



微算機實驗報告

Lab #14

姓名：楊哲睿

系級：電機 10

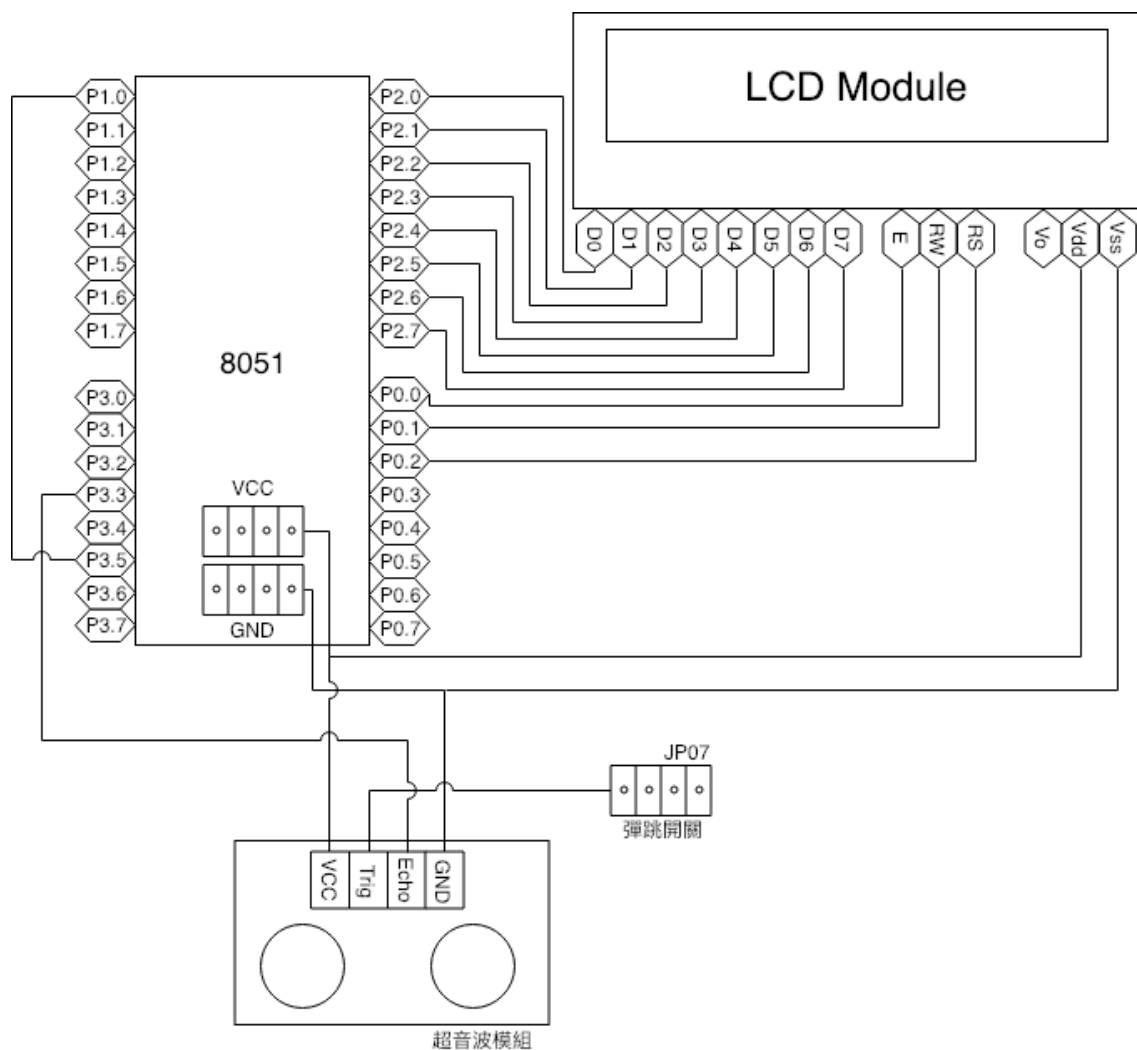
學號：0610780

上課時間：4EF、4IJ

一、實驗目的：

- 瞭解如何用 C 語言撰寫微控器程式。
- 瞭解 Timer/Counter 工作原理進行距離量測。
- 利用超音波模組實現測距的功能，將超音波反射的概念結合。

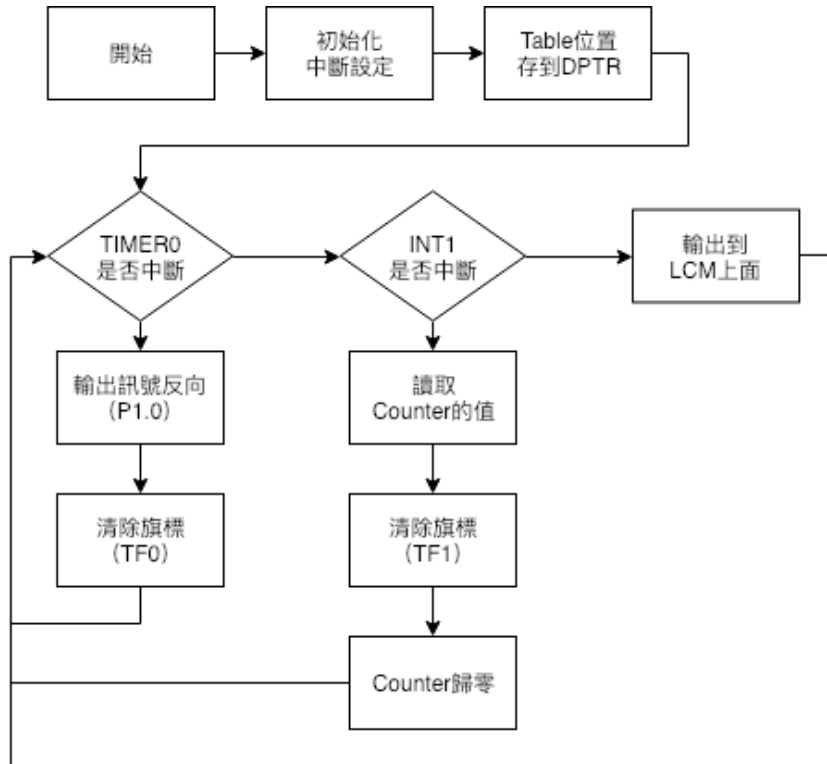
二、硬體架構：



三、程式流程圖：

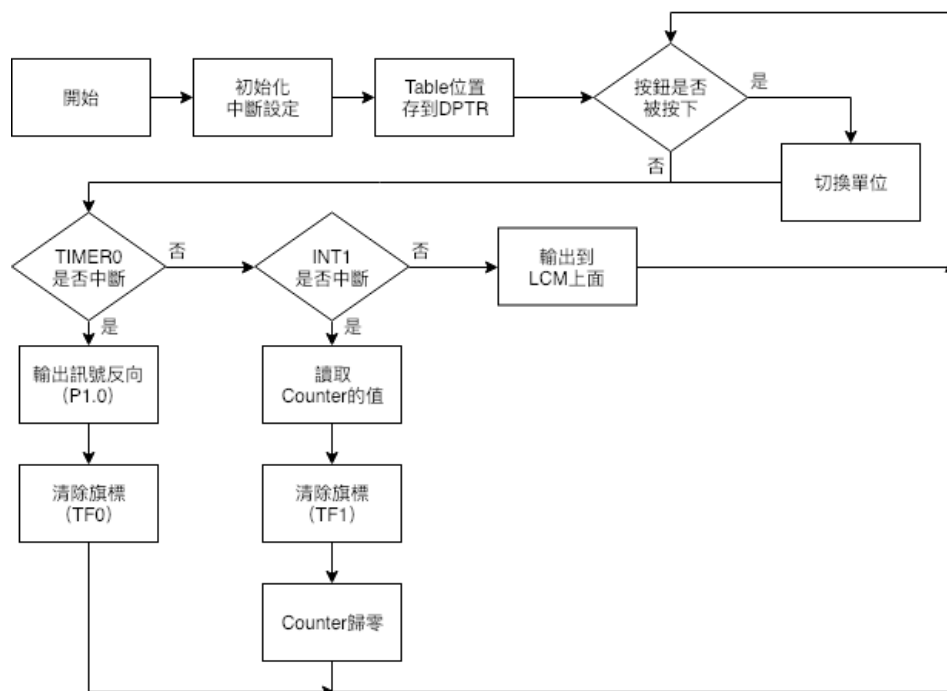
- 基本題

顯示以公分為單位的距離(要到小數點第一位)。



- 進階題

加入按鈕模組，當按下按鈕可以切換成以公尺為單位的距離(要到小數點第三位)。



四、問題與討論：

- 如何用 C 語言設定某腳位為 input-only？

實際上，沒有什麼方法可以將 8051 的 pin 設置為只能輸入或輸出專用；因為所有引腳都是雙向的。

但是，如果向一個腳位寫「0」，漏極開路驅動器（open-drain driver）會被設為 ON，任何外部的訊號都無法改變該腳位「0」的狀態。此時輸入只能讀到「0」。

但如果向一個腳位寫「1」，則會將漏極開路驅動器設為 OFF，因此外部信號可以將該腳位設置為高電位（1）或低電位（0），也就是說，此時輸入可以讀取外部信號的狀態。

所以說，當我們把 PORT 設為 0xFF 時，會讓外部資料可以傳送進 PORT，算是部分 input-only；反之如果不希望外部資料輸入，就可以先把 PORT 設為 0x00。

- 在使用 C 語言時，有 R0 到 R7 這些暫存器嗎？

8051 的指令集中，可以直接定址的有 R0 到 R7 一共 8 個暫存器，Keil C51 拿其中的 6 個暫存器來組成 3 個 16 位元的暫存器作為傳遞 C 的函數參數之用，因此函數的參數大於 3 個的部份就必需用「記憶體」而不能直接使用「暫存器」。

也就是說，R0 到 R7 不能像組合語言一樣直接使用，但是仍可以使用這些儲存空間，只要定義好儲存空間在記憶體中的位址，就可以使用這些暫存器。

- 如何使用 C 語言宣告變數，以及有哪些該注意的地方？

一般來說，就跟平時寫 C 語言一樣那樣宣告變數，只是變數宣告在程式中間時好像會有問題，所以在宣告變數時最好在程式的開頭或是函式的起始位置等地方宣告。

- 組語有 TABLE，那 C 語言該怎麼實踐？

可以用 C 語言裡面的 array 去實現組合語言中 TABLE 的功能。

五、程式碼與註解：

- 基本題

顯示以公分為單位的距離（要到小數點第一位）。

以下的 code 不是很完整，沒有完全跑出結果@@

```

#include <reg52.h>
#include <stdio.h>
#include <LCD_8_bit.h>
#include <math.h>

#define sound_velocity 34300      /* sound velocity in cm
per second */

#define period_in_us pow(10,-6)
#define Clock_period 1.085*period_in_us      /* period for
clock cycle of 8051*/

sbit Trigger_pin = P2^6;          /* Trigger pin */
sbit Echo_pin = P2^7;            /* Echo pin */

void Delay_us()
{
    TL0 = 0xF5;
    TH0 = 0xFF;
    TR0 = 1;
    while (TF0==0);
    TR0 = 0;
    TF0 = 0;
}

void init_timer(){
    TMOD = 0x01;
    /*initialize Timer*/
    TF0 = 0;
    TR0 = 0;
}

void send_trigger_pulse(){
    Trigger_pin = 1;              /* pull trigger pin HIGH
*/
    Delay_us();                   /* provide 10uS Delay*/
    Trigger_pin = 0;              /* pull trigger pin LOW*/
}

```

```

void main()
{
    float distance_measurement, value;
    unsigned char distance_in_cm[10];
    LCD_Init();          /* Initialize 16x2 LCD */
    LCD_String_xy(1, 1, "Distance:");
    init_timer();        /* Initialize Timer*/

    while(1)
    {
        send_trigger_pulse();    /* send trigger pulse
of 10us */

        while(!Echo_pin);        /* Waiting for
Echo */

        TR0 = 1;                /* Timer
Starts */

        while(Echo_pin && !TF0);  /* Waiting for
Echo goes LOW */

        TR0 = 0;                /* Stop the
timer */

        /* calculate distance using timer */
        value = Clock_period * sound_velocity;
        distance_measurement = (TL0 | (TH0 << 8)); /* read
timer register for time count */
        distance_measurement =
(distance_measurement*value) / 2.0; /* find distance(in
cm) */

        sprintf(distance_in_cm, "%.2f",
distance_measurement);
        LCD_String_xy(2,1,distance_in_cm); /* show
distance on 16x2 LCD */
        LCD_String("  cm  ");

        delay(100);
    }
}

```

```

    }
}

```

● 進階題

加入按鈕模組，當按下按鈕可以切換成以公尺為單位的距離(要到小數點第三位)。

以下的 code 不是很完整，沒有完全跑出結果@@

```

#include <reg52.h>

#define sound_velocity 34300    /* sound velocity in cm
per second */

#define period_in_us 0.000001
#define Clock_period 1.085*period_in_us    /* period for
clock cycle of 8051*/

sbit Trigger_pin = P3^6;        /* Trigger pin */
sbit Echo_pin = P3^7;          /* Echo pin */

#define LcdDataBus P2
sbit LCD_RS = P1^0;
sbit LCD_RW = P1^1;
sbit LCD_EN = P1^2;

/* local function to generate delay */
void delay_us(int cnt) {
    for(int i = 0; i < cnt; i++);
}

/* Function to send the command to LCD */
void Lcd_CmdWrite(char cmd) {
    LcdDataBus = cmd;    // Send the command to LCD
    LCD_RS = 0;          // Select the Command Register by
pulling RS LOW
    LCD_RW = 0;          // Select the Write Operation by
pulling RW LOW

```

```

        LCD_EN = 1;          // Send a High-to-Low Pusle at
Enable Pin
        delay_us(10);
        LCD_EN = 0;
        delay_us(1000);
    }

/* Function to send the Data to LCD */
void Lcd_DataWrite(char dat) {
    LcdDataBus = dat;  // Send the data to LCD
    LCD_RS = 1;        // Select the Data Register by
pulling RS HIGH
    LCD_RW = 0;        // Select the Write Operation by
pulling RW LOW
    LCD_EN = 1;        // Send a High-to-Low Pusle at
Enable Pin
    delay_us(10);
    LCD_EN = 0;
    delay_us(1000);
}

void delay(int k) {
    for(int i = 0; i < k; i++)
        for(int j = 0; j < 112; j++);
}

void Delay_us() {
    TL0 = 0xF5;
    TH0 = 0xFF;
    TR0 = 1;
    while(TF0 == 0);
    TR0 = 0;
    TF0 = 0;
}

void init_timer() {
    TMOD = 0x01;
    /*initialize Timer*/

```

```

    TF0 = 0;
    TR0 = 0;
}

void send_trigger_pulse() {
    Trigger_pin = 1;          /* pull trigger pin HIGH
*/
    Delay_us();               /* provide 10uS Delay*/
    Trigger_pin = 0;          /* pull trigger pin LOW*/
}

void main()
{
    float distance_measurement = 10;
    //unsigned char distance_in_cm[10];

    Lcd_CmdWrite(0x38);       // enable 5x7 mode for
chars
    Lcd_CmdWrite(0x0E);       // Display OFF, Cursor ON
    Lcd_CmdWrite(0x01);       // Clear Display
    Lcd_CmdWrite(0x80);       // Move the cursor to
beginning of first line

    init_timer();             /* Initialize Timer*/
    while(1) {
        float distance_measurement = 10, value;
        int temp, temp1, temp2, temp3, temp4, temp5;
        Lcd_CmdWrite(0xc0);

        send_trigger_pulse(); /* send trigger pulse
of 10us */

        while(!Echo_pin);     /* Waiting for
Echo */

        TR0 = 1;              /* Timer
Starts */

        while(Echo_pin && !TF0); /* Waiting for

```



```

Echo goes LOW */
        TR0 = 0;                                /* Stop the
timer */

        /* calculate distance using timer */
        value = Clock_period * sound_velocity;
        distance_measurement = (TL0|(TH0<<8)); /* read
timer register for time count */
        distance_measurement =
(distance_measurement*value)/2.0; /* find distance(in
cm) */
        distance_measurement = distance_measurement *
1000;

        temp = distance_measurement/10;
        temp1 = distance_measurement-temp*10;

        distance_measurement =distance_measurement/10;
        temp = distance_measurement/10;
        temp2 = distance_measurement-temp*10;

        distance_measurement =distance_measurement/10;
        temp = distance_measurement/10;
        temp3 = distance_measurement-temp*10;

        distance_measurement =distance_measurement/10;
        temp = distance_measurement/10;
        temp4 = distance_measurement-temp*10;

        distance_measurement =distance_measurement/10;
        temp = distance_measurement/10;
        temp5 = distance_measurement-temp*10;

        Lcd_DataWrite('0'+temp5);
        Lcd_DataWrite('0'+temp4);
        Lcd_DataWrite('.');
        Lcd_DataWrite('0'+temp3);
        Lcd_DataWrite('0'+temp2);
        Lcd_DataWrite('0'+temp1);

```

```
Lcd_DataWrite('c');  
Lcd_DataWrite('m');  
  
delay(100);  
}  
}
```

六、心得：

這次實驗真的世界無敵難的，我到現在還是不知道怎麼讓他動起來，程式碼看了好久好久，改了一堆地方都沒有用，幸好這已經是最後一次實驗了，希望之後如果還有機會的話，可以知道怎麼讓他正常的運作。感覺會用 C 語言控制 8051 也是一件很重要必須要會的事情，雖然這次上課沒弄懂，相信之後總有一天我會征服他的。