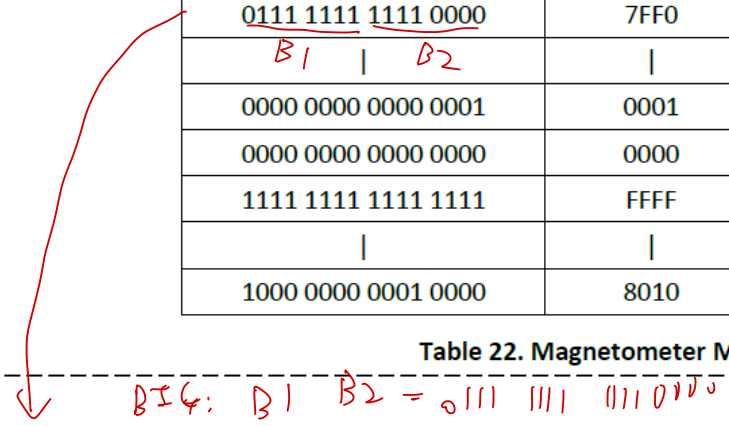
- two's complement

```
ax, ay, az, gx, gy, gz = struct.unpack(">hhhhhh", bytearray(data))
x, y, z = struct.unpack("<hhh", bytearray(data))
```

Measurement data is stored in two's complement and Little Endian format. Measurement range of each axis is from --32752 to 32752 in 16-bit output.

MEASUREMENT DATA (EACH AXIS) [15:0]			MAGNETIC FLUX DENSITY [µT]
TWO'S COMPLEMENT	HEX	DECIMAL	
0111 1111 1111 0000	7FF0	32752	4912(max.)
|	|	|	|
0000 0000 0000 0001	0001	1	0.15
0000 0000 0000 0000	0000	0	0
1111 1111 1111 1111	FFFF	-1	-0.15
|	|	|	|
1000 0000 0001 0000	8010	-32752	-4912(min.)

Table 22. Magnetometer Measurement Data Format



Convert unit

- Gyroscope

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
GYROSCOPE SENSITIVITY						
Full-Scale Range	GYRO_FS_SEL=0		±250		dps	1
	GYRO_FS_SEL=1		±500		dps	1
	GYRO_FS_SEL=2		±1000		dps	1
	GYRO_FS_SEL=3		±2000		dps	1
Gyroscope ADC Word Length			16		bits	1
Sensitivity Scale Factor	GYRO_FS_SEL=0		131		LSB/(dps)	1
	GYRO_FS_SEL=1		65.5		LSB/(dps)	1
	GYRO_FS_SEL=2		32.8		LSB/(dps)	1
	GYRO_FS_SEL=3		16.4		LSB/(dps)	1

- Accelerometer

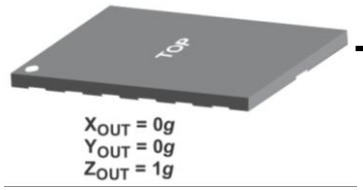
PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
ACCELEROMETER SENSITIVITY						
Full-Scale Range	ACCEL_FS=0		±2		*G*	1
	ACCEL_FS=1		±4		*G*	1
	ACCEL_FS=2		±8		*G*	1
	ACCEL_FS=3		±16		*G*	1
ADC Word Length	Output in two's complement format		16		Bits	1
Sensitivity Scale Factor	ACCEL_FS=0		16,384		LSB/*g*	1
	ACCEL_FS=1		8,192		LSB/*g*	1
	ACCEL_FS=2		4,096		LSB/*g*	1
	ACCEL_FS=3		2,048		LSB/*g*	1

- Magnetometer

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
MAGNETOMETER SENSITIVITY						
Full-Scale Range			±4900		µT	1
Output Resolution			16		bits	1
Sensitivity Scale Factor			0.15		µT / LSB	1

ACC Brief summary

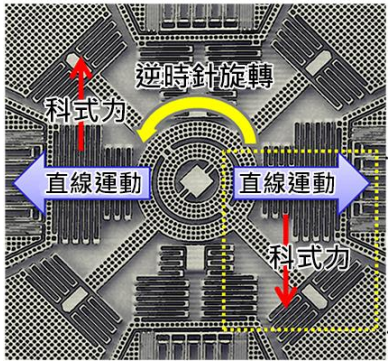
- What can we obtain from accelerometer?
  - 1. Movement along with xyz-axis
    - Unit: byte, G, or m/s^2
  - 2. Posture (Roll, Pitch)
    - Unit: Radians or Degree

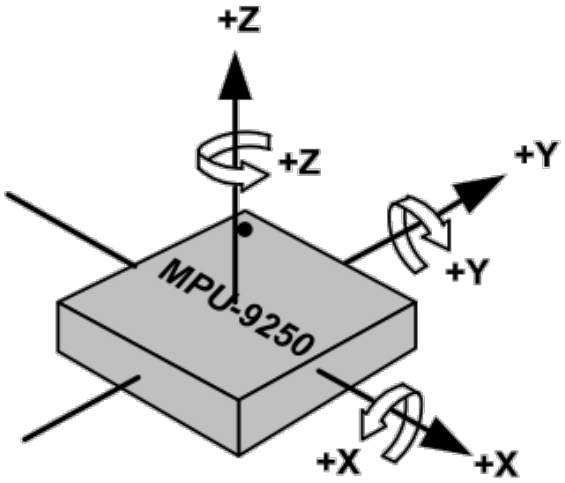


Read value → ACC data: Ax, Ay, Az (byte, G, or m/s^2) → calculation → Posture: Roll, Pitch (Radians or Degree)

Gyro Brief summary

- What can we obtain from gyroscope?
  - 1. rotation (angular velocity)
    - Unit: byte, DPS (Degree per Second)
  - 2. rotated angle (integral “angular velocity” to “angle”)
    - Unit: degree. You can convert it to rad. 積分修正

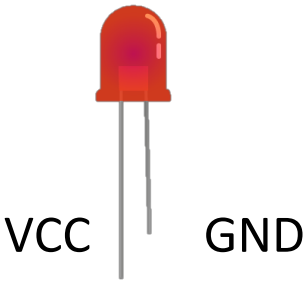




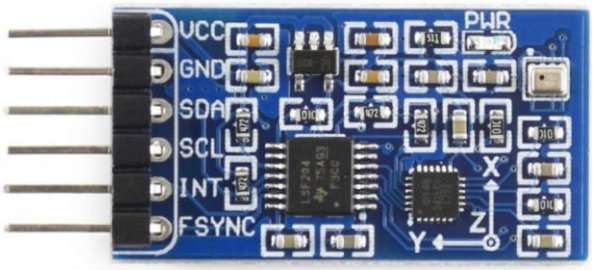
Quiz 2

- Flat notification: Combine LED and IMU
- Use IMU to detect whether the object is placed on a flat surface. When the object is stable, turn on LED to notify user. Else, the LED is turn off.

LED



ICM20948



IMU
(Inertial measurement unit)

Summary

- Labs: access GPIO pins and read sensing data (自主練習)
  - 1. LED
  - 2. Ultrasonic
  - 3. IMU

jas

- Write down the answer for discussion, upload to e-campus. Deadline (before next class): 13:10, 3/21(Fri.)
  - Discussion 1: Identify the resistors
  - Discussion 2: ultrasonic questions
  - 書面問答, 請上傳到e3

- [Quiz 1] Distance notification: Combine LED and ultrasonic
- [Quiz 2] Flat notification: Combine LED and IMU
  - 請於課堂上完成, 找助教demo