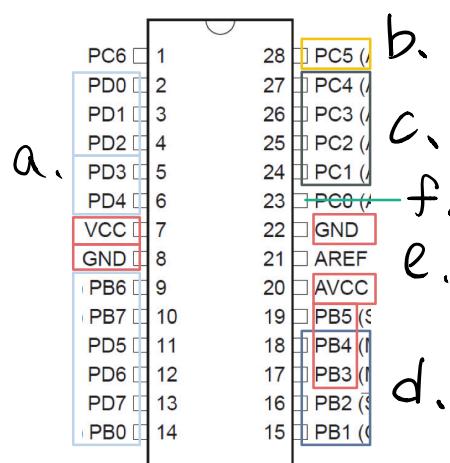
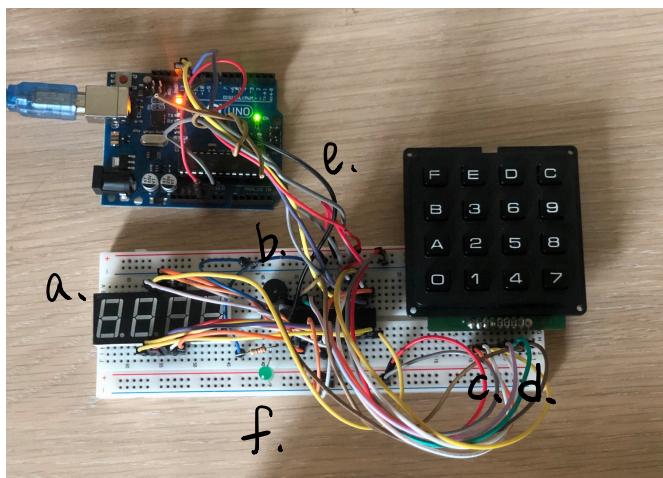


機電整合一 期末專題報告

第六組：陳柏霖、劉耀文



硬體設計說明：

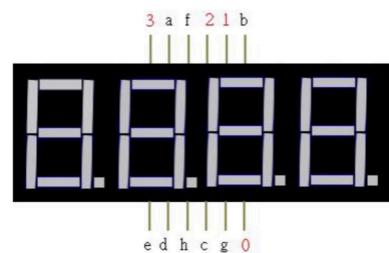


Part a (PD0-PD7 PB0 PB6PB7) :

接multiplexed 7-segment

其中我們利用快速切換PD0 PD1 PD2
來顯示後三位數字

PD3-PD7 PB0 PB6 PB7 用來調整7-
segment



Part b (PC5) :

接buzzer

從頻率及一些相關的數字推出n值，使用
timer以發出不同頻率的聲音



Part c (PC1-PC4) :

連接keypad其中四個腳位，判斷是否有按鍵被按下

Part d (PB1-PB4) :

連接keypad另外四個腳位，判斷是否有按鍵被按下

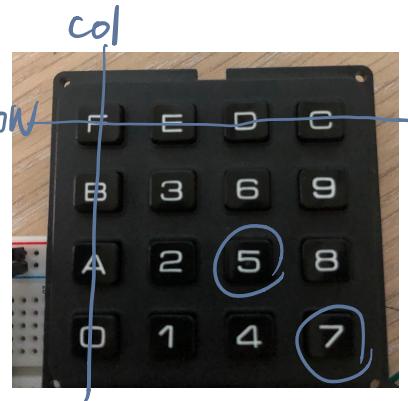
Part e :

連接ATmega328p和arduino，使arduino作為其programmer的
設定

Part f :

我們裝一顆LED在PC0的位置上，原本是拿來debug測試keypad
程式用的，後來留著作為指示燈，每分鐘整的時候會閃爍，在
piano mode時壓下聲音的同時也會亮。

我們組的按鍵分布除了「7」之外，主要是在row1和col1上，因此演算法我們先設定PB(col)為output、PC(row)為input，判斷「0」「A」「B」「F」；再設定PB(col)為input、PC(row)為output，判斷「E」「D」「C」；「7」則另外處理。



演算法說明：

我們最核心的函式是count(int digit0,int digit1,int digit2)

此函式多次使用含有timer的迴圈來記時間，每呼叫完一次即為一秒；我們想到它裡面的迴圈可以在一秒內跑很多次，剛好和multiplex 7-segment的概念相符，所以在每一次迴圈中加入show0(int digit0) show1(int digit1) show2(int digit2)函式，利用短暫的delay切換顯示每一個位數，達成邊記秒數邊顯示所在秒數的功能。

我們還利用noise(int n)函式發出聲音，n值是我們根據C3到A3的頻率換算的，裡面包含開關buzzer的迴圈，裡面也包含show0(int digit0) show1(int digit1)使multiplex顯示音階的功能。

我們的主函式首先以watch mode開始正數計時，進入的while迴圈主要結構如下：

```

while(1)
{
    count(digit0,digit1,digit2); // to count each second
    t++; // increasing value
    h--; // decreasing value

    if(mode==0) // piano mode
    {
        ...
    }
    else if(mode==1) // increasing watch mode
    {
        ...
    }
    else if(mode==2) // countdown mode
    {
        ...
    }
}

```

藉由判斷keypad按下按鍵的if()，來做換mode、顯示數字、發出聲音、亮燈的指令，由於if(mode==x),x=0,1,2裡面只有小到可以忽略的delay，在計數時幾乎不占時間，只有count()函式會計算時間；另外，t代表的是正數的時間，h代表的是倒數的時間，歸零只要在歸零鍵的指令下再次設定t,h的值即可。

至於硬體設計的keypad按鍵我們因為排列比較特殊，所以我們嘗試輪流用PC1-4 PB1-4當output，另一方當input，Bincout() 和BoutCin()是設定DDR設定的函式，裡面也有包括pull up。

Bin Cout							
	7	6	5	4	3	2	1
DDRB				0	0	0	0
DDRC							
DDRD							

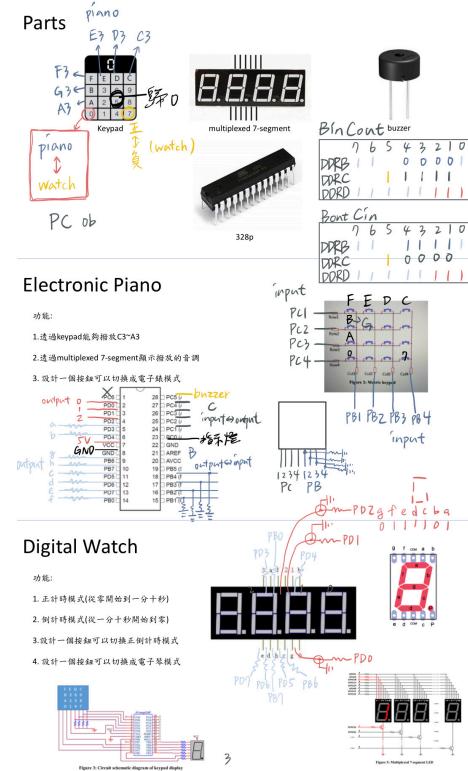
Bout Cin							
	7	6	5	4	3	2	1
DDRB							
DDRC				0	0	0	0
DDRD							

我們有額外做的部分是LED指示燈，在按下各個音調的時候都會亮，可以幫我們檢查它確實有按下去；在每個整數分鐘時(不管正

數倒數)會閃爍兩下，這可以幫我們檢查在piano mode的時候watch有沒有繼續運作。

製作過程：

右圖是我在規則講解投影片上做的草稿，這次會需要很多線，所以要先規劃好。我做到一半蠻後悔我一開始的按鈕設定和keypad連接的腳位，如果我早點想到程式的邏輯可能就會把按鈕設成橫向兩排，並且把keypad連的8條線都連到PD，會好寫很多，但我最後還是努力把原本的概念實現了。



心得：

一開始很慌張，因為會用到keypad和multiplexed 7-segment，這兩個零件都是我想做bonus時遇到的難題，當時因為覺得觀念方面和程式方面(不是陳世芳)都感覺很複雜，試一試覺得試不出來就不做了，沒想到期末專題又遇到了。但努力去想一陣子後，我覺得根本沒有我做lab的時候的感覺的難，反而會去欣賞這個零件的設計，例如multiplexed 7-segment就不用至少 $(7+1)\times 4$ 個腳，只要12個就行。pull up也是我一直過不去的坎，每次都要想

是0還是1，要不要加「～」符號想半天，但這次非常密集的訓練確實讓我把以前還不敢面對的東西都翻出來了，這是平常lab做不到的，幸虧有這次期末專題讓我把這學期應該要會的東西用壓力逼我都學到了！

很遺憾沒有辦法和原本想的一樣以自走車的競賽作為期末專題，原本以為期末專題改家裡做之後還是自走車，因此把電池充好了，連部分的線都接好了，才發現不是要做車而是電子琴和電子錶。看到題目時覺得lab有做過類似的，應該研究一下就做出來了，沒想到它比想像中的還要困難許多，因為要設定的腳位很多，常常一沒注意一個1或0，就不小心動到另一個地方的設定了，不僅考驗邏輯也很考驗細心。但是我認為這次專題比起自走車有更大的公平性，因為車子的狀況很多，如果又一不小心燒起來就完了，很難在家完成也蠻考驗運氣的；digital watch 和 electric piano 相較起來更加考驗程式的部分，雖然困難很多，但從中也可以成長不少！

