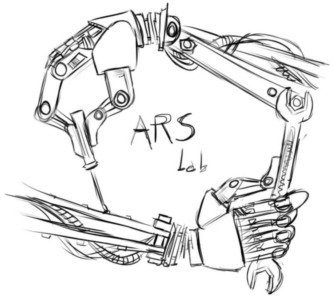
Setting Up the Environment for the Ready-for-PIC-28 Board

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory
Dipartimento di Matematica e Informatica - Università di Catania, Italy
santoro@dmi.unict.it



L.A.P. 1 Course

The “Ready-for-PIC-28” Evaluation Board

The Ready-for-PIC-28 is an evaluation board which includes:

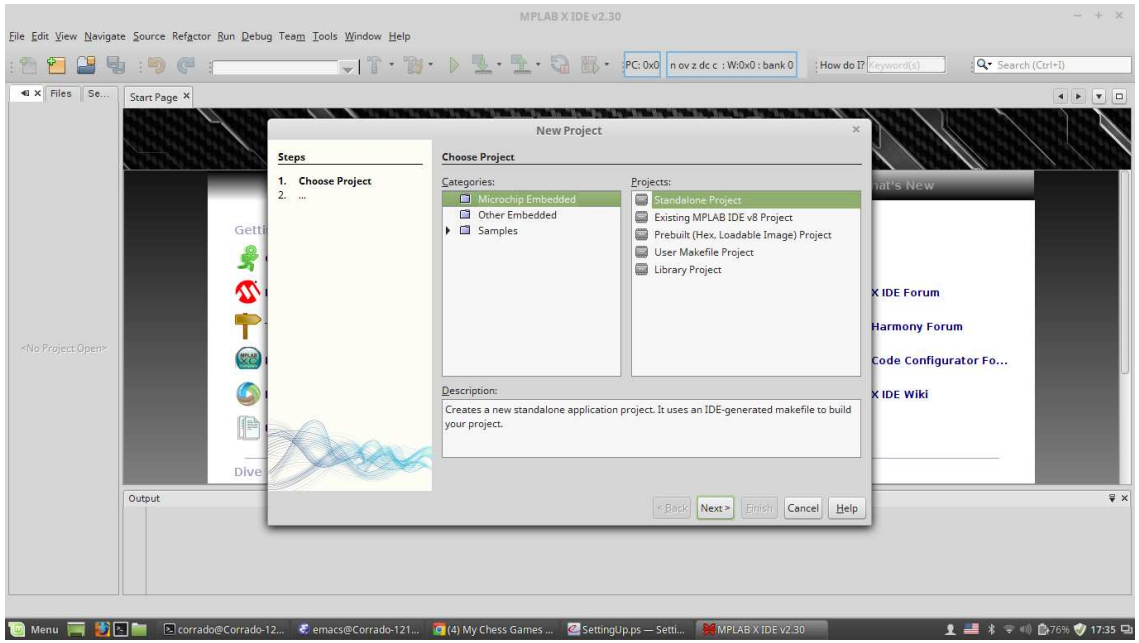
- A **PIC18F25K22** microcontroller
- A **power** circuit
- A **UART-USB** interface
- A **prototyping area** to implement simple test circuits

Resources needed to work with the board

- Microchip MPLABX IDE:
  `http://www.microchip.com/mplabx`
- Microchip C compiler for 8-bit MCU, XC8:
  `http://www.microchip.com/mplabx`
- A **terminal emulator** program, such as:
  - **minicom** or **cutecom**, for Linux;
  - **ZOC Terminal**, for Win and MacOS.
- The “Data Sheet” of the MCU **PIC18F25K22**:
  `http://www.microchip.com/`

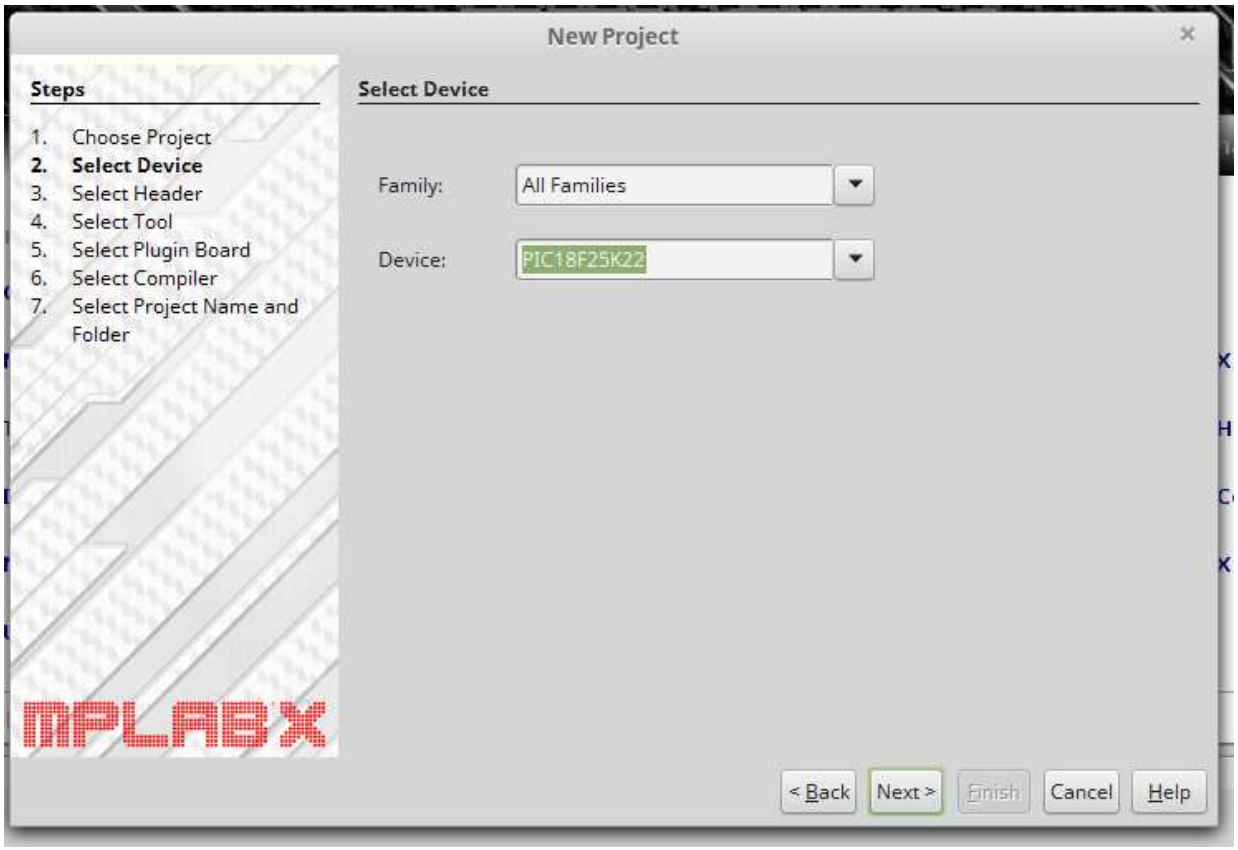
Creating a Project using the MPLABX environment (1)

- Open MPLABX
- Select File → New Project
- A wizard window will appear
- Choose Microchip Embedded, Stand alone project, and click Next



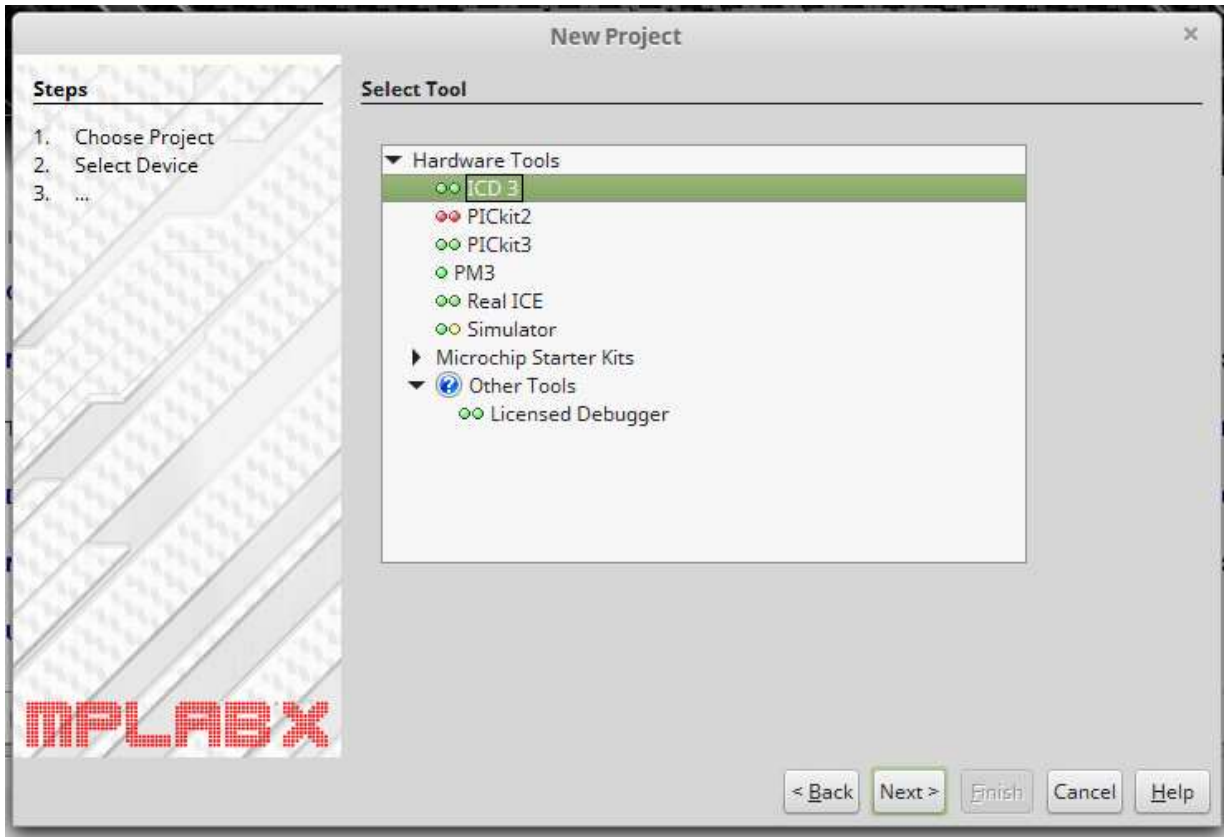
Creating a Project using the MPLABX environment (2)

- The wizard will open the device selection window
- Click into Device and type PIC18F25K22 (our microcontroller)
- Then click Next



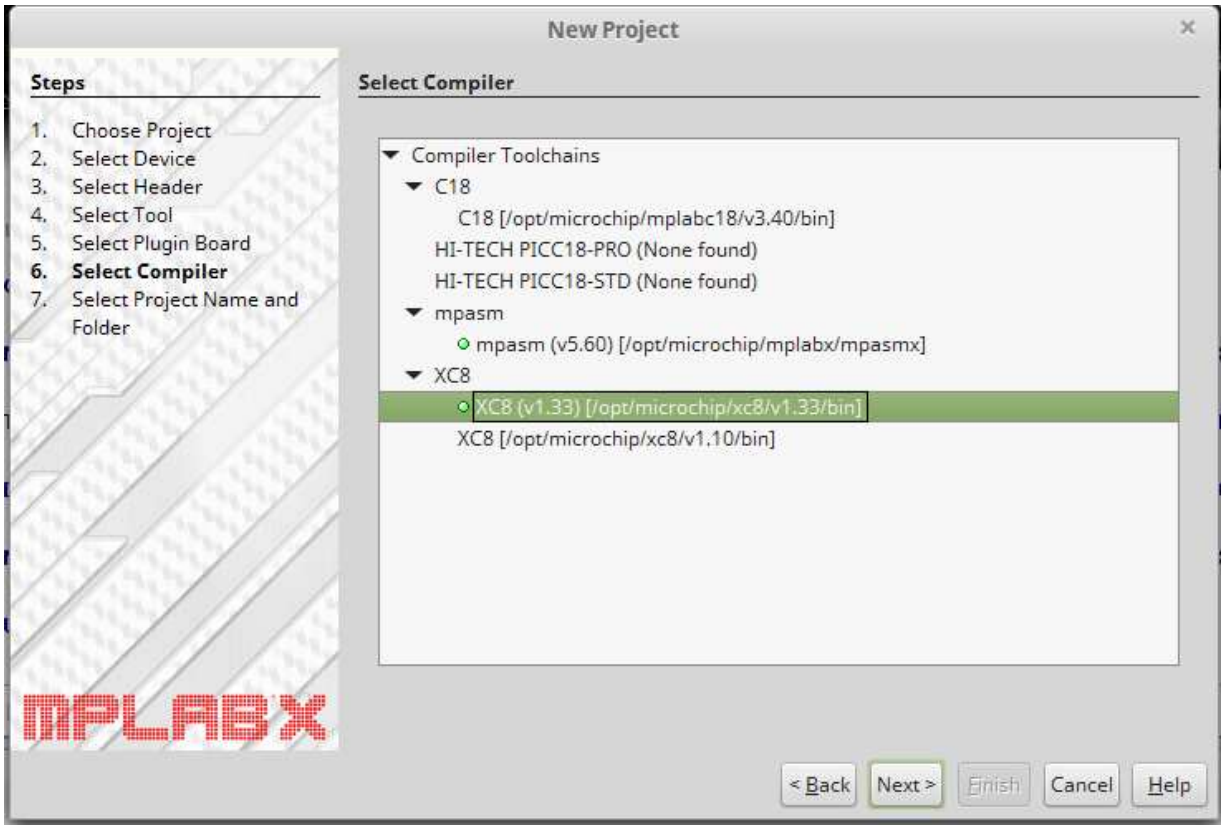
Creating a Project using the MPLABX environment (3)

- Next window will let you to select hardware tool used to program the device: we won't use a particular hardware tool thus this selection is not critical
- Make any selection (the first is ok), then click Next



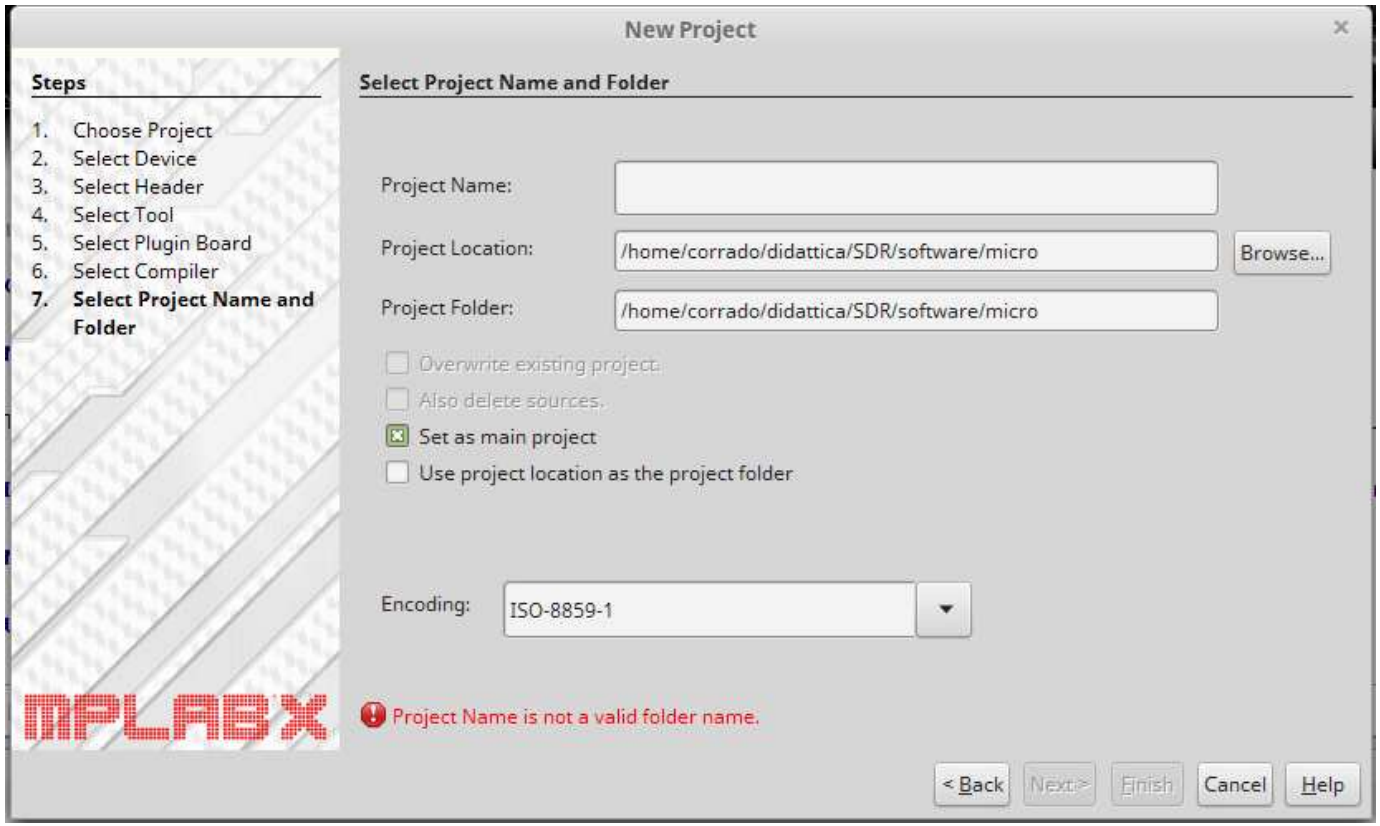
Creating a Project using the MPLABX environment (4)

- Next window will let you to select the compiler to use
- Since our MCU is an 8-bit, select XC8, then click Next



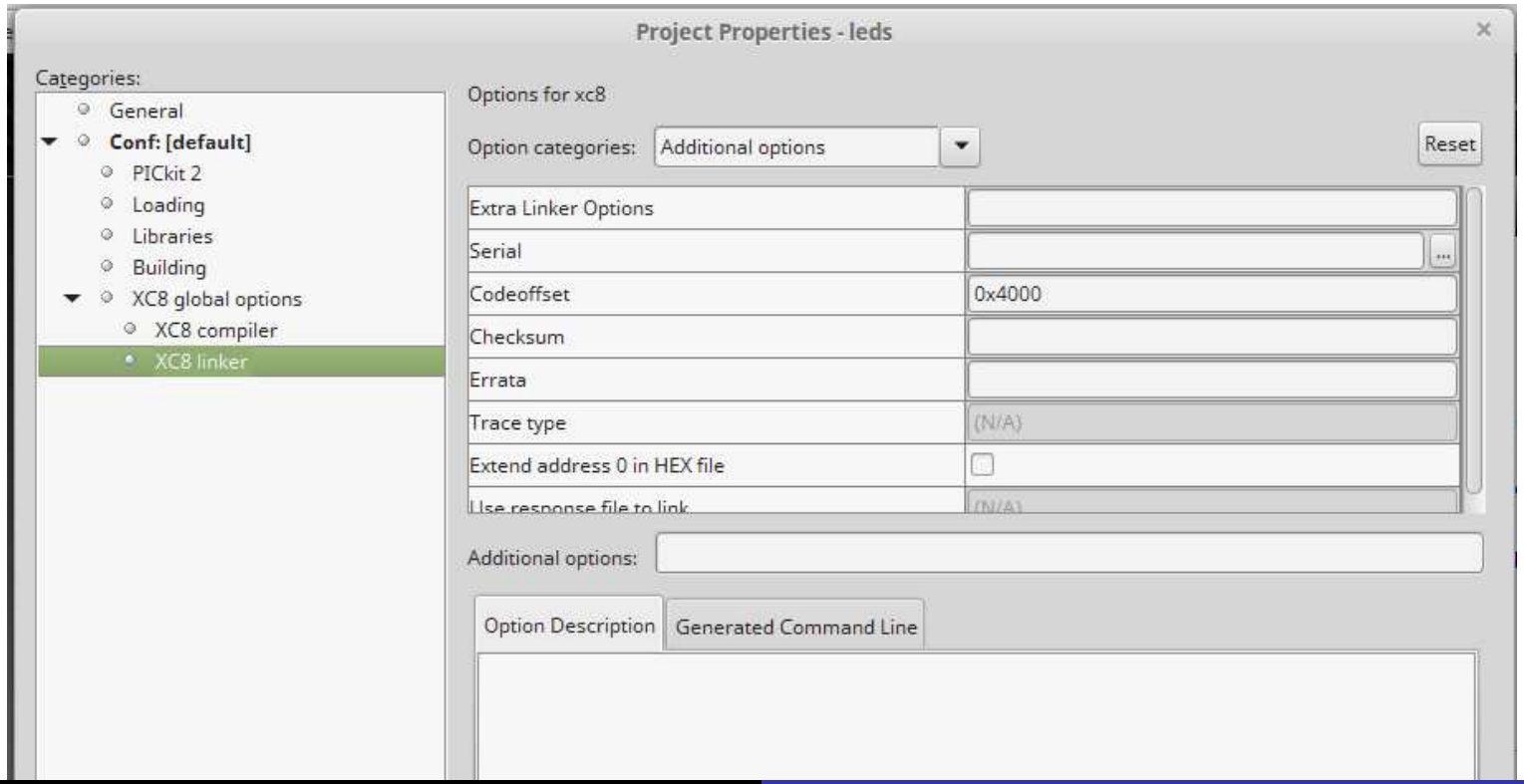
Creating a Project using the MPLABX environment (5)

- Next window will let you to select the name of your project and the folder which will contain the project files
- Insert proper values and click Finish, your project will be created shortly



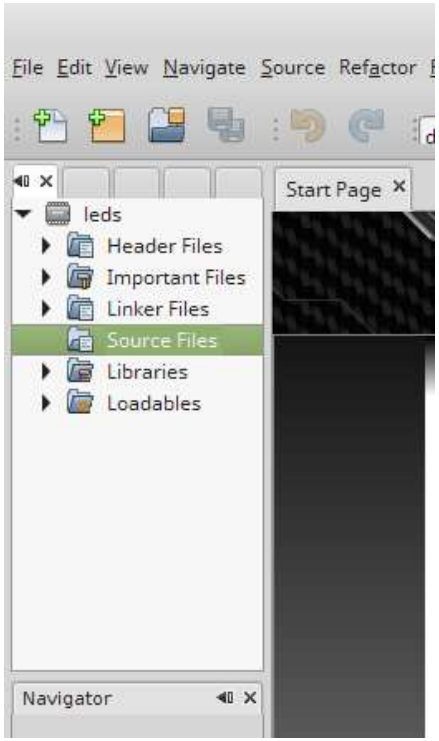
Creating a Project using the MPLABX environment (6)

- After the project has been created, an additional option must be set, to this aim select File → Project Properties
- In the open window, select XC8 Linker (left-panel) and, from **Option Categories** (right-panel), select Additonal Options
- Locate the **Code Offset** option, type, in the textbox, the value 0x4000, then click **Ok**
- Your project is now ready



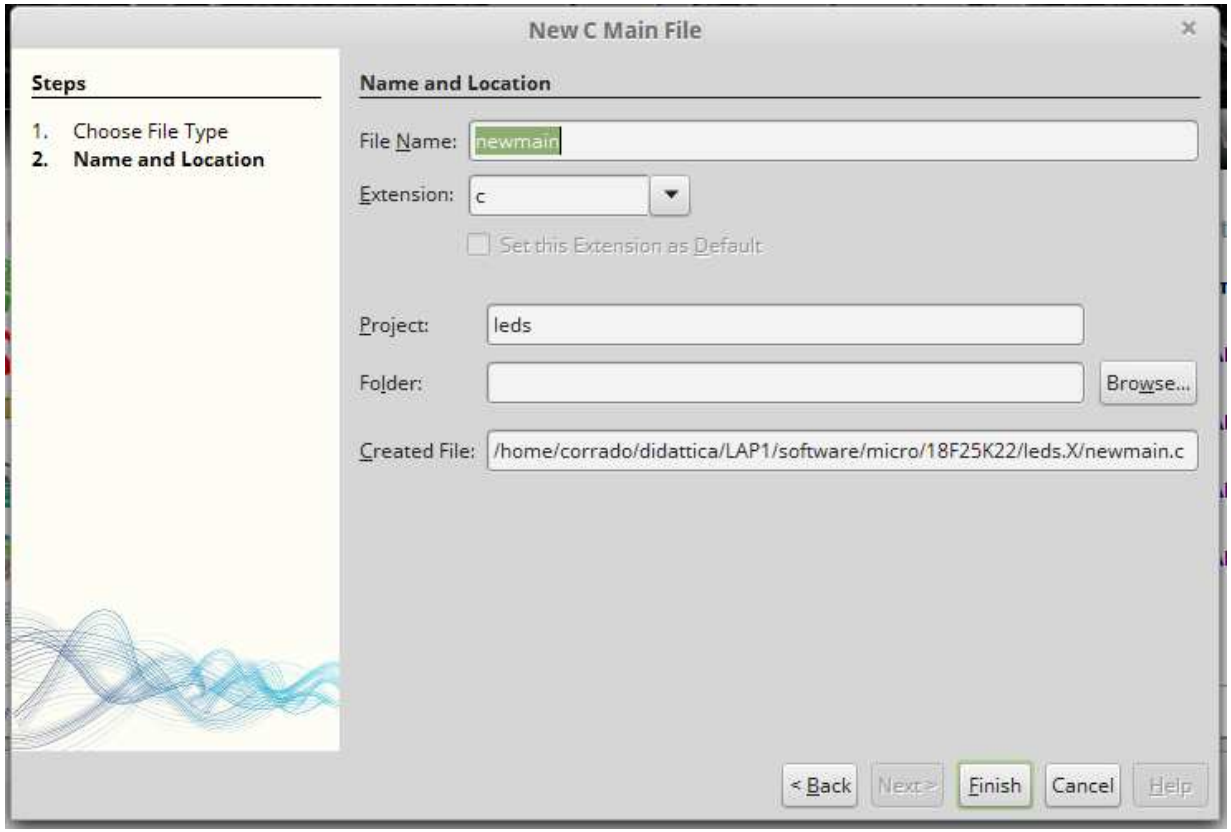
Writing the source file and compiling the program

- After the project has been created, the left panel of MPLABX contains the project files window
- Select Source files, right-click and then select
  - New ...
  - C Main file



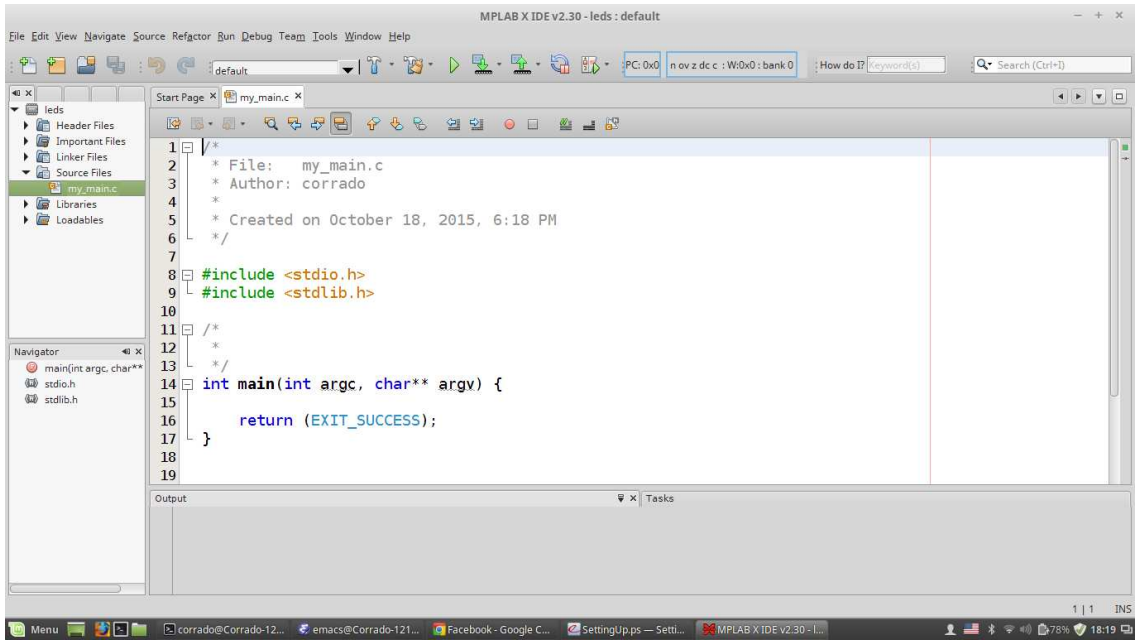
Writing the source file and compiling the program

- In the window that will appear, type-in the name you wish of your main file,
- then click **Finish**.



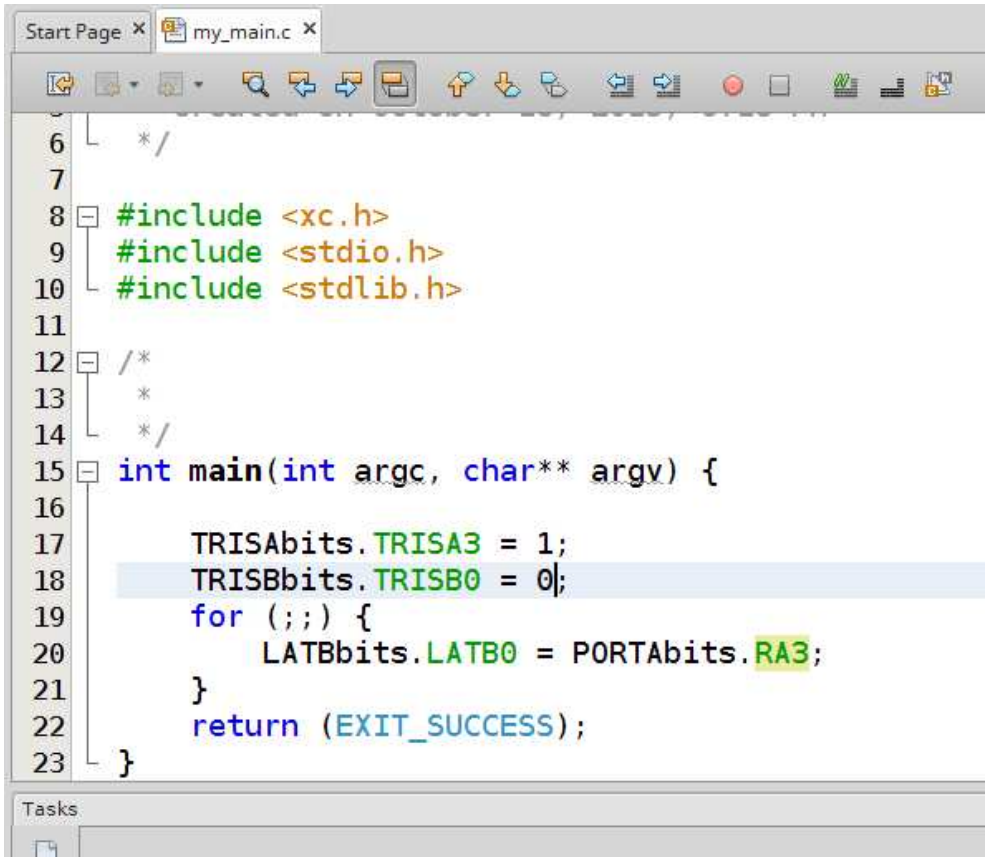
Writing the source file and compiling the program

- The file will be created with a skeleton source code and it will be open



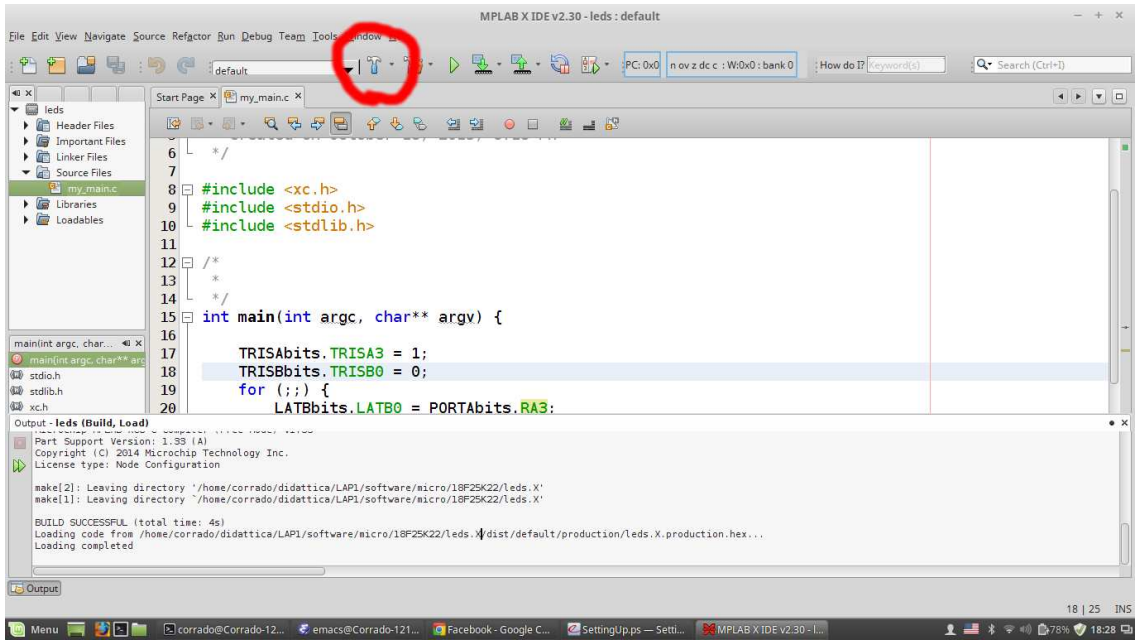
Writing the source file and compiling the program

- Now let's write our code (don't forget to include "xc.h") ...



Writing the source file and compiling the program

- ... and compile it:
- Click on the “hammer” icon in the toolbar to start compilation
- The **Output** window will appear to show the compilation process
- If everything succeeds the last message is ”BUILD SUCCESSFUL”
- Otherwise, see the error messages and apply proper corrections



Uploading the code to the MCU (instruction for Linux and Cutecom)

You first need the packages `cutecom` and `lrzsz`.

1. Connect the board using the USB;
2. Start Cutecom with "superuser" privileges: `$ sudo cutecom` (insert your password if needed)
3. Type/select the following settings:
   - Device: /dev/ttyUSB0
   - Baud rate: 19200
   - Data bits: 8
   - Stop bits: 1
   - Parity: None
   - Handshake: uncheck both software and hardware
   - Open for: check both reading and writing
   - Check Apply settings when opening
4. Then click **Open Device**

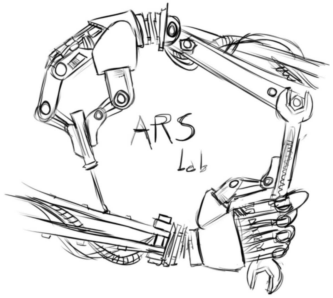
Uploading the code to the MCU (instruction for Linux and Cutecom)

1. If no error occurs, push the RESET button of your board;
2. If everything is ok, the welcome page will appear together with a "$>$" prompt.
3. To upload a file, perform the following operations:
   - In the "**Send file**" combobox (lower part of the window), select: XModem
   - In the "**Input**" textbox, type up and press ENTER
   - Now click the "**Send file**" button and select, from the browser, your binary (**.HEX**) file
   - It is located (starting from project folder) in the folter `dist/default/production/` and it is the sole **.HEX** file. Select it.
4. To start your code, when the upload process has been (successful) completed, type go in the "**Input**" textbox and press ENTER.

Setting Up the Environment for the Ready-for-PIC-28 Board

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory
Dipartimento di Matematica e Informatica - Università di Catania, Italy
santoro@dmi.unict.it



L.A.P. 1 Course