CREATING THE FIRST PROJECT IN mikroC PRO for PIC

MikroElektronika
SOFTWARE AND HARDWARE SOLUTIONS FOR EMBEDDED WORLD ... making it simple

TO OUR VALUED CUSTOMERS

I want to express my thanks to you for being interested in our products and having confidence in MikroElektronika.
It is our intention to provide you with the best quality products. Furthermore, we will continue to improve our performance to better suit your needs.

Nebojsa Matic
General Manager

Project

mikroC PRO for PIC organizes applications into projects consisting of a single project file (file with the ***.mcppi*** extension) and one or more source files (files with the ***.c*** extension). Source files are usually called headers in the mikroC programming language. The *mikroC PRO for PIC* compiler allows you to manage several projects at a time. Source files can be compiled only if they are part of the project.

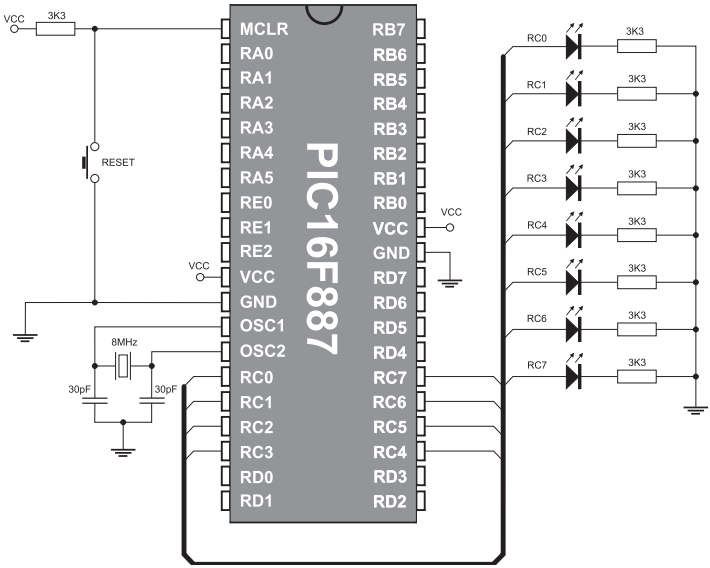
A project file contains:

- Project name and optional description;
- Target device in use;
- Device clock;
- List of the project source files;
- Binary files (***.mcl***); and
- Other files.

In this reference guide, we will create a new project, write code, compile it in the *mikroC PRO for PIC* compiler and test the results. The purpose of this example is to make LED diodes on the microcontroller PORTC blink, which will be easy to test.

Hardware Connection

For the purpose of testing this example on a PIC microcontroller, it is necessary to connect hardware as per schematic below. This program causes all the ports to change their logic state. However, in this example, LED diodes are, for the purpose of demonstration, connected to PORTC only.



Prior to creating a new project, it is necessary to do the following:

Step 1: Install the compiler

Install the *mikroC PRO for PIC* compiler from the product CD, **CD:\zip\mikroc_pro_pic\mikroc_pro_pic_setup.exe**. Desktop shortcut and start menu shortcut will be automatically created.

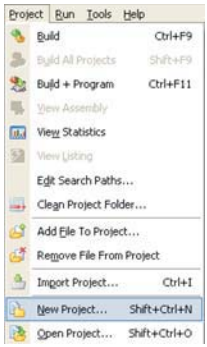
Step 2: Start up the compiler

Start up the *mikroC PRO for PIC* compiler by double clicking the appropriate icon. The *mikroC PRO for PIC IDE* (Integrated Development Environment) will appear on the screen.

Now you are ready to start creating a new project.

New Project

The process of creating a new project is very simple. Select the ***New Project*** option from the ***Project*** menu, as shown in Figure on the right.

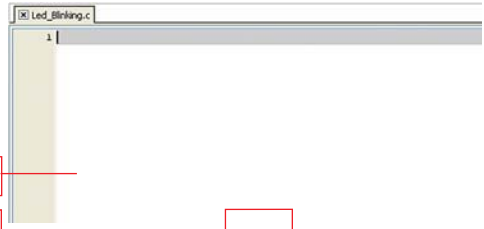


A window called ***New Project Wizard***, which will guide you through the process of creating a new project, appears. The introductory window of this application contains a list of actions to be performed when creating a new project. Click ***Next***.

The process of creating a new project can be broken up into five steps:

1. Selecting the microcontroller to write a program for. In this case it is PIC16F887.
2. Selecting the device clock. In this case, it is 8 MHz clock.
3. Selecting the name and location of the project. In this case, the project name is *First Project* and it will be saved in the *C:\My projects* folder. The compiler automatically append the ***.mcppi*** extension to the project name and a source file having the same name (*First Project* ***.c***) will be created within it.
4. In the event that the project consists of several source files, it is necessary to specify them all and include into the project by clicking the ***Add*** button. In this example, there are no additional source files within the project.
5. After all, it is necessary to confirm all selected options by clicking ***Finish***.

After creating the project, a new blank window to write a program in will appear. See Figure below.



Type your program here.

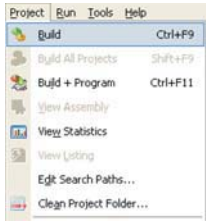
Your program is to look as follows:

Led_Blinking.c

```
/*
 * Project name:
     LED_Blinking (Simple 'Hello World' project)
 * Copyright:
     (c) MikroElektronika, 2005-2008
 * Description:
     This is a simple 'Hello World' project. It turns on/off diodes connected to
     PORTC. It uses bitwise negation to toggle PORTC pins.
 * Test configuration:
     MCU:             PIC16F887
     Dev.Board:       EasyPIC5
     Oscillator:      HS, 08.0000 MHz
     Ext. Modules:    -
     SW:              mikroC v8.0
 * NOTES:
     None.
*/

void main() {
  PORTC = 0;                // Initialize PORTC
  TRISC = 0;                // Configure PORTC as output
  ANSELH = 0;               //Configure AN pins as digital I/O
  ANSEL = 0;

  while(1) {

   PORTC =  ~PORTC;         // toggle PORTC
    Delay_ms(1000);         // one second delay
  }
}
```

Compilation

When the program is written, it is necessary to compile it into a program (.hex) code, by selecting one of the build options from the ***Project*** menu:



- To create a HEX file, select ***Build*** (Ctrl+F9) from the ***Project*** menu or click the ***Build*** icon from the ***Project*** toolbar.
- The ***Build All Projects*** (Shift+F9) option builds all files within the project, libraries (if there is a source code for them) and **def** files for the chip in use.
- The ***Build + Program*** (Ctrl+F11) option is special as it enables the *mikroC PRO for PIC* compiler to automatically load the program into the microcontroller after compilation. The process of programming is performed by using the *PICflash* programmer.

All the errors detected during compilation will be shown in the ***Messages*** window. If no errors are encountered, the *mikroC PRO for PIC* compiler will generate output files.

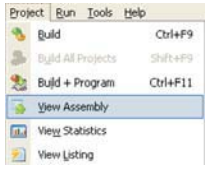
Output Files

The *mikroC PRO for PIC* compiler generates output files in the project folder containing the project file. Output files are summarized in the table below:

Format	Description	File Type
Intel HEX	Intel hex style records. It is used for programming PIC microcontrollers.	.hex
Binary	Compiled Library which may be included in other projects.	.mcl
Assembler File	Assembly file with symbolic names.	.asm
List File	Overview of PIC memory allotment. The List File represents an extended version of assembly code, i.e. contains addresses of instructions, registers, routines and labels.	.lst

Assembly Code Overview

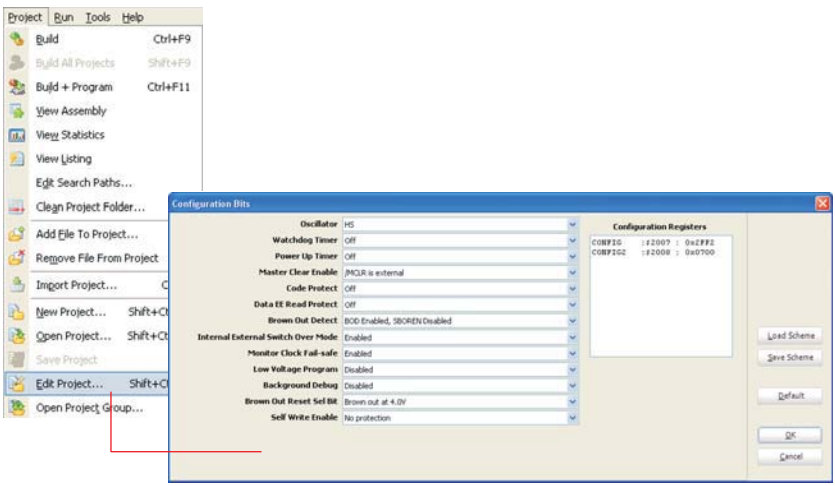
After compilation, click the ***View Assembly*** icon or select the ***View Assembly*** option from the ***Project*** menu to review the generated assembly code in a new window. To overview the complete *List File*, select the ***View Listing*** option from the same menu.



Project Settings

By selecting the ***Project Settings*** option from the ***View*** menu, a new window providing possibility to change type and clock frequency of the microcontroller as well as the mode of operation of the compiler and debugger after compilation, appears.

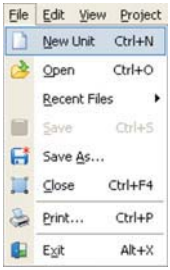
It is also possible to change configuration word by entering new data in the ***Configuration Bits*** window which appears by selecting the ***Edit Project*** option in the ***Project*** menu. Any change made in this window refers to the currently active project only.



Source Files

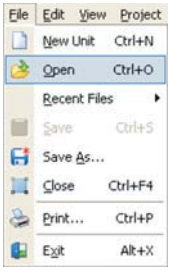
Creating a new source file

In order to create a new source file, it is necessary to select the ***New Unit*** option from the ***File*** menu, or press CTRL+N or click the ***New File*** icon from the ***File*** toolbar. A new window, i.e. a new source file automatically appears. Select the ***Save*** option from the ***File*** menu, or press CTRL+S or click the ***Save File*** icon from the ***File*** toolbar and name it as you want.



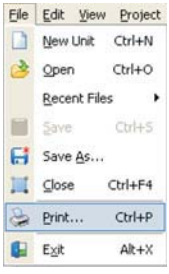
Opening an existing file

In order to open a saved file, it is necessary to select the ***Open*** option from the ***File*** menu, or press CTRL+O or click the ***Open File*** icon from the ***File*** toolbar. In the ***Open*** dialog box, browse the location of the file that you want to open, select it and click the ***Open*** button.
The file will be automatically displayed in its own window. If such file is already open, it becomes active.



Printing an active file

First of all, it is necessary to make sure that the file you want to print is active. Select the ***Print*** option from the ***File*** menu or press CTRL+P and click the ***OK*** button.
In the ***Print Preview*** window, set a desired layout of the document and click the ***Print*** icon.



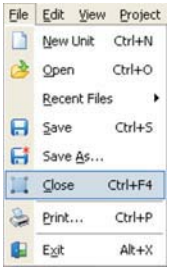
Saving file

Make sure that the file you want to save is active.
In order to save it, it is necessary to select the ***Save*** option from the ***File*** menu, or press Ctrl+S, or click the ***Save File*** icon from the ***File*** toolbar.

Saving file under different name

Make sure that the file you want to save is active.
In order to save it under different name, it is necessary to select the ***Save As*** option from the ***File*** menu. The ***Save As*** dialog box will appear. Here you can browse the folder in which you want to save the file.
In the ***File Name*** field, modify the name of the file you want to save and click the ***Save*** button.

Closing file



Make sure that the file you want to close is active.
In order to close it, it is necessary to select the ***Close*** option from the ***File*** menu, or right click the window of the file you want to close and select the ***Close*** option. If the file has been changed since it was last saved, you will be prompted to save the changes.

DISCLAIMER

All the products owned by MikroElektronika are protected by copyright law and international copyright treaty. Therefore, this manual is to be treated as any other copyright material. No part of this manual, including product and software described herein, may be reproduced, stored in a retrieval system, translated or transmitted in any form or by any means, without the prior written permission of MikroElektronika. The manual PDF edition can be printed for private or local use, but not for distribution. Any modification of this manual is prohibited.

MikroElektronika provides this manual 'as is' without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties or conditions of merchantability or fitness for a particular purpose.

MikroElektronika shall assume no responsibility or liability for any errors, omissions and inaccuracies that may appear in this manual. In no event shall MikroElektronika, its directors, officers, employees or distributors be liable for any indirect, specific, incidental or consequential damages (including damages for loss of business profits and business information, business interruption or any other pecuniary loss) arising out of the use of this manual or product, even if MikroElektronika has been advised of the possibility of such damages. MikroElektronika reserves the right to change information contained in this manual at any time without prior notice, if necessary.

All the product and corporate names appearing in this manual may or may not be registered trademarks or copyrights of their respective companies, and are only used for identification or explanation and to the owners' benefit, with no intent to infringe.

HIGH RISK ACTIVITIES

The products of MikroElektronika are not fault – tolerant nor designed, manufactured or intended for use or resale as on – line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of Software could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). MikroElektronika and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

If you want to learn more about our products, please visit our website: www.mikroe.com

If you are experiencing some problems with any of our products or just need additional information, please place your ticket at: www.mikroe.com/en/support

If you have any question, comment or business proposal, do not hesitate to contact us: office@mikroe.com