EE2004

Microcomputer System

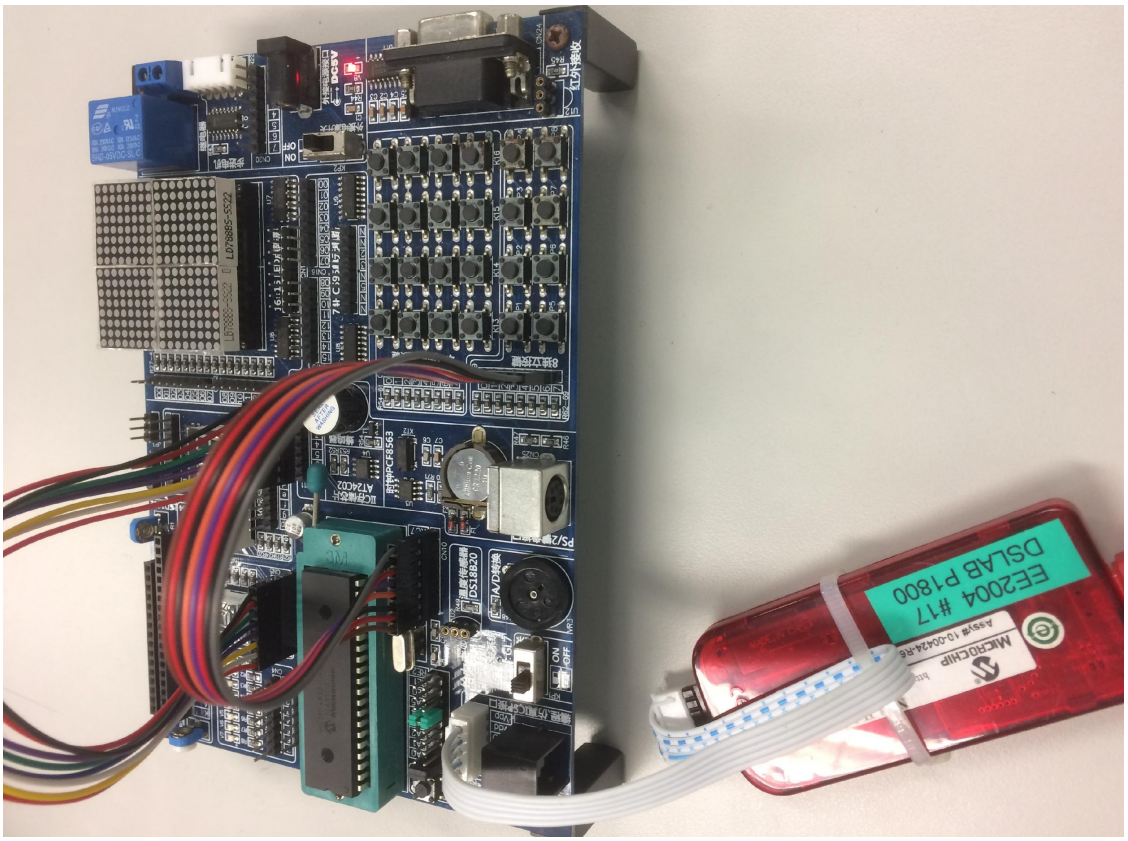
Mini Project
(Digital Piano)

Group member:

Wong Kin Ho, 54403088
Chan Man Hong, 54390457

Purpose

In this project, we are going to build a simple digital piano using the pic18F4520, through the buzzer (output) and the 8 independent button (input). We aims to create a simple piano which can produce Do Re Mi Fa So La Ti 7 sounds on the broad.



Problems Faced

-Produce Different Sounds

At the very beginning of our project, we encountered a serious matter. That is we did not know how to change the sounds of the buzzer. We only know how to activate the buzzer but with the same sound. We had tried to put different value on the output and tested it, but the result is the same. After we revise how a sound is produced, we figure out the main point is the frequency of the sound wave. When come to frequency, delay always is the relative things. Thus, it is maked sense that our program should produce specific frequency to the buzz input.

Design and Solution

After having brain storming on the problems , we figure that different frequency can produce different sounds. In this project, we are going to design a 7 key piano which is a C major in octave 4. C major is a basic major scale with the pitches C,D,E,F,G,A,B.According to below website, their corresponding frequency(Hz) are C4(261), D4(293), E4(329), F4(349), G4(391), A4(440), B4(493).

Back to the pic18, we are going to use timer0 to generate a frequency of the above pitches. In the main program, we decide to use the instruction BTFSS to perform the keyscan part.

The tmr0H and tmr0L are calculated by the following ways.
FFFF-XXXX+1= mc , XXXX represent tmr0H and tmr0L
1/(mc*1us*2)=frequency of the pitch

follow the above equation, we have

C4 262 Hz	f8b3
D4 293 Hz	f972
E4 329 Hz	fa14
F4 349 Hz	fa66
G4 391 Hz	fb03
A4 440 Hz	fb8f
B4 494 Hz	fc0b

http://www.liutaiomottola.com/formulae/freqtab.htm

```
Main:                   movlw   0x0F                            ;Set all Ports digital I/O
                                movwf   ADCON1
                                clrf    TRISD           ;Set PORTD to be output
                                clrf    PORTD
                                movlw   0xFF            ;Set PORTC to be input
                                movwf   TRISC
                                MOVLW 0X08              :no prescaler
                                MOVWF T0CON


MainLoop:               call    KeyScan
                                movwf   PAT_NUM
                                movff   PAT_NUM, WREG


                                ;call   Delay
                                bra             MainLoop
.
```

Set PORTD be the output, PORTC be the input.
TMR0 = ‘00001000’. no prescaler.

```
KeyScan:                        btfss   PORTC, 0
                                GOTO    C_NOTE
                                btfss   PORTC, 1
                                GOTO    D_NOTE
                                btfss   PORTC, 2
                                GOTO    E_NOTE
                                btfss   PORTC, 3
                                GOTO    F_NOTE
                                btfss   PORTC, 4
                                GOTO    G_NOTE
                                btfss   PORTC, 5
                                GOTO    A_NOTE
                                btfss   PORTC, 6
                                GOTO    B_NOTE
                                btfss   PORTC, 7
                                GOTO    UC_NOTE
                                MOVLW   0X5B
                                MOVWF   PORTD
```

keyscan: scan 8 pins of the PORTC, if the pin is low which means pressing the key, goto the timer part.

movlw 0x5b uses to stop the buzz when there is no input during the keyscan.

```
C_NOTE:         MOVLW 0XF8
                        MOVWF TMR0H
                        MOVLW 0XB3
                        MOVWF TMR0L
                        BCF INTCON,TMR0IF
                        COMF PORTD
                        BSF T0CON,TMR0ON
C_A:            BTFSS INTCON,TMR0IF
                        BRA C_A
                        BCF     T0CON,TMR0ON
                        RETURN
```

The above is one of the pitch(C4). This is a basic timer0. With different input tmr0H and tmr0L. The delay generated will be different. COMF instruction is used to toggle PORTD which is connected to the buzz in order to provide a wave pattern with the suitable period.

Test and rationale of the test

The technical we used to change the sound is Timer which can allow us to make changes of the output frequency at the buzzer.

At first, we just random to type in the value in timer. Because we do not know technique of timer will work or not. The first test is successful. The sound of buzzer are different than the very beginning. Therefore, we started to calculate the real value of the seven melody and write into timer.

The second test is to check if we can use several buttons input to activate the buzzr through looping and ‘btfss’. Its result is same as our goals which let the buzzer to product different sounds when we pass different buttons. However, because of our buzzer is not professional, couple of melody it generated are not precise, although the frequency we type in is correct.

In order to make it more like a real piano, we decided to modify the frequency of those problem melody. Our method is simple, we change the frequency a little each time and test it again and again to find out the suitable value of that melody for the buzzer. After many testing, the simple piano is nearly perfect.

Techniques that I learn

During this project, I have learnt several techniques. Using the instruction btfss to complete the keyscan part is the first element in this project. We use several btfss statement to check whether the 8-key are pressed within a loop, then we can perform actions that we want. Secondly, producing specific delay to generate sound period is one of the important part in this project. As mention before, piano pitches have specific frequency resulted to different period. In order to produce those sound, we use timer0 in our project. Timer0 can generate delay by simply input the value of tmr0H and tmr0L. The method that how to calculate these two value we had mentioned before.

To conclude, using btfss to perform keyscan and timer0 to generate suitable pitches are the main techniques that I have learnt.