VANIER COLLEGE – Computer Engineering Technology – Autumn 2017

Introduction to Microprocessors (247-302)

Lab 1

Introduction to PIC uC and MPLAB X IDE

NOTE:
To be completed in ONE lab sessions of 3 hrs.
One report has to be submitted **not later than one week** after the last lab session.
This exercise is to be done **individually** except where specified in the procedure. **Each** student must submit a lab report with original design, observations and conclusions.

OBJECTIVES:

After completing the project, the student will be able to:

1. Prototype a basic PIC microcontroller circuit, with PICKIT3 programming and debugging capability.
2. Create a basic project using MPLAB X IDE.
3. Build codes and load to the prototyped board via PICKIT3.

THEORY:

A microcontroller is an inexpensive single-chip computer. The microcontroller's most important feature is its capabilities of STORING and RUNNING a program. It emphasizes self-sufficiency and cost-effectiveness, in contrast to a general-purpose microprocessor (such as those used in a PC).

Every Microcontroller (also MCU) consists of several major units:

- Input / Output Ports
- Control Pins: reset, power, clock
- Processor (CPU)
- Memory (RAM, ROM, EEPROM)
- Serial and parallel ports
- Timers
- Analog-to-digital (A/D) and digital-to-analog (D/A) converters

When a PIC is described as being 8- or 16-bit, this refers to the width of the data memory (registers) and ALU (arithmetic and logic unit).

The low-end PICs, which operate on data 8-bit at a time, are divided into four architectural families:

– **Baseline** (12-bit instructions)

  These PICs are based on original PIC architecture, back to the 70s. They are limited, but simple to work with.

  Modern examples include the 6-pin 10F200, 8-pin 12f509 and 14-pin 16F506.

– **Mid-range** (14-bit instructions)

  This is an extension of baseline architecture, adding support for interrupts, more memory and peripherals, including PWM, serial interface and LCD controllers.

  Modern examples include the 8-pin 12F629, 20-pin 16F690 and 40-pin 16F887

– **Enhanced mid-range** (14-bit instructions)

  Similar to mid-range, with additional instructions, simplified memory access (optimized for C compilers), and more memory, peripherals and speed.

  Examples are all the 12F1xxx and 16F1xxx series.

– **High-end** (16-bit instructions)

  Otherwise known as the 18F series, this architecture overcomes some of the limitations of the mid-range devices, provide more memory and advanced peripherals, including USB and Ethernet.

PIC Programmers and Debuggers

During the early period of PICs devices, they could only be erased by shining UV light through a window on the chip (except for parts without a window, which could only be programmed once), and programmed by placing them into a special programmer.

These days, PICs use electrically erasable flash memory. They can be programmed without having to be taken out of the prototyping environment, through a protocol called In-Circuit Serial Programming (ICSP). But instead of worrying about designing your circuit to accommodate the ICSP protocol, it can easier to use a programming adapter, which is simply a minimal circuit that allows a PIC to be programmed by an ICSP programmer.

Microchip's PICkit 3 In-Circuit Debugger/Programmer uses in-circuit debugging logic incorporated into each chip with Flash memory to provide a low-cost hardware debugger and programmer. The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface. The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

Development Software

MPLAB® X IDE (Integrated Development Environment) is the latest free software program that runs on a PC (Windows®, Mac OS®, Linux®) to develop applications for Microchip microcontrollers and digital signal controllers. It is called an IDE because it provides a single integrated "environment" to develop code for embedded microcontrollers.

The MPLAB® X IDE includes Microchip's assembler (MPASM), an editor, and a software simulator, which allows you to debug your application before committing it to the chip. MPLAB directly supports the PICKit3 as a programmer for PIC microcontrollers.

For more details about MPLAB® X IDE, you may download a copy of user guide from
http://ww1.microchip.com/downloads/en/DeviceDoc/52027B.pdf

Both MPLAB® X IDE and MPASM can be downloaded from www.microchip.com, and they are free.

Microcontroller and oscillators

Microcontrollers need oscillator (an oscillator is a device that oscillates with a repeatable variation, with respect to time) to function. This is because the hardware inside the microcontroller is based on sequential logic, such as flip-flops. Without an oscillator, or probably more accurately a clock, a microcontroller cannot execute program instructions, perform timers function etc. Hence it is very important to understand the nature of oscillators and timing settings for your microcontroller.

The PIC16 series of microcontrollers are advanced devices where you can either use an external oscillator or their powerful internal oscillator.

The labs of this course will be based on 8-bit PIC microcontroller, MPLAB X IDE tool and PICKIT3 debugger. Microchip's PICkit 3 In-Circuit Debugger/Programmer uses in-circuit debugging logic incorporated into each chip with Flash memory to provide a low-cost hardware debugger and programmer.

Lab procedures

PART A : Prototyping PIC microcontroller with PICKIT3

1. Pre-lab work. Show your work to teacher before proceed to step 2.

   You are required to design a basic prototype board based on the PIC16F887 microcontroller. Study the details of the pin out and functionality of the microcontroller and draw a schematic with the following basic features:
   a) Ensure all power lines are connected.
   b) Contains an external switch to reset the microcontroller.
   c) Ready to be interfaced with PICKIT3 (study and reseach pinout and functionality of PICKIT 3 is required)
   d) Contain a power-up LED.

2. Once the schematic design is approved, you will be provided with necessary components to prototype the design. Your bread boarding will be marked based on the correct use of colour coded cables, tidiness and functionality.

3. Once you have finished breadboarding, show your work to the teacher. Your board will then be verified by downloading a simple code.

4. In your report, include your schematic design, and discuss/describe any technical difficulties or knowledge obtained during the process of prototyping.

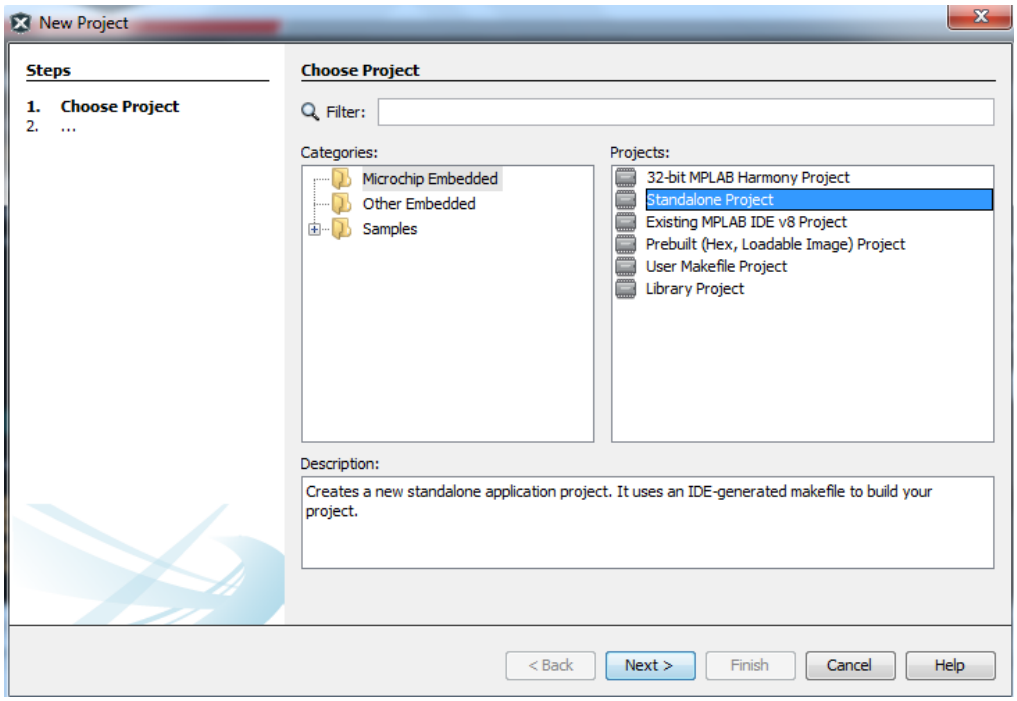
PART B: Basic flow of MPLAB X IDE

Note : *This lab is based on your PIC16F887 prototype board built in part A. Make sure your circuit is correctly designed and connected before you proceed with this lab. Show your prototyped board to your teacher.*

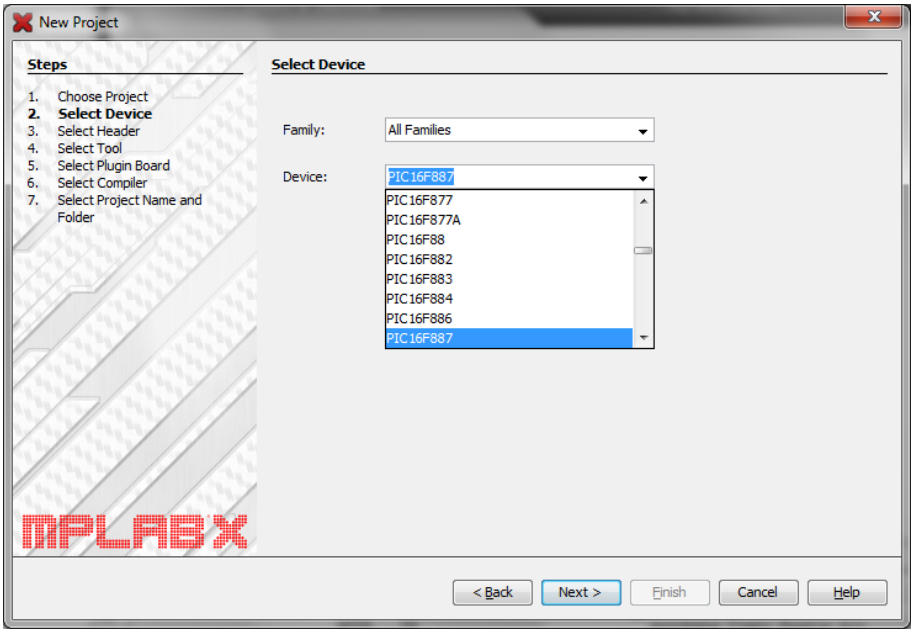
5. Run **MPLAB X IDE v3.61** on your PC.

6. Start the new project wizard by selecting **Create New** at Start Page as shown below, or from **File** menu.

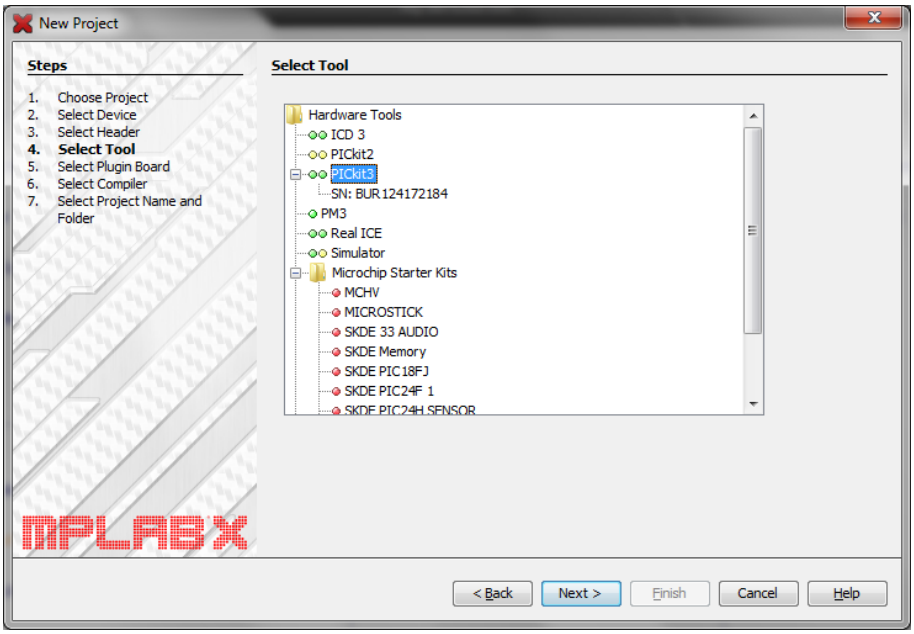
7. Choose the project type :
    a) From the **Categories**: choose **Microchip Embedded**
    b) On the **Projects** column: choose **Standalone Project**



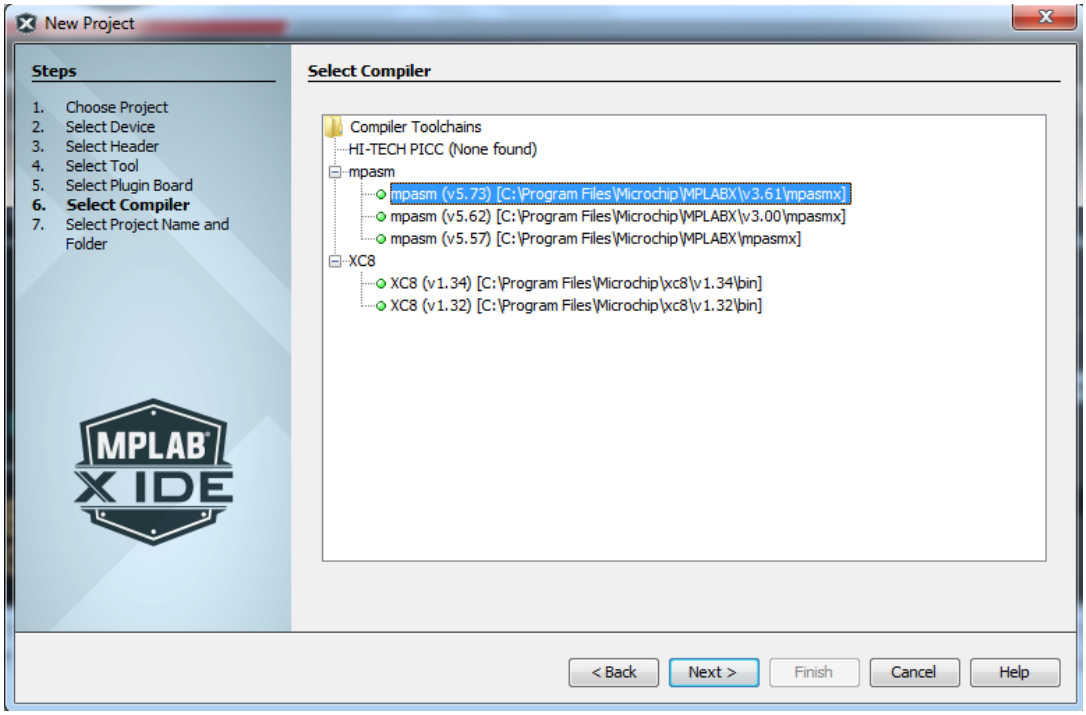
8. Choose the type of device by typing **PIC16F887** in the device selection.



9. Select the tool by choosing **PICKit3** under Hardware Tools.



10. Select the compiler. For this course, we use assembler **mpasm**. Choose the latest version available on your machine.

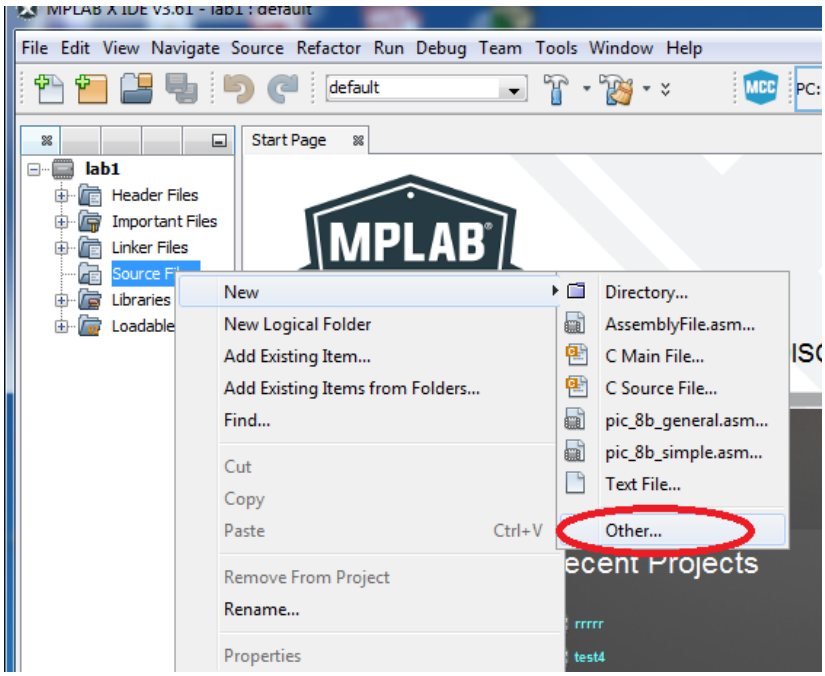


11. Finally name the project and choose a directory. Click finish to complete the new project wizard.

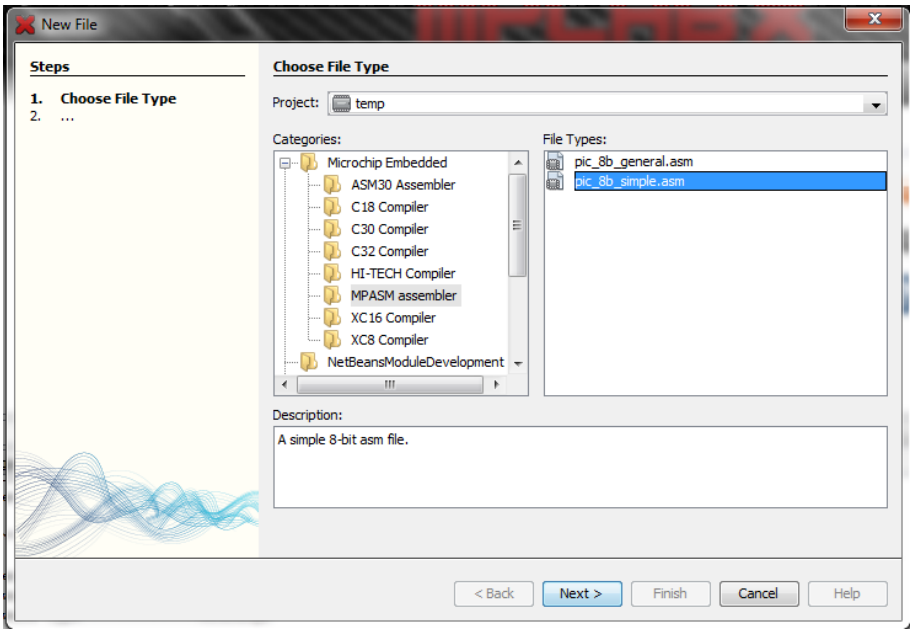
12. Next step, create a basic source code for the project.

    a) A project tree will appear in the upper left part of the IDE. You are now ready to add/create the source files to it.

    b) Right click on the **Source Files ► New ► Other...**

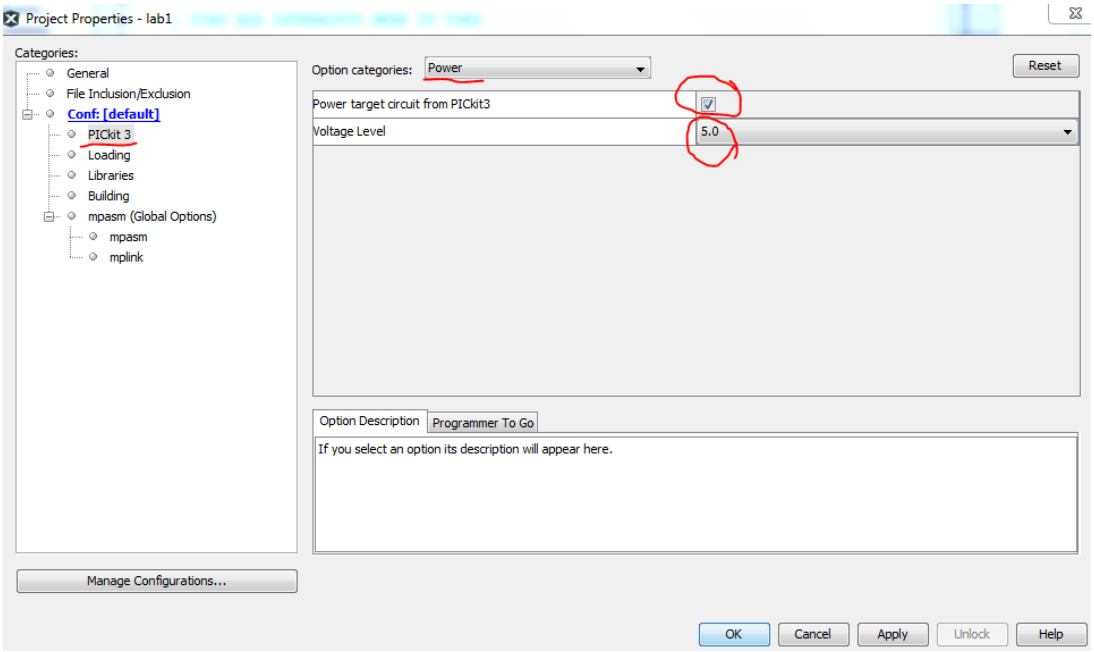


13. Under **Microchip Embedded**, choose **MPASM assembler**. Select **pic_8b_simple.asm** as a basic template for the source code. Type in your filename to finish the creation of simple template source code in assembly code. Press finish.



14. After the basic flow of creating project and source code for the microcontroller, you are now required to setup the PICKit3 to supply power to your prototyped board.

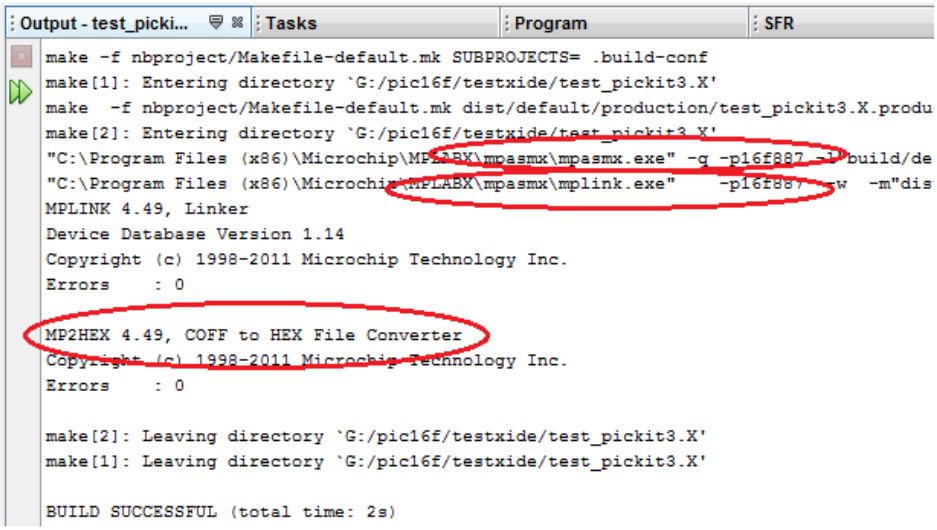
    a) Select **File ► Project properties**. Select **PICKit3** under **Conf**, and change the **Option categories** to **Power**. Select the option to **Power target circuit from PICKit3** and make sure your default Voltage Level is set at 5.0.



    b) Click Apply.

15. Build the codes by clicking the  icon on the toolbar.

    a) Explain the function of mpasm and mplink.



b) What is the purpose of mp2hex, and what are the differences between COFF and HEX file?

16. Load the program to your device.
    a) Connect the PICKit3 (with the device connected) to the computer.
    b) Press  icon to load the program to the device
    c) You should now see that the power LED on your prototyped board is lighted up via supplies from PICKit3.

17. Demonstrate your board with successful loading to your teacher, and get approval.