樹莓派-實驗二：距離感測

國立中央大學 通訊工程學系

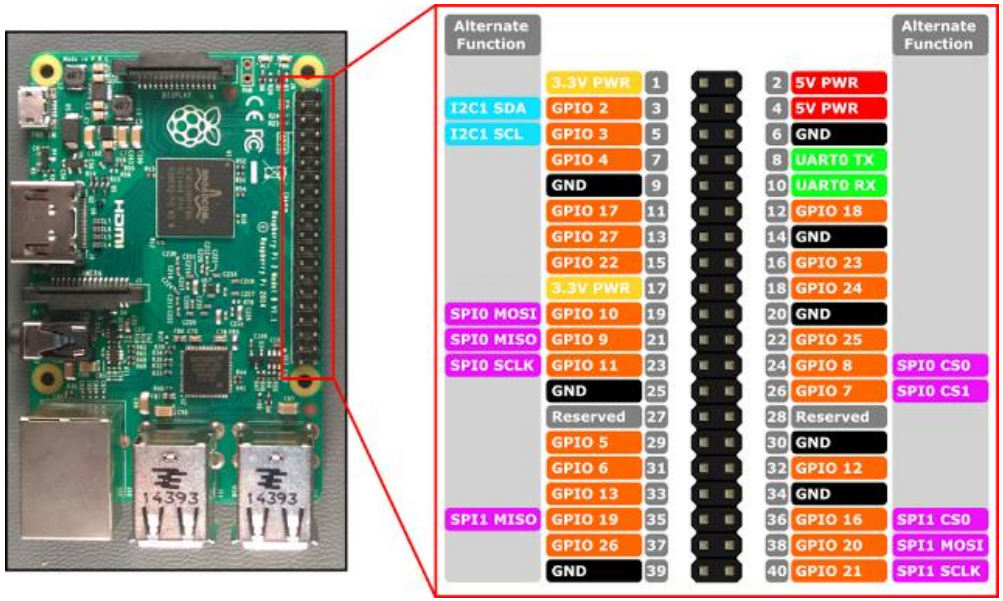
OUTLINE

- 實驗硬體介紹
- 距離感測實驗
- 倒車雷達
- 課堂實作

實驗硬體介紹

Raspberry Pi – BCM 腳位

- GPIO（General-purpose input/output），通用型之輸入輸出的簡稱，可透過指令設成輸出或輸入，設為輸出時也可透過指令做開關的動作

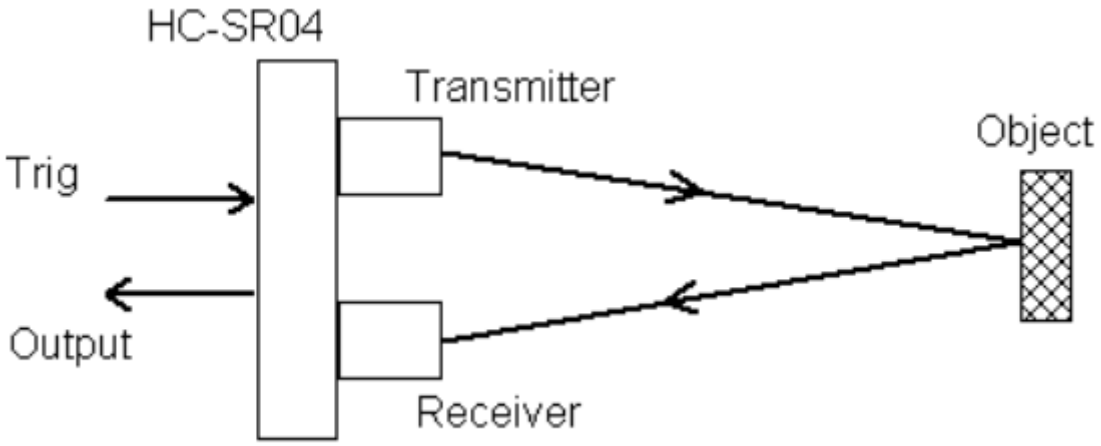


超聲波感測器

- 超音波感測器是由超音波發射器、接收器和控制電路所組成。當它被觸發的時候，會發射一連串 40 kHz 的聲波並且從離它最近的物體接收回音。

超聲波感測器原理

- 如下圖所示，超音波測量距離的方法，是測量聲音在感測器與物體之間往返經過的時間：



超聲波感測器速度計算

- 我們可以透過超聲波傳送/接收的時間差，以聲音速度340.3m/s來換算出物體與感測器間的距離，公式如下：

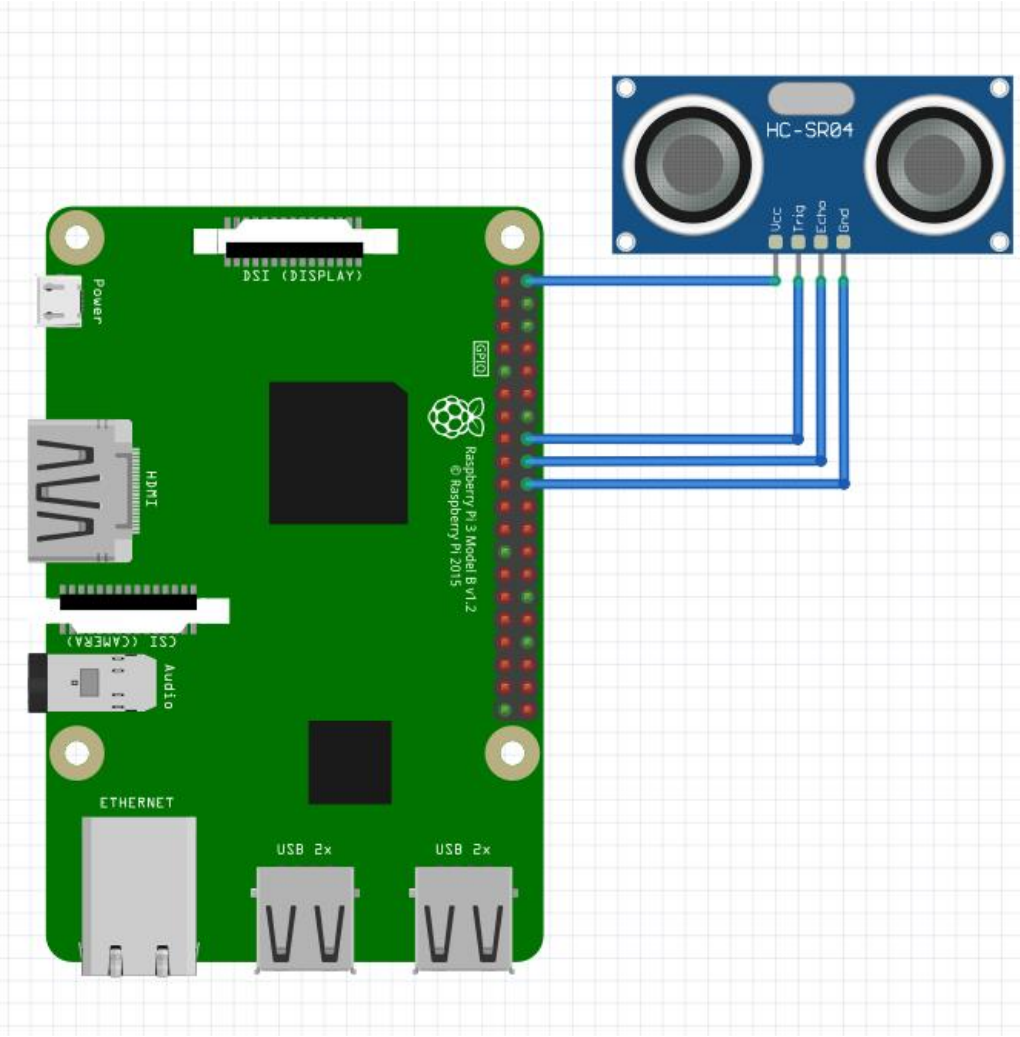
$$time(sec) * \frac{34030(cm)}{2} = time(sec) * 17015(cm)$$

距離感測實驗

實驗步驟(1/5)

- 將感測器與樹梅派用杜邦線連接

- Vcc ：腳位2
- Trig ：GPIO23
- Echo：GPIO24
- GND：腳位20



實驗步驟(2/5)

- 將TRIG和ECHO腳位設為23以及24，並設為輸出與輸入

```
TRIG = 23
ECHO = 24

GPIO.setup(TRIG, GPIO.OUT)
GPIO.setup(ECHO, GPIO.IN)
```

實驗步驟(3/5)

- 先將TRIG設為false，休眠0.1秒後再設為True，經過0.0001秒後再設為False
- GPIO接收到ECHO為0，設成開始時間，為1設成結束時間
- 兩個相減乘上17015即是距離(單位為公分)

```
def get_distance():

    GPIO.output(TRIG, False)
    time.sleep(0.1)
    GPIO.output(TRIG, True)
    time.sleep(0.00001)
    GPIO.output(TRIG, False)

    while GPIO.input(ECHO)==0:
        start = time.time()

    while GPIO.input(ECHO)==1:
        end = time.time()

    return (end - start) * 17015
```

實驗步驟(4/5)

- 將結果依照0.5秒的間隔做輸出

```
while True:
    distance = get_distance()
    print (distance)
    time.sleep(0.5)
```

實驗步驟(5/5)

- 輸出結果如圖

Shell

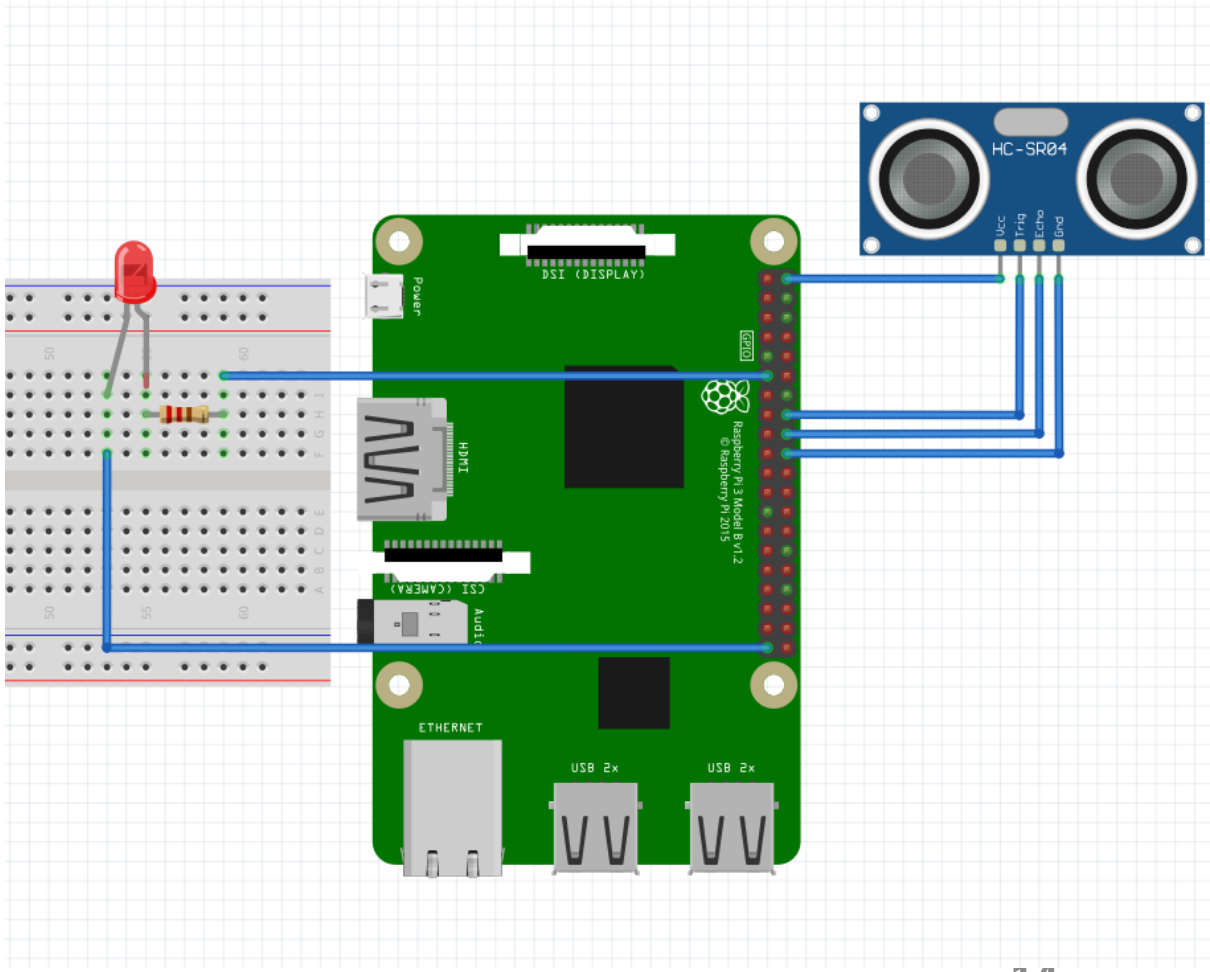
```
205.9704291820526
205.73919773101807
206.4004385471344
205.95825910568237
207.61744618415833
10.198523998260498
206.3355314731598
205.97854256629944
207.2401738166809
```

倒車雷達

實驗步驟(1/5)

- 將LED與樹梅派用杜邦線連接

- LED高電位：GPIO17
- 接地：腳位39



實驗步驟(2/5)

- 變數預設second=1,distance=1000

```
second = 1
distance = 1000
TRIG = 23
ECHO = 24
```

實驗步驟(3/5)

- 將腳位17設為GPIO 17做控制端輸出，距離小於30公分就會根據second的設定做閃爍(預設為1秒為間格)。

```
def job():
    while True:
        GPIO.setup(17, GPIO.OUT)
        GPIO.output(17,False)
        if distance <=30:
            time.sleep(second)
            GPIO.output(17,True)
            time.sleep(second)
```

實驗步驟(4/5)

- 因為距離感測與LED要同時進行，因此使用thread的方式讓他能同時運作

```
t =threading.Thread(target = job)

t.start()
```

實驗步驟(5/5)

- 將23及24腳位當作輸出以及輸入，即可顯示距離

```
while True:
    GPIO.setup(23, GPIO.OUT)
    GPIO.setup(24, GPIO.IN)
    print (get_distance())
    distance = get_distance()
```

課堂實作

- 讓LED能根據距離的遠近，改變閃爍的頻率
  - 距離小於30公分 閃爍頻率為0.2s
  - 距離小於20公分 閃爍頻率為0.1s
  - 距離小於10公分 閃爍頻率為0.05s