How to write Assembly programs in Keil

(Version 1.3)

Sepehr Naimi

Mazidi & Naimi tutorials

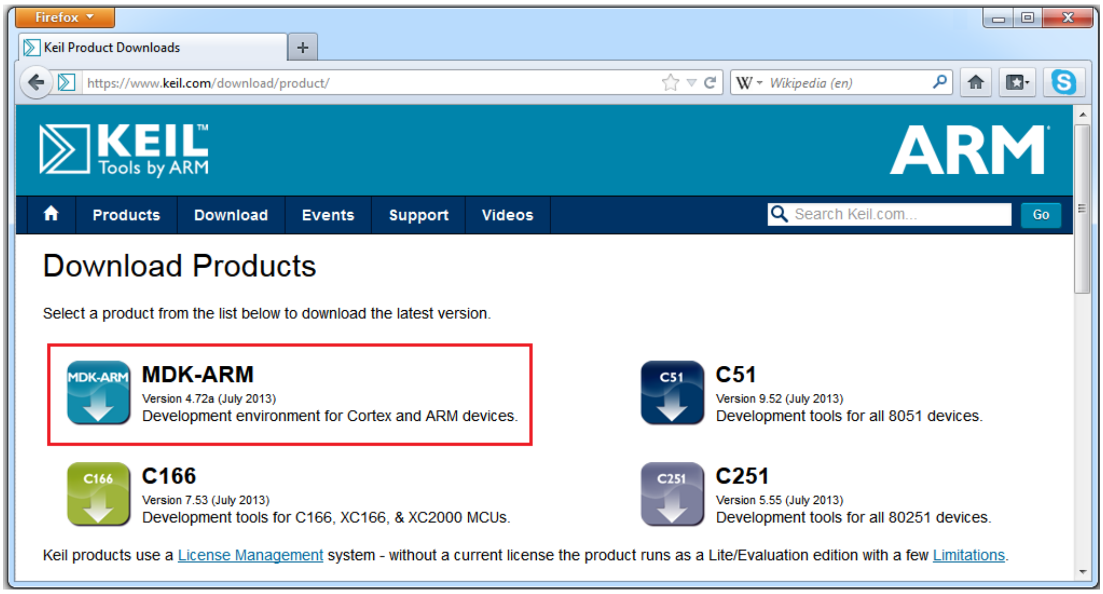
MicroDigitalEd.com and BIHE University

Contents

Downloading and installing the Keil IDE

1. Click on the following link:
   https://www.keil.com/download/product/

2. Choose MDK-ARM.



Download Products

Select a product from the list below to download the latest version.

MDK-ARM
Version 4.72a (July 2013)
Development environment for Cortex and ARM devices.

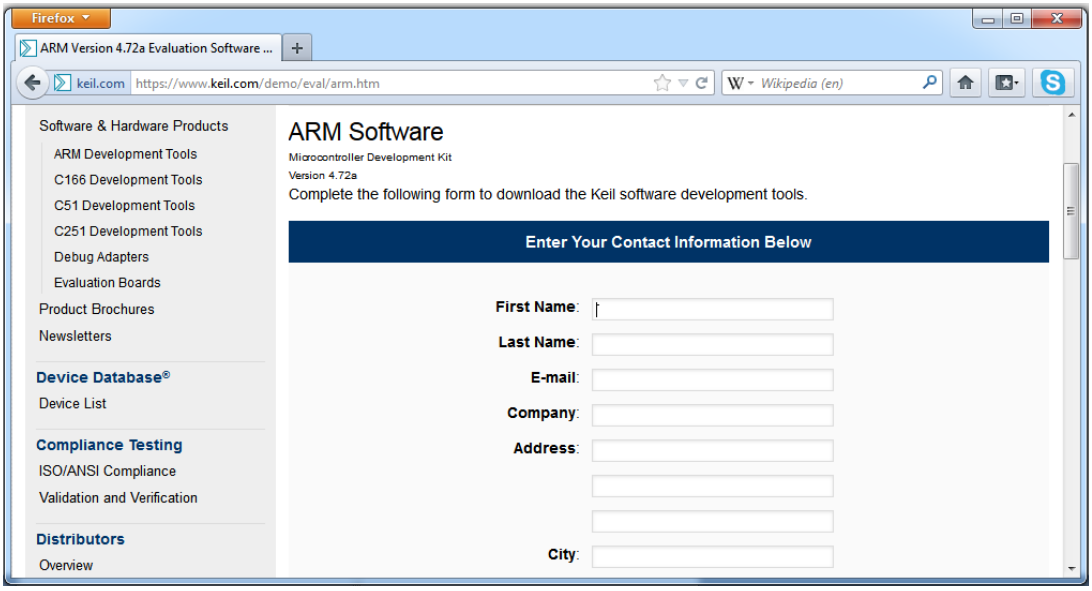
C51
Version 9.52 (July 2013)
Development tools for all 8051 devices.

C166
Version 7.53 (July 2013)
Development tools for C166, XC166, & XC2000 MCUs.

C251
Version 5.55 (July 2013)
Development tools for all 80251 devices.

Keil products use a License Management system - without a current license the product runs as a Lite/Evaluation edition with a few Limitations.

3. Fill out the form and then click on the ***submit*** button, at the bottom of the page.

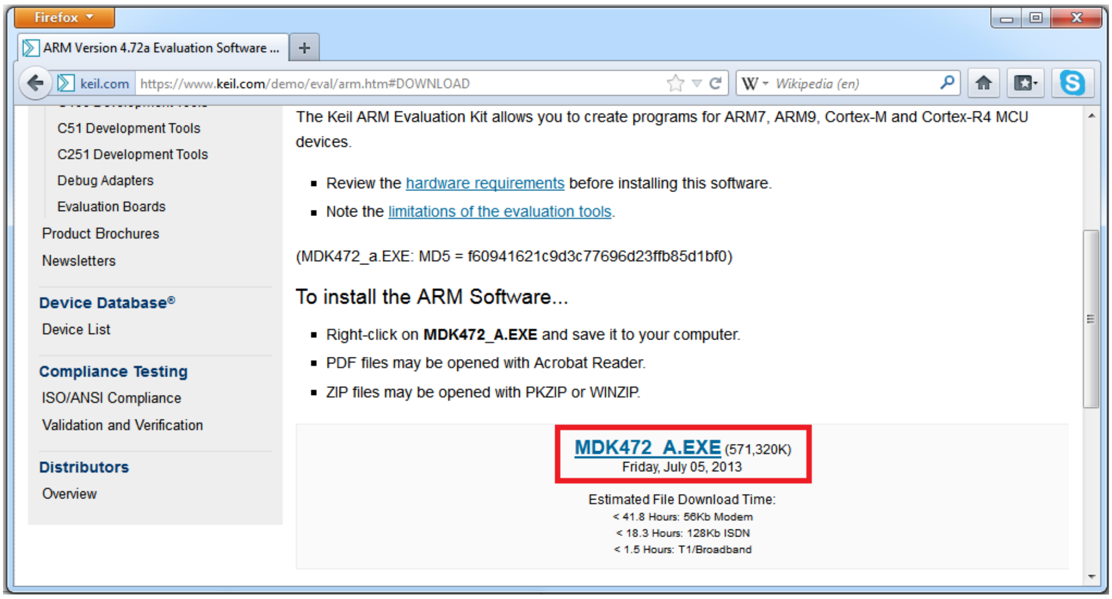


ARM Software
Microcontroller Development Kit
Version 4.72a
Complete the following form to download the Keil software development tools.

Enter Your Contact Information Below

First Name:
Last Name:
E-mail:
Company:
Address:
City:

4. Click on the ***MDKxxx.EXE*** link.



The Keil ARM Evaluation Kit allows you to create programs for ARM7, ARM9, Cortex-M and Cortex-R4 MCU devices.

- Review the hardware requirements before installing this software.
- Note the limitations of the evaluation tools.

(MDK472_a.EXE: MD5 = f60941621c9d3c77696d23ffb85d1bf0)

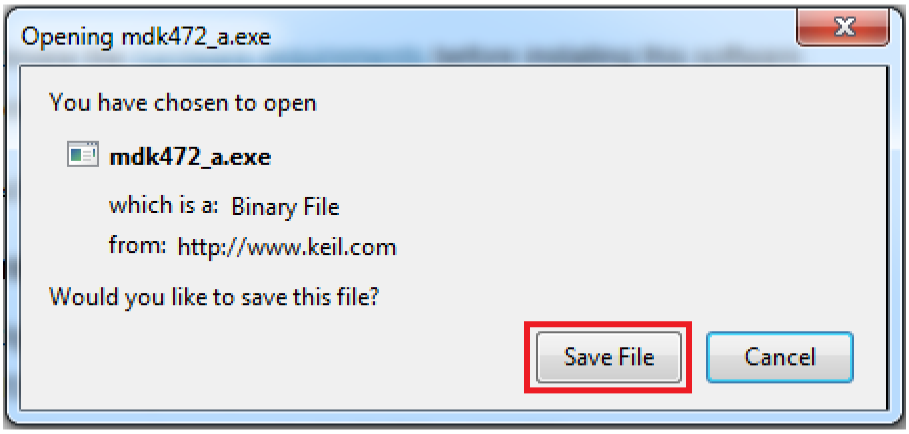
To install the ARM Software...

- Right-click on **MDK472_A.EXE** and save it to your computer.
- PDF files may be opened with Acrobat Reader.
- ZIP files may be opened with PKZIP or WINZIP.

MDK472_A.EXE (571,320K)
Friday, July 05, 2013

Estimated File Download Time:
< 41.8 Hours: 56Kb Modem
< 18.3 Hours: 128Kb ISDN
< 1.5 Hours: T1/Broadband

5. Choose ***Save***.



Opening mdk472_a.exe

You have chosen to open

mdk472_a.exe

which is a: Binary File

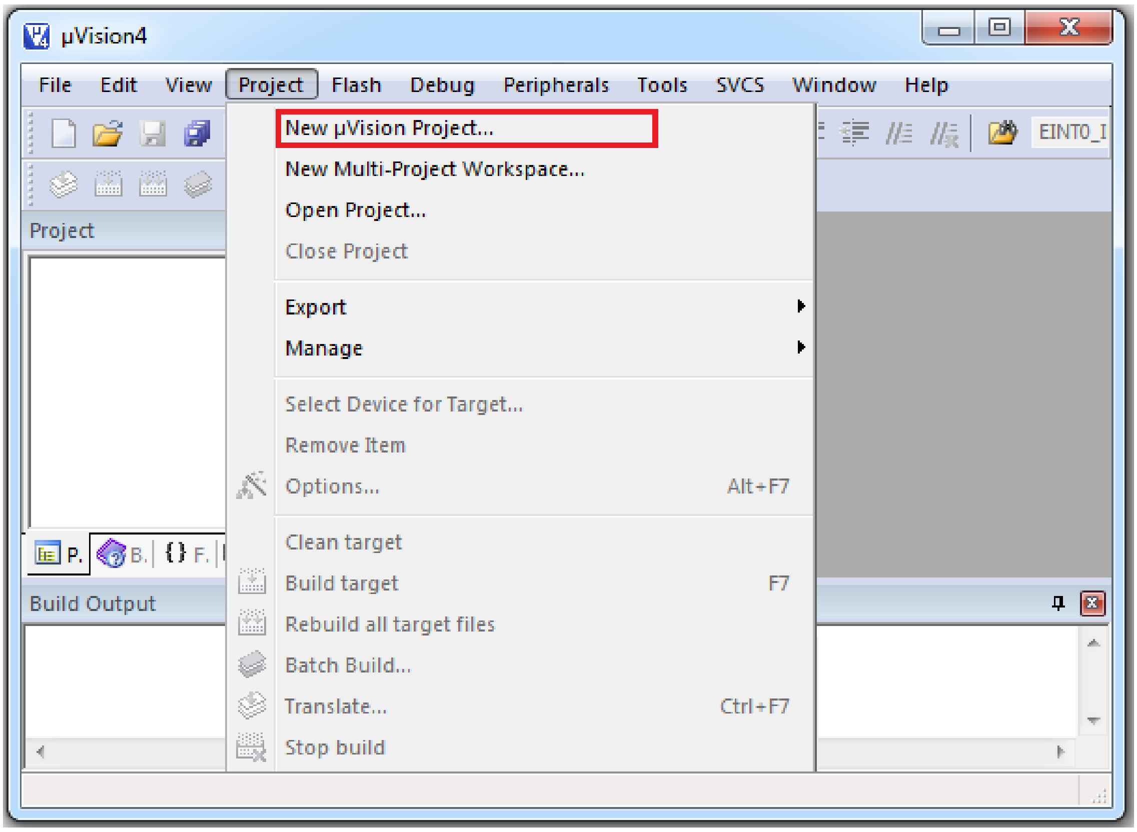
from: http://www.keil.com

Would you like to save this file?

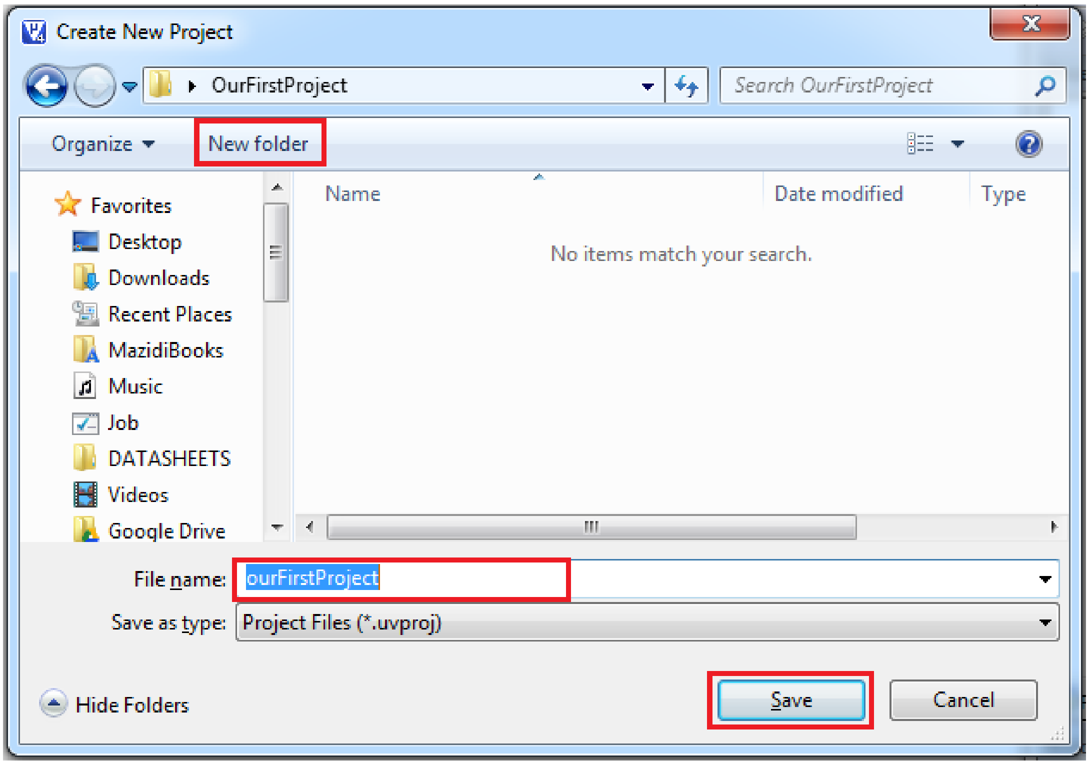
6. Execute the downloaded file. Installing the software is straight forward; choose the ***Next*** button in each step, until the setup finishes.

Creating an Assembly project in Keil

7. Open the Keil IDE by clicking on its icon on the desktop.
8. Choose ***New uVision Project*** from the ***Project*** menu.



9. Create a new folder and Name it ***OurFirstProject.*** Type the name ***ourFirstProject*** for the project and click ***Save***.



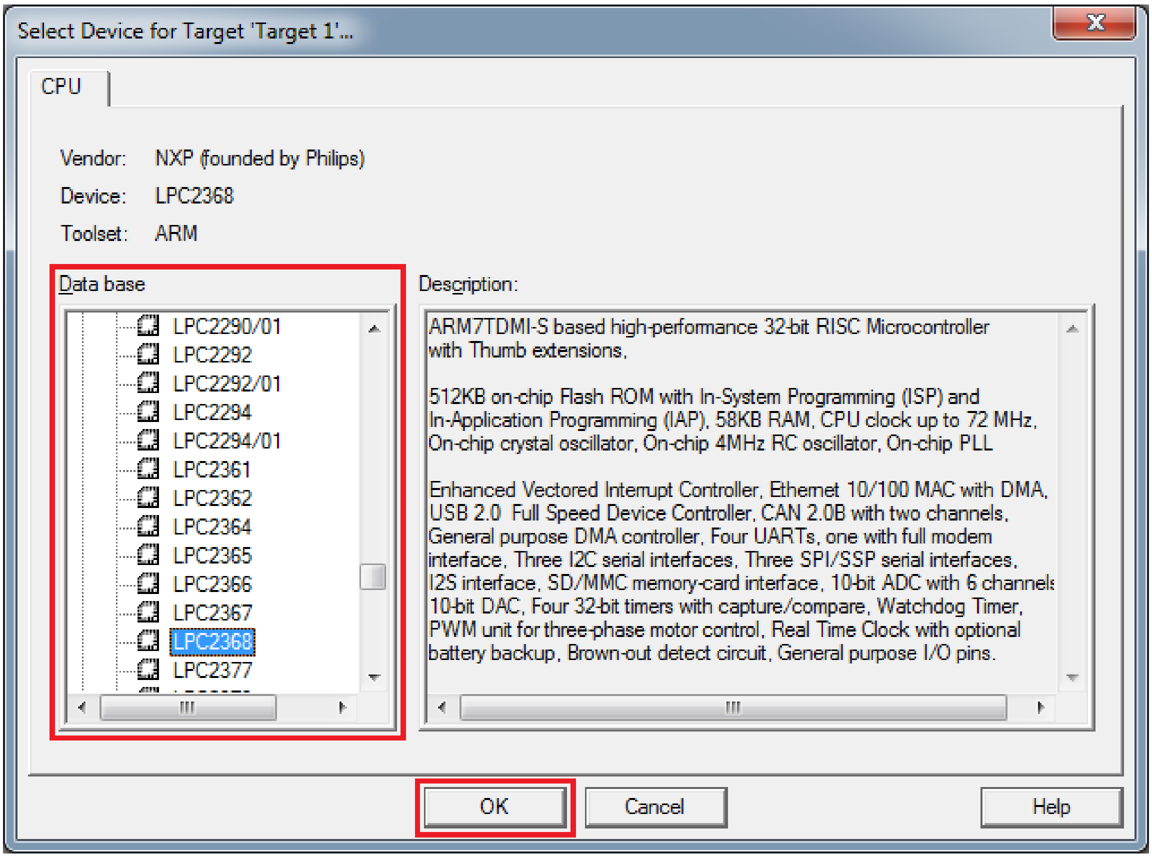
File name: ourFirstProject

Save as type: Project Files (*.uvproj)

10. In the ***Data base*** tree, choose the vendor and then the chip you want to use and then click ***OK***. For example, if you want to use the ***LPC2368***, click on the ***NXP*** and then on the ***LPC2368*** and then press ***OK***.

Caution

When you choose the chip some general information of the chip is shown in the ***Description*** box.



Vendor: NXP (founded by Philips)

Device: LPC2368

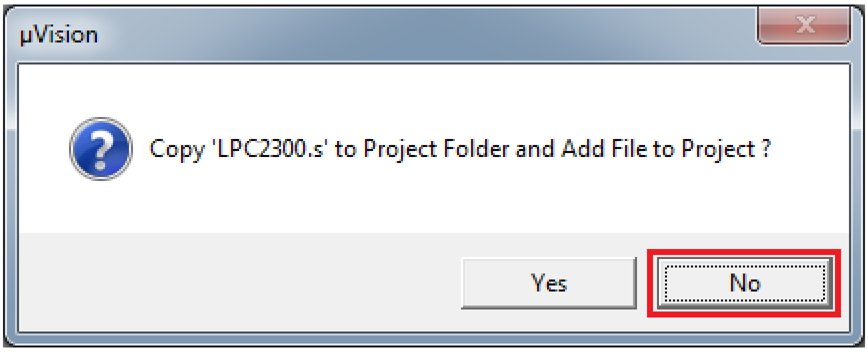
Toolset: ARM

ARM7TDMI-S based high-performance 32-bit RISC Microcontroller with Thumb extensions.

512KB on-chip Flash ROM with In-System Programming (ISP) and In-Application Programming (IAP), 58KB RAM, CPU clock up to 72 MHz, On-chip crystal oscillator, On-chip 4MHz RC oscillator, On-chip PLL

Enhanced Vectored Interrupt Controller, Ethernet 10/100 MAC with DMA, USB 2.0 Full Speed Device Controller, CAN 2.0B with two channels, General purpose DMA controller, Four UARTs, one with full modem interface, Three I2C serial interfaces, Three SPI/SSP serial interfaces, I2S interface, SD/MMC memory-card interface, 10-bit ADC with 6 channels, 10-bit DAC, Four 32-bit timers with capture/compare, Watchdog Timer, PWM unit for three-phase motor control, Real Time Clock with optional battery backup, Brown-out detect circuit, General purpose I/O pins.

11. Click the ***No*** button to add the startup file to the project.

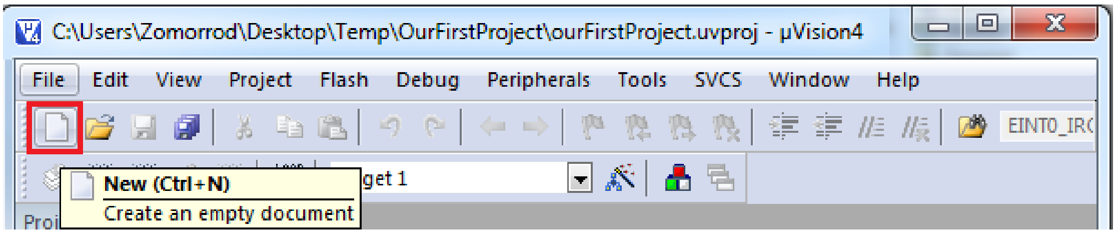


Copy 'LPC2300.s' to Project Folder and Add File to Project ?

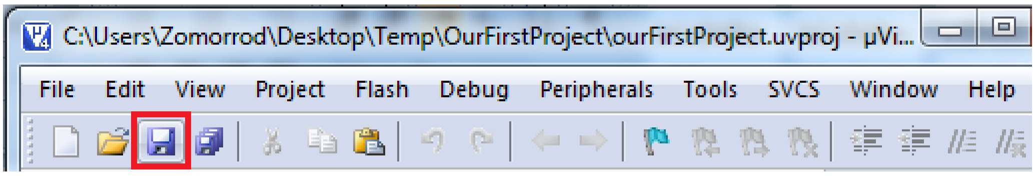
Caution

In Cortex series we should choose to add the startup file, as will be discussed in a separate tutorial.

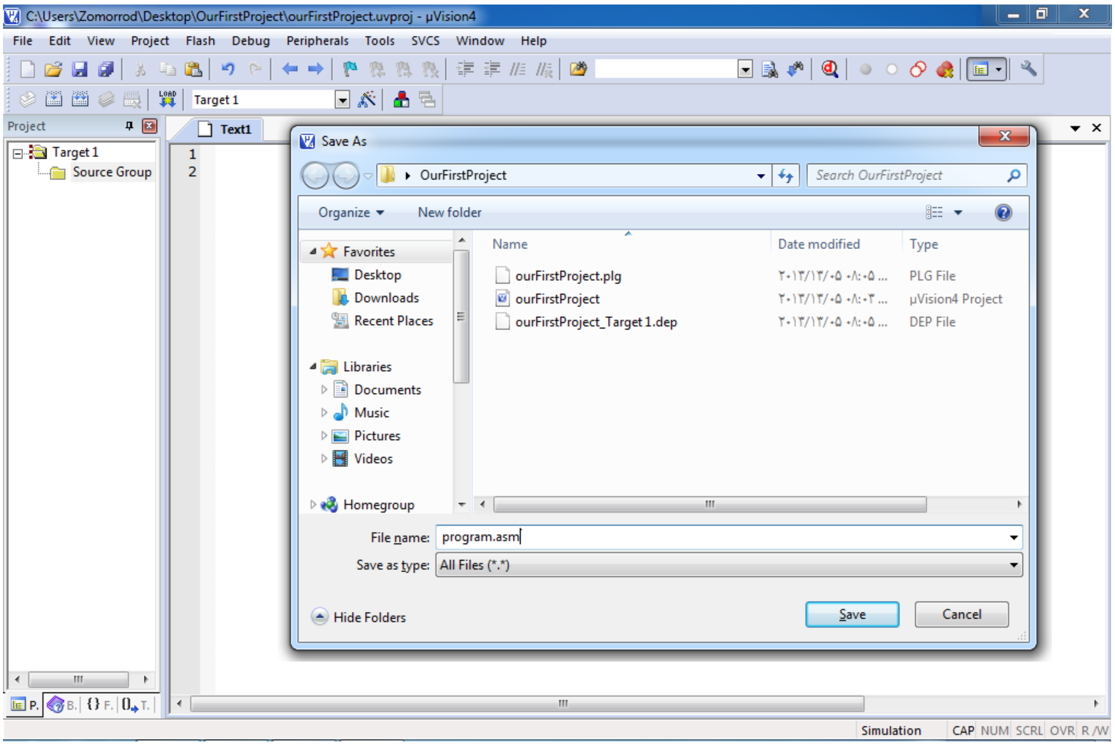
12. Make a new file by clicking on the New Icon (you can make a new file by pressing ***Ctrl+N*** or choosing ***New*** from the ***File*** menu, as well.)


C:\Users\Zomorrod\Desktop\Temp\OurFirstProject\ourFirstProject.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
EINT0_IRQ
New (Ctrl+N)
Create an empty document
get 1

13. Press ***Ctrl+S*** to save the new file. (You can also save the file by choosing ***Save*** from the ***File*** menu.)


C:\Users\Zomorrod\Desktop\Temp\OurFirstProject\ourFirstProject.uvproj - µVi...
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help

14. Name the file as ***program.asm*** and save it in the ***OurFirstProject*** directory.

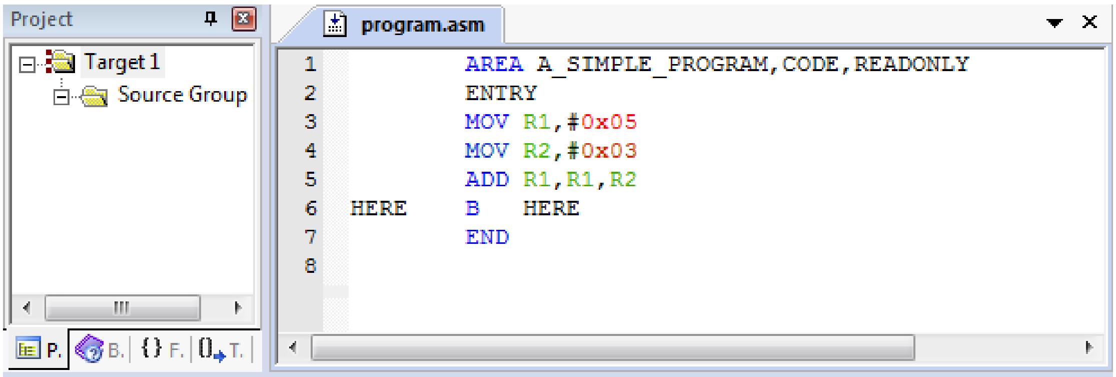


Name	Date modified	Type
ourFirstProject.plg	٢٠١٣/١٢/٠٥ ٠٨:٠٥ ...	PLG File
ourFirstProject	٢٠١٣/١٢/٠٥ ٠٨:٠٢ ...	µVision4 Project
ourFirstProject_Target1.dep	٢٠١٣/١٢/٠٥ ٠٨:٠٥ ...	DEP File

File name: program.asm

Save as type: All Files (*.*)

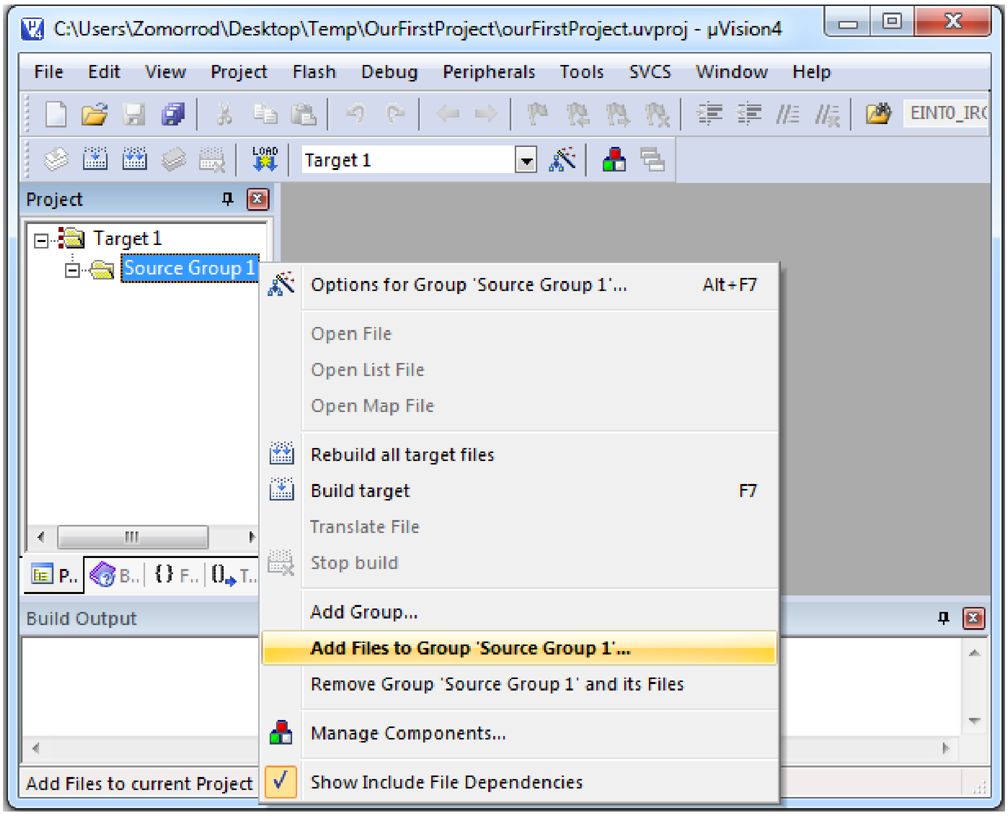
15. Type the following sample program in the file



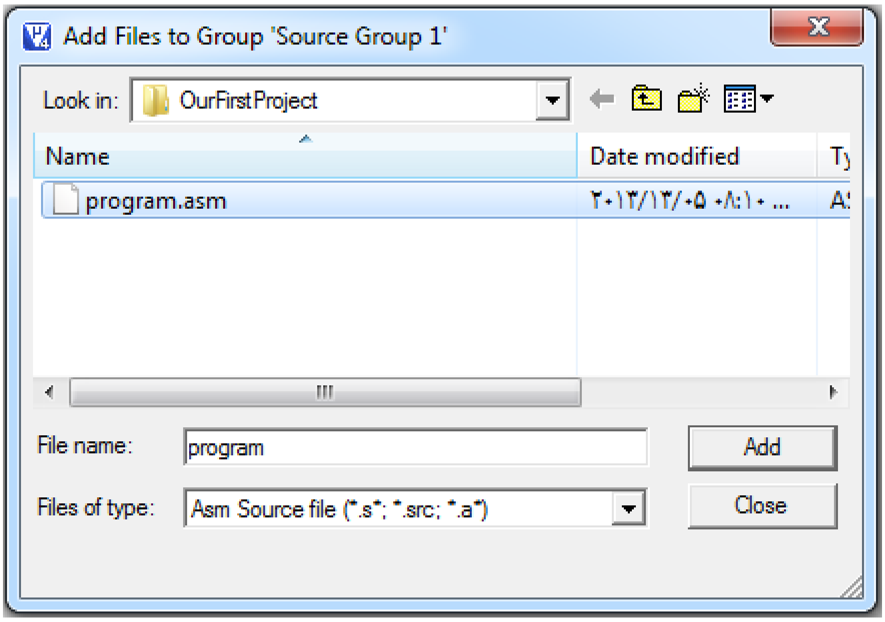
```
        AREA A_SIMPLE_PROGRAM,CODE,READONLY
        ENTRY
        MOV R1,#0x05
        MOV R2,#0x03
        ADD R1,R1,R2
HERE    B   HERE
        END
```

16. Add the ***program.asm*** file to the project. To do so:

   a. Right click on ***Source Group 1*** and choose ***Add Files to Group***.


C:\Users\Zomorrod\Desktop\Temp\OurFirstProject\ourFirstProject.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
EINT0_IRQ
Target 1
Project
Target 1
Source Group 1
Options for Group 'Source Group 1'... Alt+F7
Open File
Open List File
Open Map File
Rebuild all target files
Build target F7
Translate File
Stop build
Add Group...
Add Files to Group 'Source Group 1'...
Remove Group 'Source Group 1' and its Files
Manage Components...
Show Include File Dependencies
P.. B.. {} F.. 0→ T..
Build Output
Add Files to current Project

   b. Then go to the ***OurFirstProject*** directory and choose ***Program.asm***, press ***Add*** and then ***Close***.


Add Files to Group 'Source Group 1'

Look in: OurFirstProject

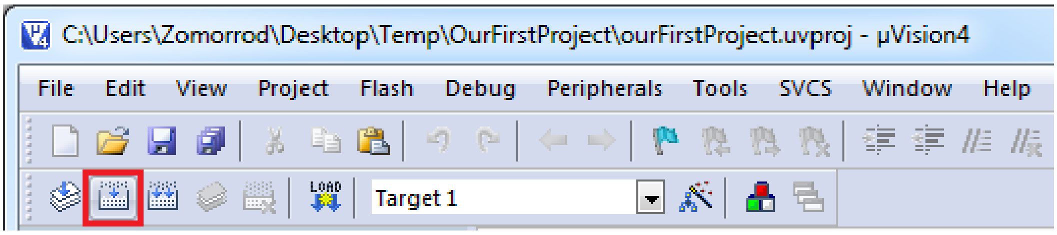
Name	Date modified	Ty
program.asm	۲۰۱۳/۱۲/۰۵ ۰۸:۱۰ ...	A:

File name: program

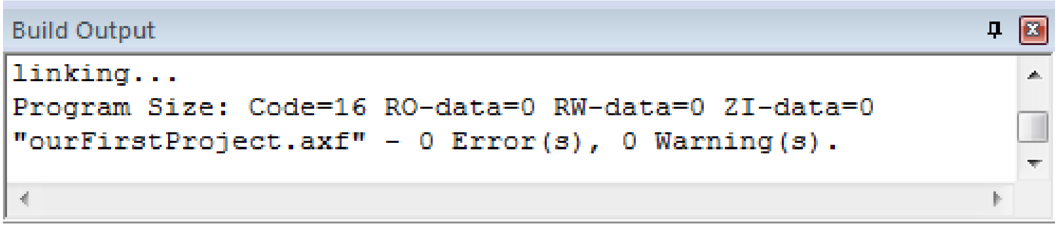
Files of type: Asm Source file (*.s*; *.src; *.a*)

Building

17. To compile click on the ***Build*** icon or choose ***build target*** from the ***Project*** menu.


C:\Users\Zomorrod\Desktop\Temp\OurFirstProject\ourFirstProject.uvproj - µVision4
File
Edit
View
Project
Flash
Debug
Peripherals
Tools
SVCS
Window
Help
LOAD
Target 1

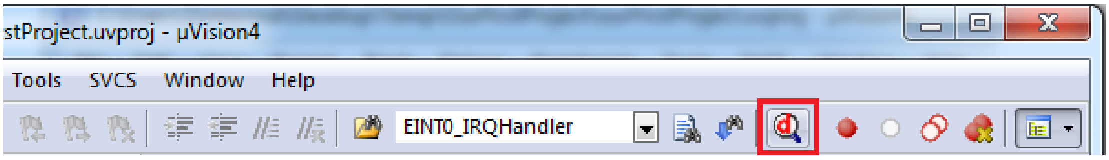
18. If the program is built successfully the following message appears:



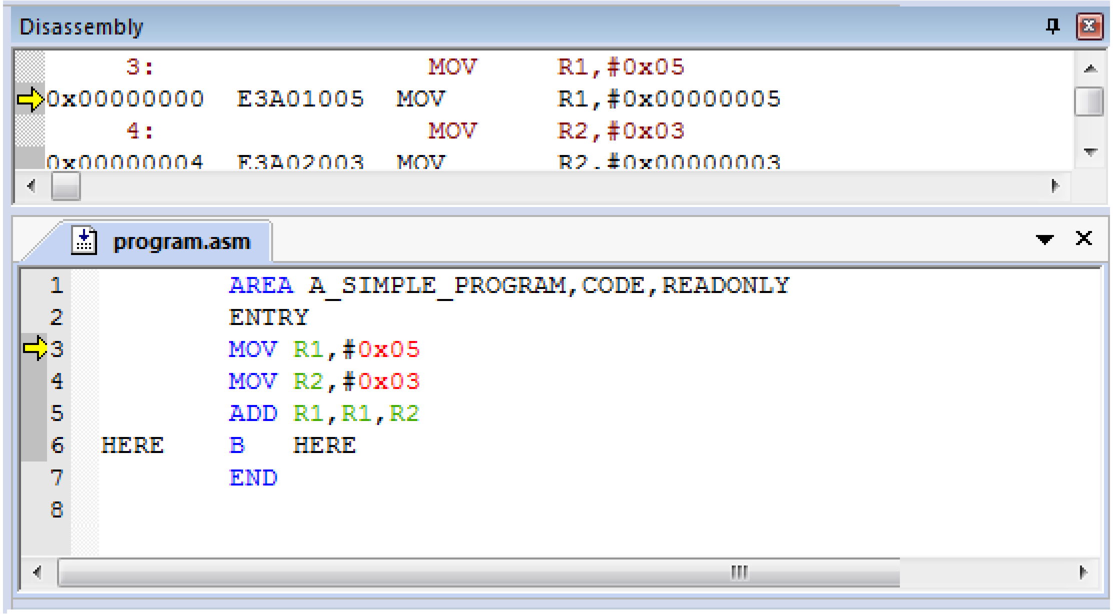
```
linking...
Program Size: Code=16 RO-data=0 RW-data=0 ZI-data=0
"ourFirstProject.axf" - 0 Error(s), 0 Warning(s).
```

Debugging and Tracing

19. To start debugging click on ***Start/Stop Debug Session*** icon or choose ***Start/Stop Debug Session*** from the ***Debug*** menu. (or simply press ***Ctrl+F5***)



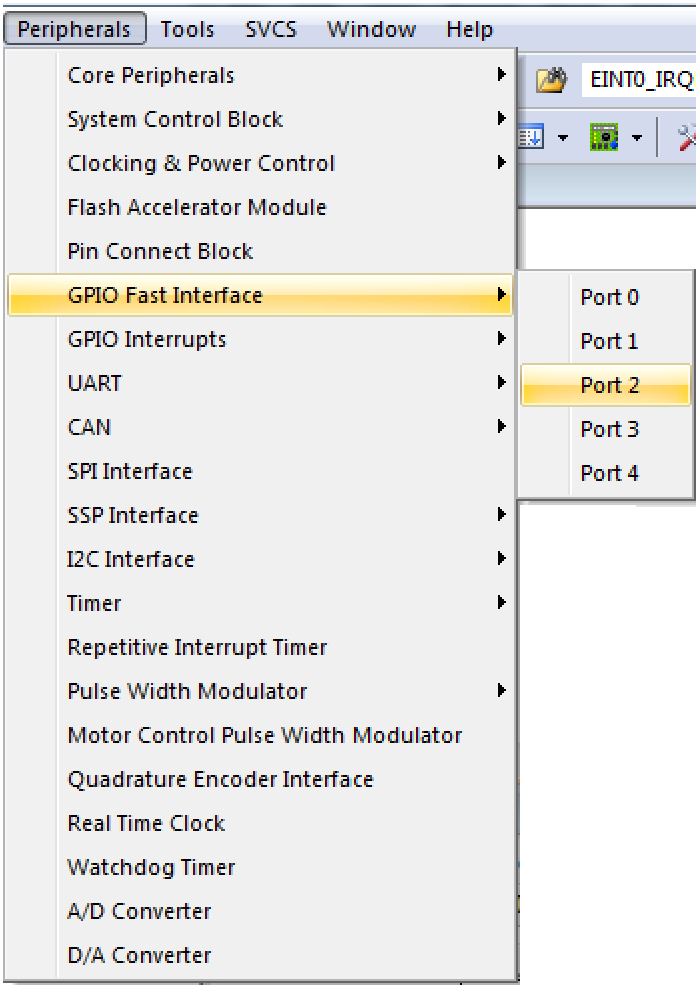
20. If it starts tracing successfully, a cursor appears in front of the next instruction to be executed.

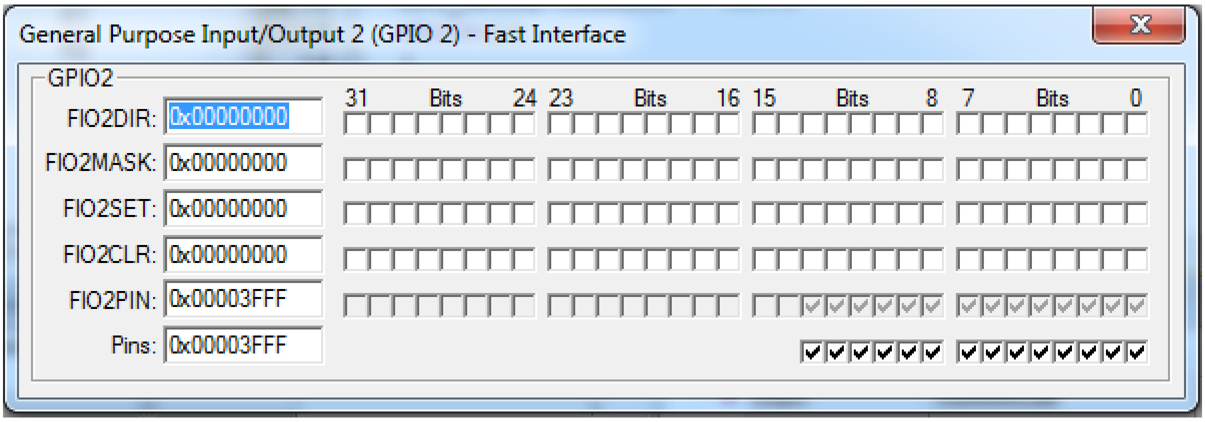


```
     3:                   MOV      R1,#0x05
0x00000000  E3A01005  MOV      R1,#0x00000005
     4:                   MOV      R2,#0x03
0x00000004  E3A02003  MOV      R2,#0x00000003
```

```
        AREA A_SIMPLE_PROGRAM,CODE,READONLY
        ENTRY
        MOV R1,#0x05
        MOV R2,#0x03
        ADD R1,R1,R2
HERE    B   HERE
        END
```

21. Go to the ***Peripherals*** menu. It has tools for monitoring different peripherals. For now, choose ***Port 2*** from ***GPIO Fast Interface***. It shows the registers of **Port 2;** you can see values of registers while tracing the program or change their values by clicking on each bit.





FIO2DIR: 0x00000000

FIO2MASK: 0x00000000

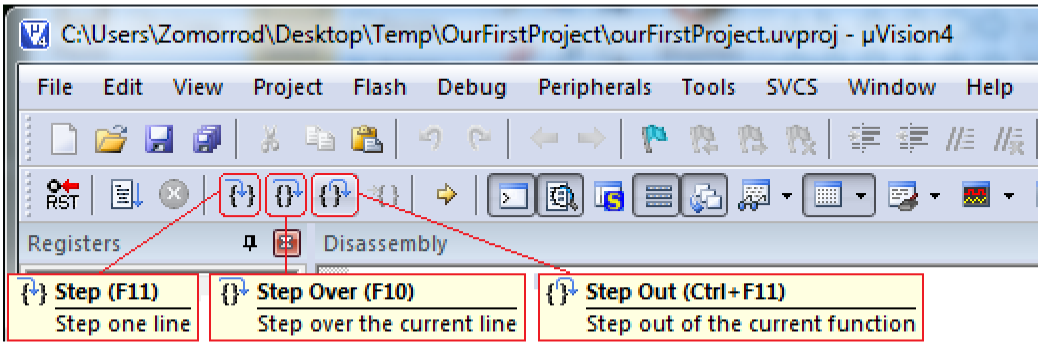
FIO2SET: 0x00000000

FIO2CLR: 0x00000000

FIO2PIN: 0x00003FFF

Pins: 0x00003FFF

22. To trace the program use the ***Step Over*** button or click on ***Step Over*** from the ***Debug*** menu. It executes the instructions of the program one after another. To trace the program you can use the ***Step*** button, as well. The difference between the ***Step Over*** and ***Step*** is in executing functions. While ***Step*** goes into the function and executes its instructions one by one, ***Step Over*** executes the function completely and goes to the instruction next to the function. To see the difference between them, trace the program once with ***Step Over*** and then with ***Step***. When you are in the function and you want the function to be executed completely you can use ***Step Out***. In the case, the instructions of the function will be executed, it returns from the function, and goes to the instruction which is next to the function call.



23. To exit from the debugging mode press ***Start/Stop Debug Session***.