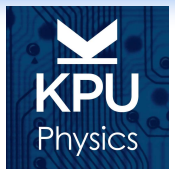


Project: Keypad Music

Author: Kouji Pan

Kwantlen Polytechnic University, Dept. of Physics
8771 Lansdowne Rd, Richmond, BC V6X 3X7



Introduction

MCU(micro controller unit)

MCU stands for microcontroller unit and is simply a computer designed to work with other machines like a laptop. We create the circuits that will use with the MCU on a “breadboard”. On the external computers, using a program called the MPLAB X IDE, we create and test the programs that run on the MCU. The program is written using the C language, and when we finished coding, the code will be transferred to the MCU using the USB connection.

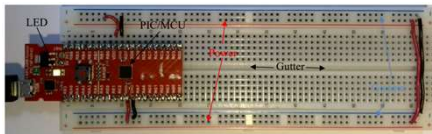


Figure.1 MCU in MPLAB Xpress Board on breadboard

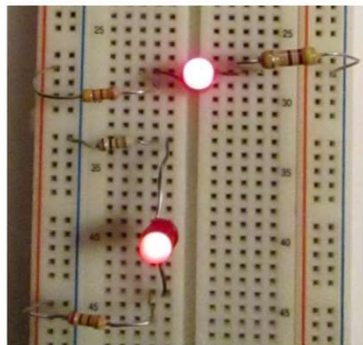


Figure.2 Series circuits

Materials and methods

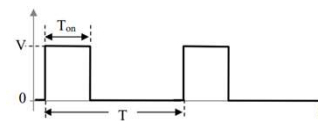
For this project, I used

- Keypad Device
- Resistors
- Wires
- MCU with breadboard
- MPLAB XIDE software

PWM (Pulse-Width-Modulation)

PWM stands for Pulse-Width-Modulation and is one kind of control signal that we can modulate pulse-width. Pulse-width is also called duty cycle that is the ratio of T-on to the period T shown in Graph.1. Moreover, we have a timer to decide the period T and frequency, so using PWM and timer, we can control the pulse emitted from the output.

The place of output can be decided by “pin manager” in MPLAB.



Graph.1 square wave signal

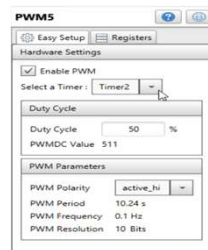


Figure.3 Settings for PWM

Module	Function	Direction	Port B	Port C	Port D
PWM5	PWM5 output	Output	7	0	1
PWM7	PWM7 output	Output	3	5	6
PWM7	GPI	Input	7	0	1
PWM7	GPO	Output	3	5	6

Figure.4. Settings of pin manager in MPLAB

Keypad

Keypad has 12 buttons and 7 connectors, and one connector on the keypad for each row and column as described in Figure.4. If we connect the one of the column to the power, the button will get the voltage when we press the any buttons for that column while buttons of other columns will get zero voltage. To use the keypad with the MCU, we connect the column connectors to three digital output pins, which enables us to decide which column of the keypad is active at any given time.

Figure.5 Keypad and connector block

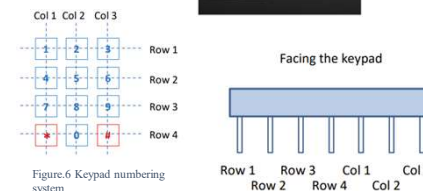


Figure.6 Keypad numbering system

Code

To make the only one button to be active when we press the button, I use the function “_SetHigh()” and “_SetLow()” to control the output from the pin shown in Figure.9.

```
while (1)
{
    key = 12; // no key press
    column1R06_SetHigh(0); //column 1 in powered up
    column2R05_SetLow(0);
    column3R04_SetLow(0);
    PWM5_LoadDutyValue(0);

    while(row1_R06_GetValue()){
        printf("You pressed 1.\n");
        PWM5_LoadDutyValue(477); // change OCPR1H:OCPR1L
        T2CONbits.ONPS = 2;
        T2PR = 0x0E; // change PWM5DCH:PWM5DCL
        n = T2CONbits.ONPS; // prescaler setting, N = 2^n
        row1_R06_SetValue(0);
    }

    while(row2_R05_GetValue()){
        printf("You pressed 4.\n");
        PWM5_LoadDutyValue(425); // change OCPR1H:OCPR1L
        T2CONbits.ONPS = 2;
        T2PR = 0x04; // change PWM5DCH:PWM5DCL
        n = T2CONbits.ONPS; // prescaler setting, N = 2^n
        row2_R05_SetValue(0);
    }
}
```

Figure.8 Partial code for Keypad Music Project

RESULTS

The final setting of breadboard is shown in Figure.10. I connect the speaker to the output pin for PWM so that speaker can detect the frequency and make different tones of sounds depending on the button press. We also need to connect digital input pins to ground with 10 kΩ resistors so that we don't have a bad reading accidentally because of the random change of electric field.

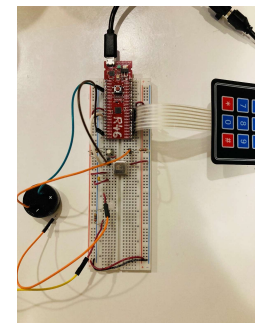


Figure.9 Device setting of Keypad Music Project

CONCLUSION

From this project, I could learn the concepts of PWM and Keypad deeply. Also, by knowing how the devices that I used work, I could easily control and fix the project. Therefore, before trying to solve the problems, we need to take a look at concept or background of problems, which can help us to clear the brain and tackle the problems efficiently.

CONTACT INFORMATION

Kouji Pan: kouji.pan@email.kpu.ca