

4.1 MPLAB Integrated Development Environment (IDE)

MPLAB Integrated Development Environment (IDE) is a free, integrated toolset for the development of embedded applications employing Microchip's PIC microcontrollers. It is easy to use and includes a host of free software components for fast application development and super-charged debugging. MPLAB support both Assembly Language and C programming languages, others language may be supported through the use of third party programs.

4.2 Process Flow Chart

4.2.1 MPLAB IDE

The MPLAB IDE is a windows operating system (OS) software program that runs on a PC to develop applications for Microchip microcontrollers and digital signal controllers. It is called an IDE (Integrated Development Environment) because it provides a single integrated "environment" to develop code for embedded microcontrollers.

4.2.2 MPLAB Configuration

Before starting development, you need to select the device and decide which development tools to use. There are a number of processors available supported. In this project, the PIC18F4520 is used.

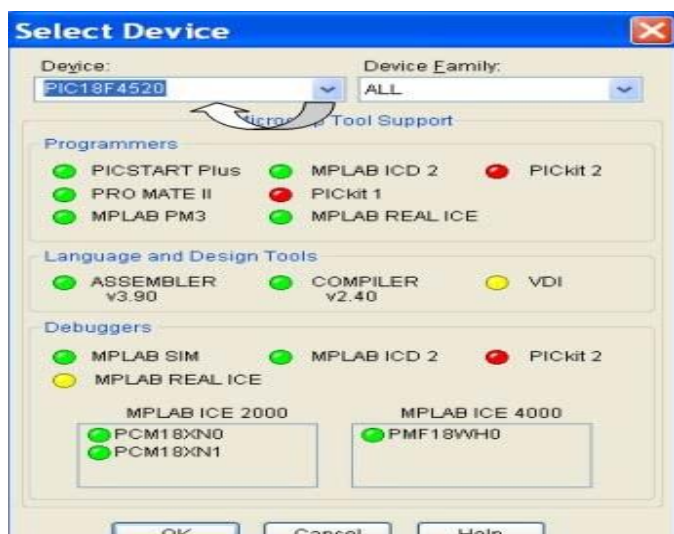


Figure 12: Selecting the device

4.3 Overview of the Program

The flowchart below illustrates the whole process of the program.

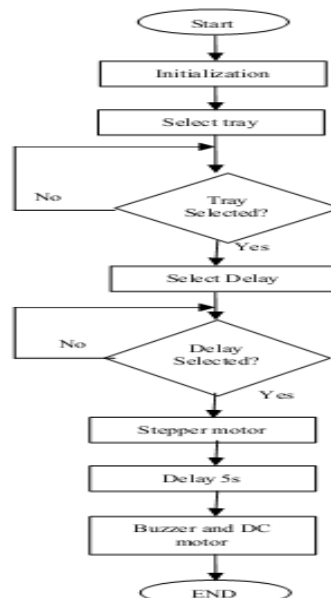


Figure 13: Flowchart of the Pet feeder

4.4 Software Programming

4.4.1 Motor

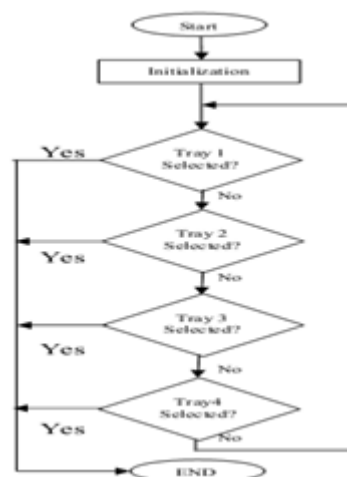


Figure 14: Flowchart of Motor

4.4.2 Timer

The other important program in this project is the timer program. It is used to set the time before the motor begins to rotate. In this program, I have only used the 3 delay options mainly 5s, 10s and 15s to act as the timer for this project.

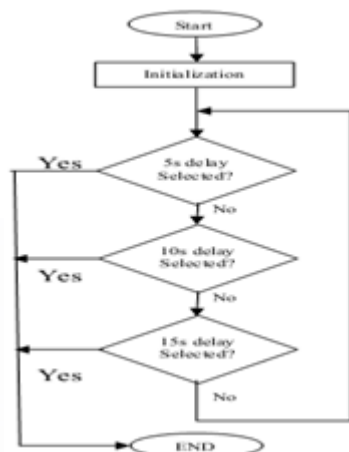


Figure 15: Flowchart of timer

4.4.3 Buzzer and DC Motor

The aim of the dc motor is to dispense food from the turn-table and the buzzer is to act as an alarm to notify pets that it is time for their meals. The program written for these 2 modules are explained together as they are similar. Both the buzzer and dc motor will always be in the waiting mode. They will only be activated once the turn-table is in position. Both the buzzer and the dc motor will then be on for 5 seconds before it turns off and back to the waiting mode.

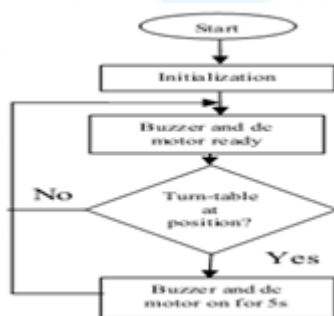


Figure 16: Flowchart of buzzer and dc motor

4.5 Source Code

```

/*****
****/
#include <p18f4520.h>
#include <delays.h>
/* macro for LCD */
#define Lcd_VEE LATDbits.LATD7 // VEE - lcd contrast
#define Lcd_EN LATDbits.LATD6 // Enable signal
#define Lcd_RW LATDbits.LATD5 // read write signal
#define Lcd_RS LATDbits.LATD4 // register select signal
/* function declaration */ void Lcd_Title (void);
Voidmotor (void);
Void Write_Lcd (unsigned char, unsignedchar); void
DelayFor18TCY (void); /* global variables */ unsigned
char count; int Tray; int Delay;
Voidmain (void)
{
/* configure port */
ADCON1 = 0x0F; // default all pins to digital

```

```

/* configure LCD pin */
TRISDbits.TRISD7 = 0; // VEE // PICDEM2 PLUS
GREEN BOARD
TRISDbits.TRISD6 = 0; // EN
TRISDbits.TRISD5 = 0; // RW
TRISDbits.TRISD4 = 0; // RS
TRISDbits.TRISD0 = 0; // DB4
TRISDbits.TRISD1 = 0; // DB5
TRISDbits.TRISD2 = 0; // DB6
TRISDbits.TRISD3 = 0; // DB7
TRISA = 0xf0;
TRISB = 0x01; //Portpins RB1-RB3 as outputs TRISC
= 0x00;
Lcd_VEE = 1;
Delay10KTCYx(15);
Write_Lcd (0x20,0);
Write_Lcd (0x28,0);
Write_Lcd (0x0c,0);
Write_Lcd (0x01,0);

```

5. Conclusion

This project has been a very rewarding experience for me. As I am not working in the semi-con industry, this project has given me a chance to get more exposure in the field of electronics. As this project consists of both hardware and software, I was given the opportunity to get some hands-on and design experience on the hardware part. In addition, the development of software algorithm also allows me to incorporate the reasoning and logical skills.

Throughout the development phase of this project, many aspects of the electronic, mechanical and programming have been touched on and it has been an interesting. Most of the time is spent on the software portion. The most challenging issues that I faced are writing and debugging of software algorithm codes. Many areas need to be studied and considered thoroughly as I am not familiar with microcontroller and the programming language.

Overall, I have gained a lot of knowledge and experience in this project due to its broad engineering nature in areas of hardware and software design. I have learned to source for solutions when problems are encountered. This project has greatly improved my project management skills. Finally I believed what I have learned throughout the course especially in this module will greatly help me in my future career.

References

Internet:

1. [http //www.perfectpetfeeder.com](http://www.perfectpetfeeder.com) Accessed on 20 Feb 2009
2. <http://www.nextag.com/automatic-dog-feeder/search.html>
3. <http://en.wikipedia.org/wiki/Microcontrollers>
4. <http://www.zilog.com/application/application.asp?appid=1>

Books:

1. Shanefield D.J, Industrial Electronics for Engineers, Chemists, and Technicians, William Andrew Publishing, Norwich, NY, 2001
2. Schaum's Outline of Programming with C++ (Paperback) by John Hubbard publishing 2005
3. Programmable logic controllers 3rd edition by Gary Dunning Publishing 2006



IJSER