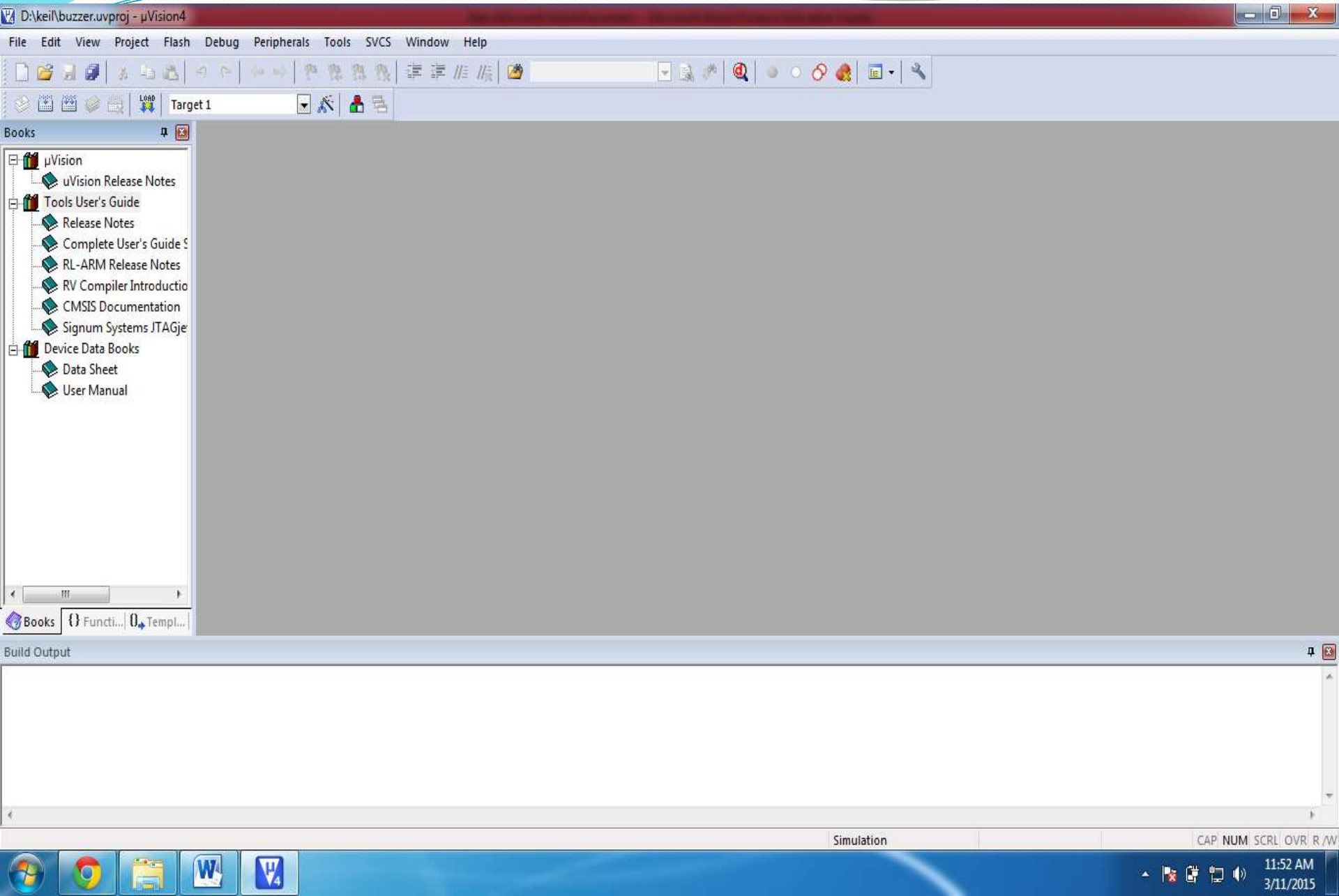


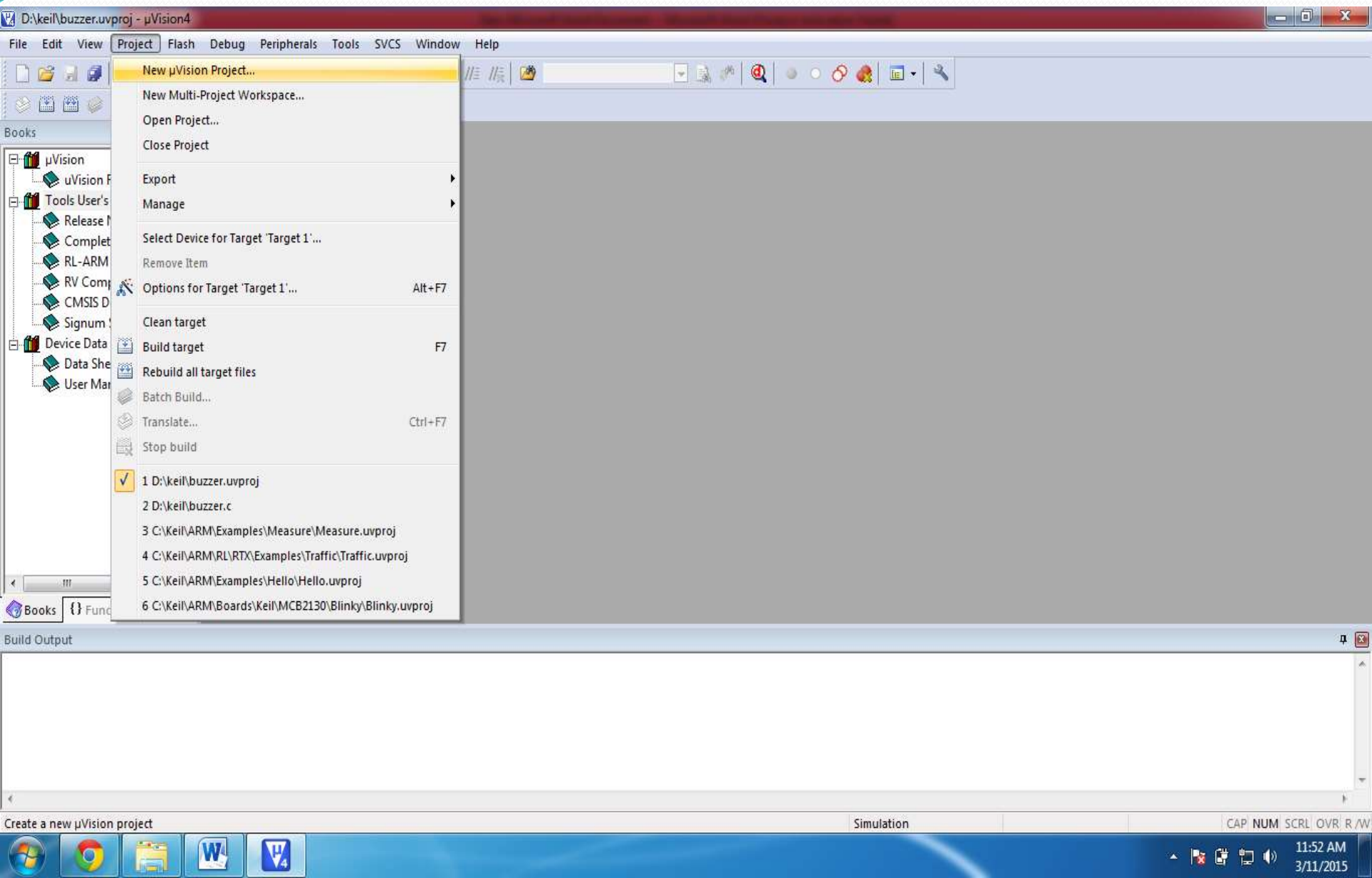


Keil software

Open keil software:

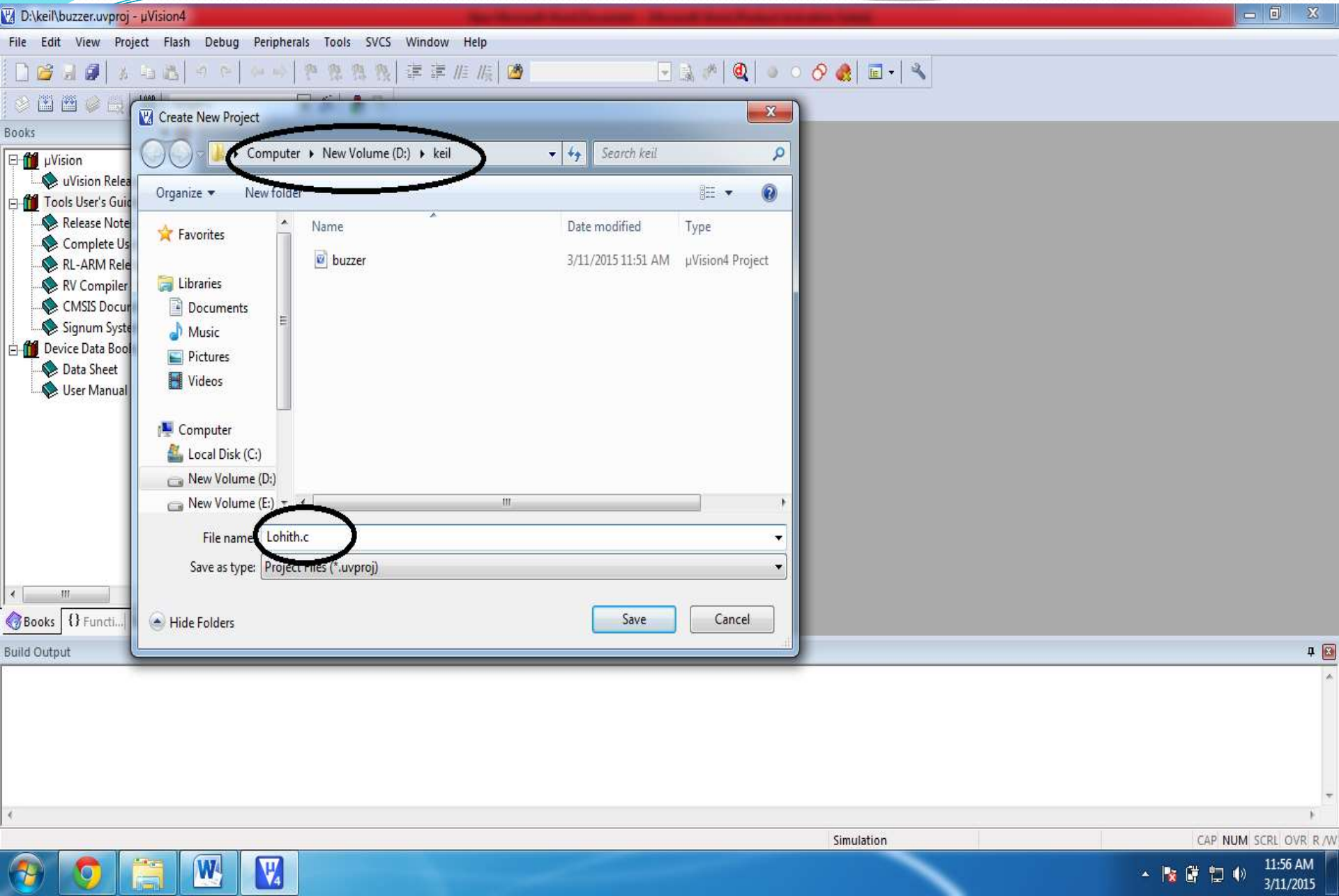


Go to Projects in the Menu and select new μ Vision project

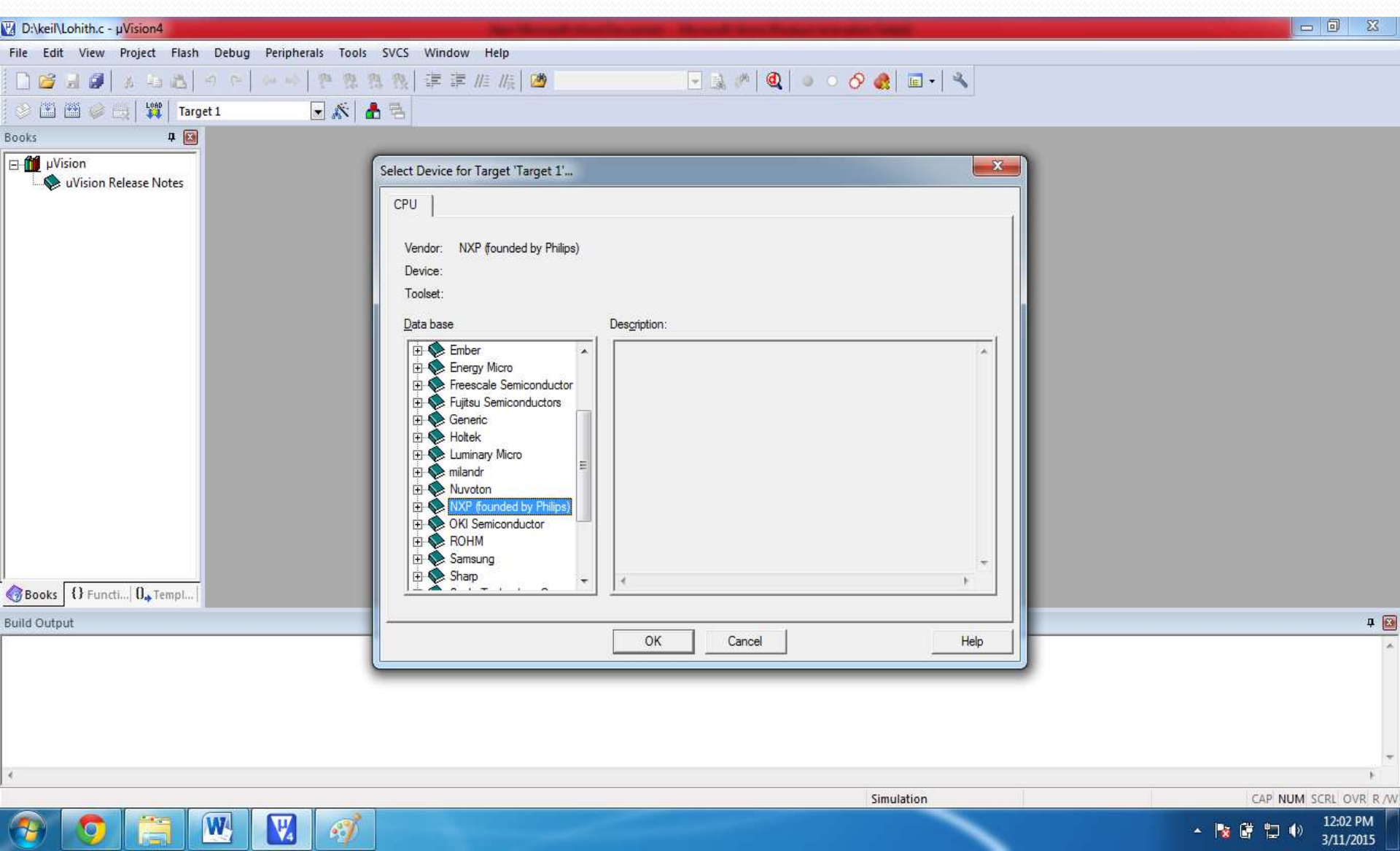


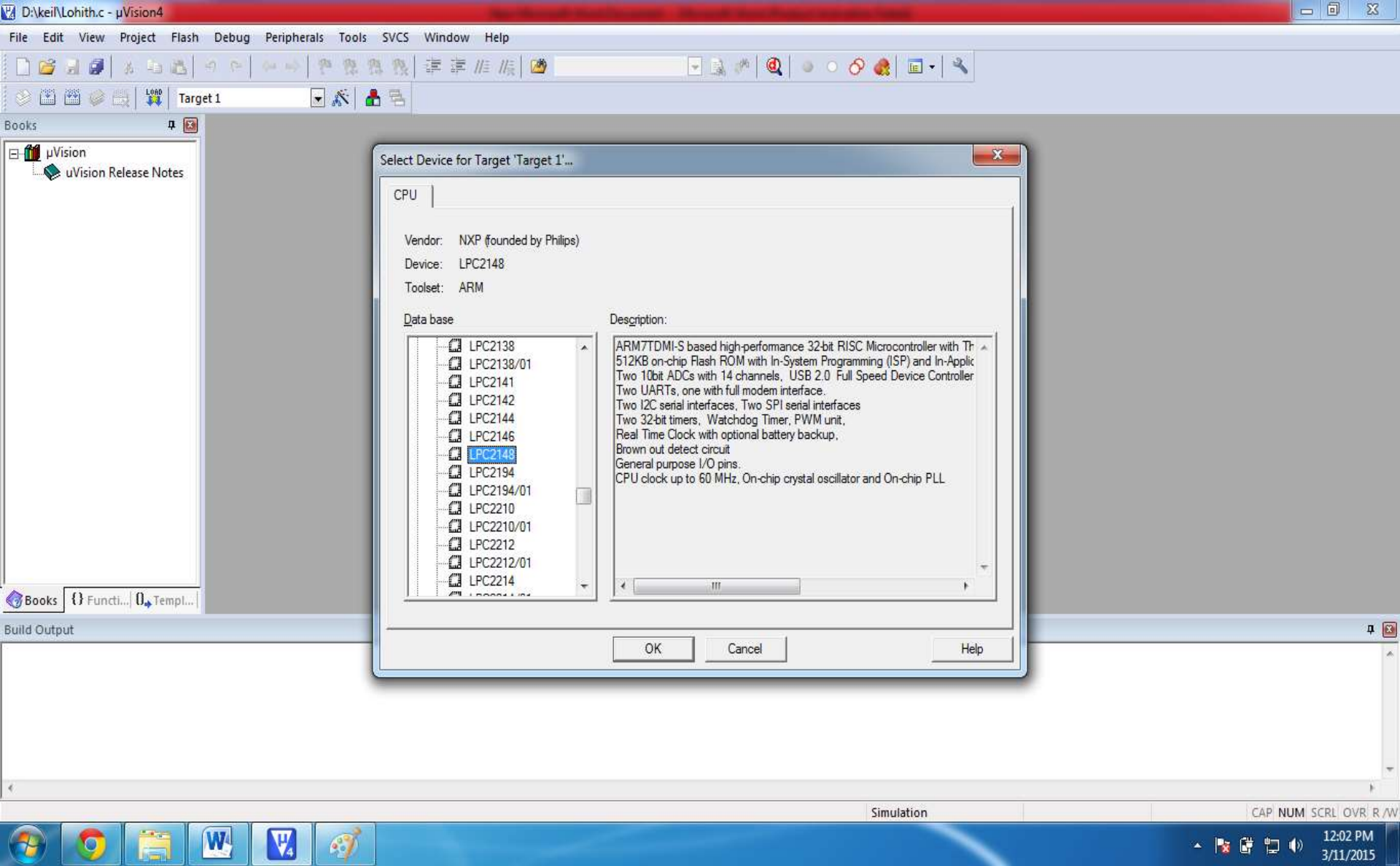
Create a folder in your local drive and in that create new file in the format of .c...

Ex : shiva.c



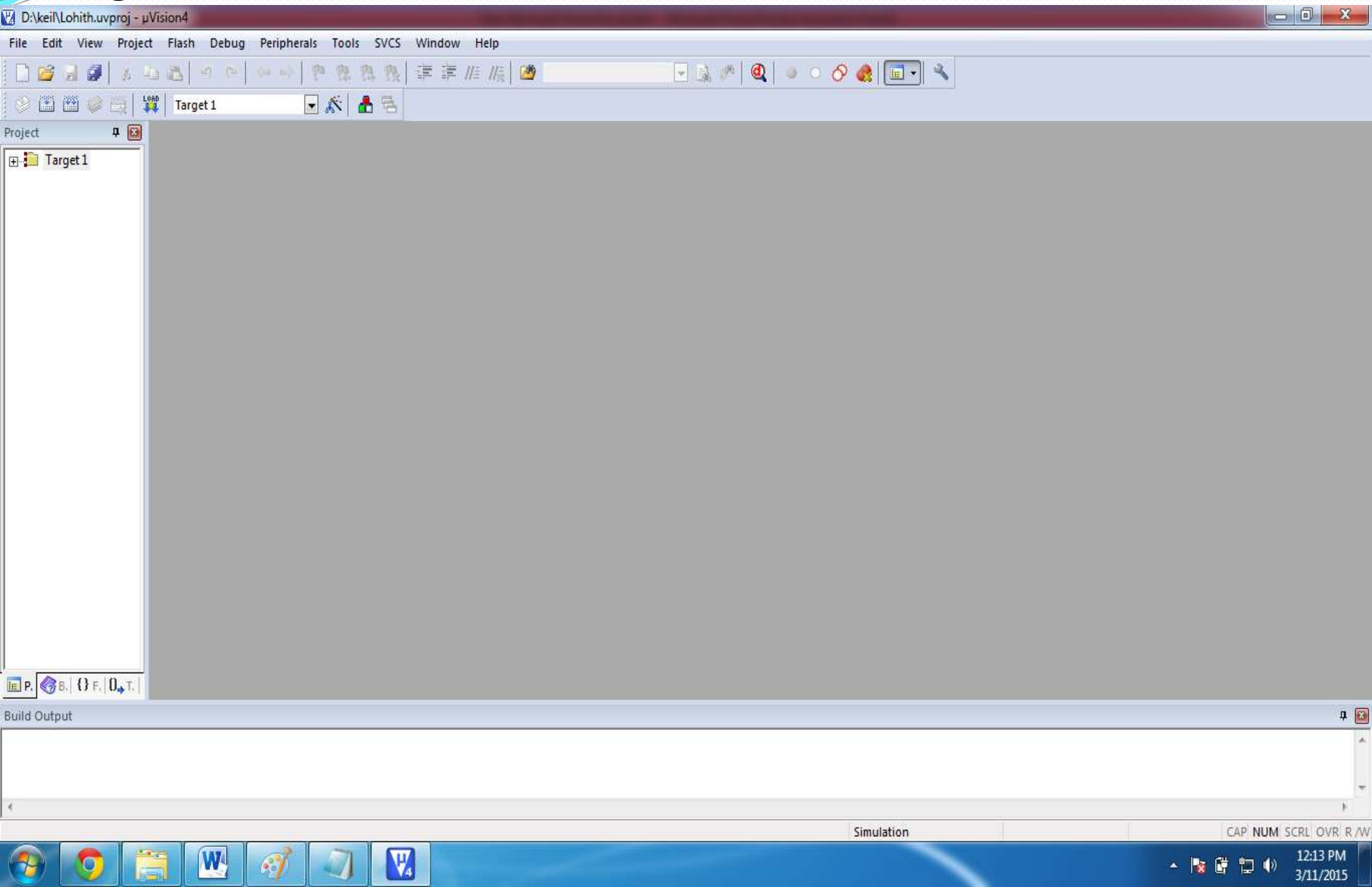
I have created folder in D drive by name of Keil and I have created the file name called shiva.c Once you save the file press yes and you will get one more window in that you have to select the device. The device name is NXP and in that you have to select LCP2148.



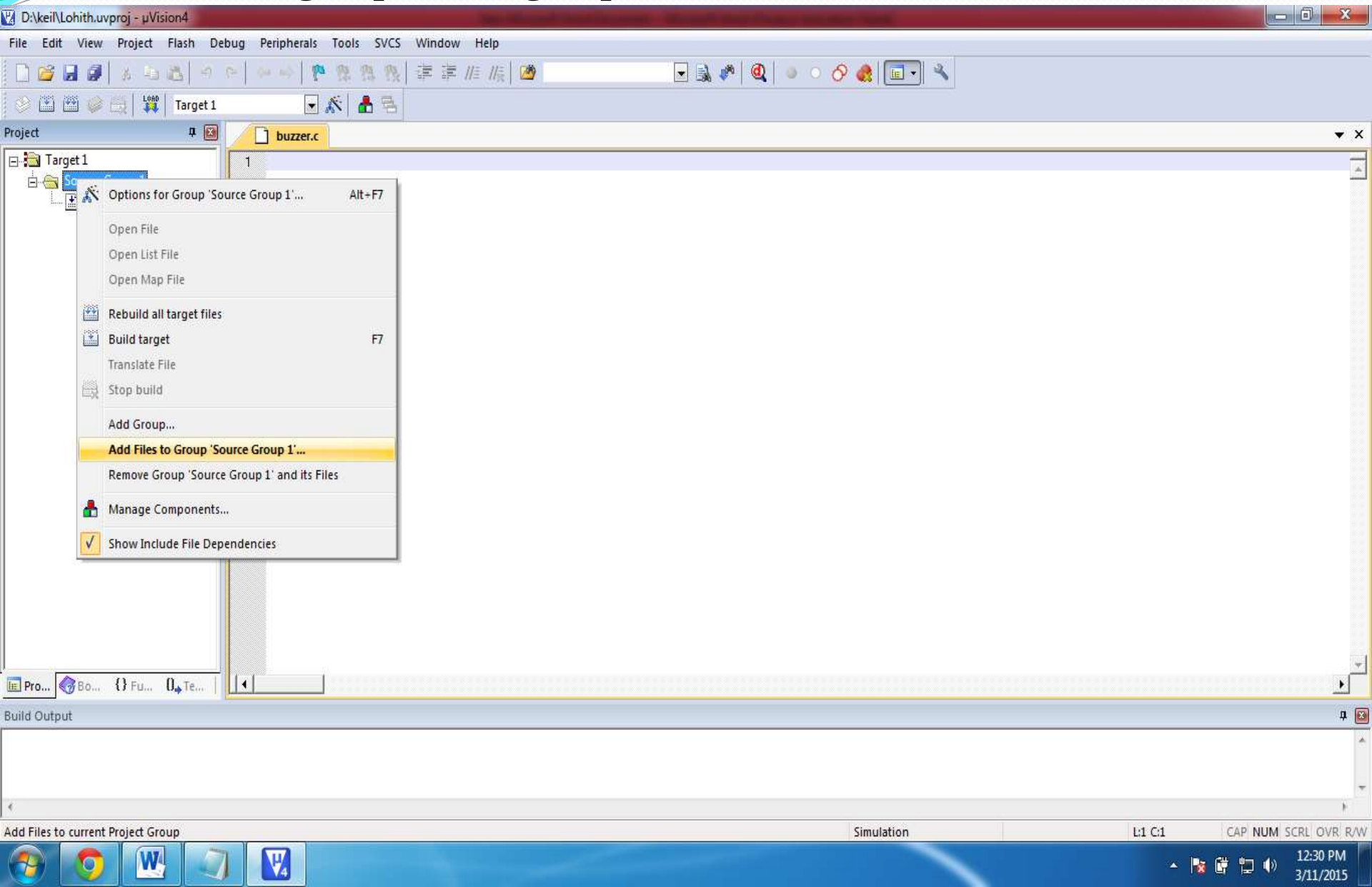


Press ok and you will get one more window to create, pls select yes, because you need to create the startup file. If you not create than program may not be work

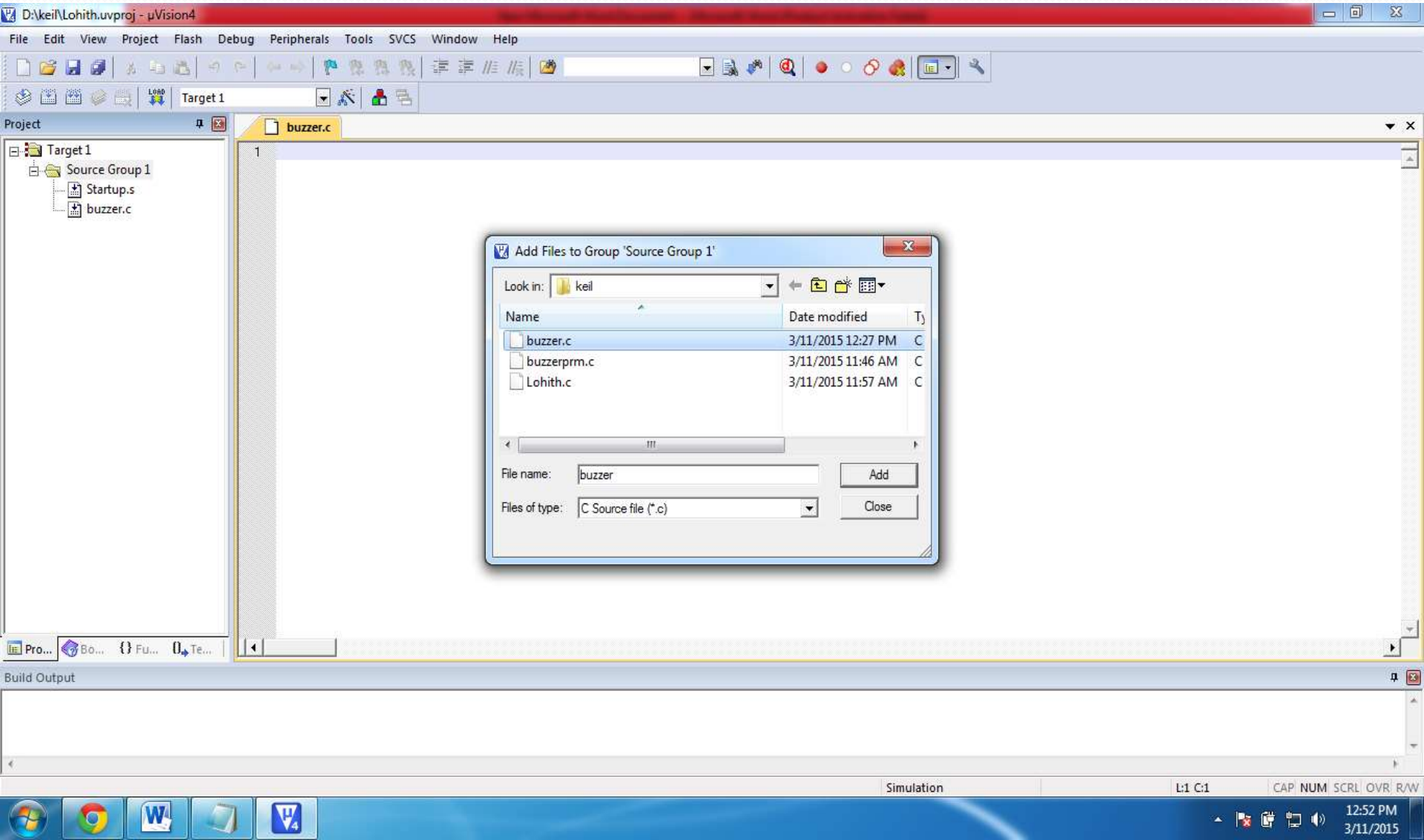
Now go to file and select new and “save as “ pls type your file and save it .
Ok once you save the program file in the left hand side you have file called
Target1

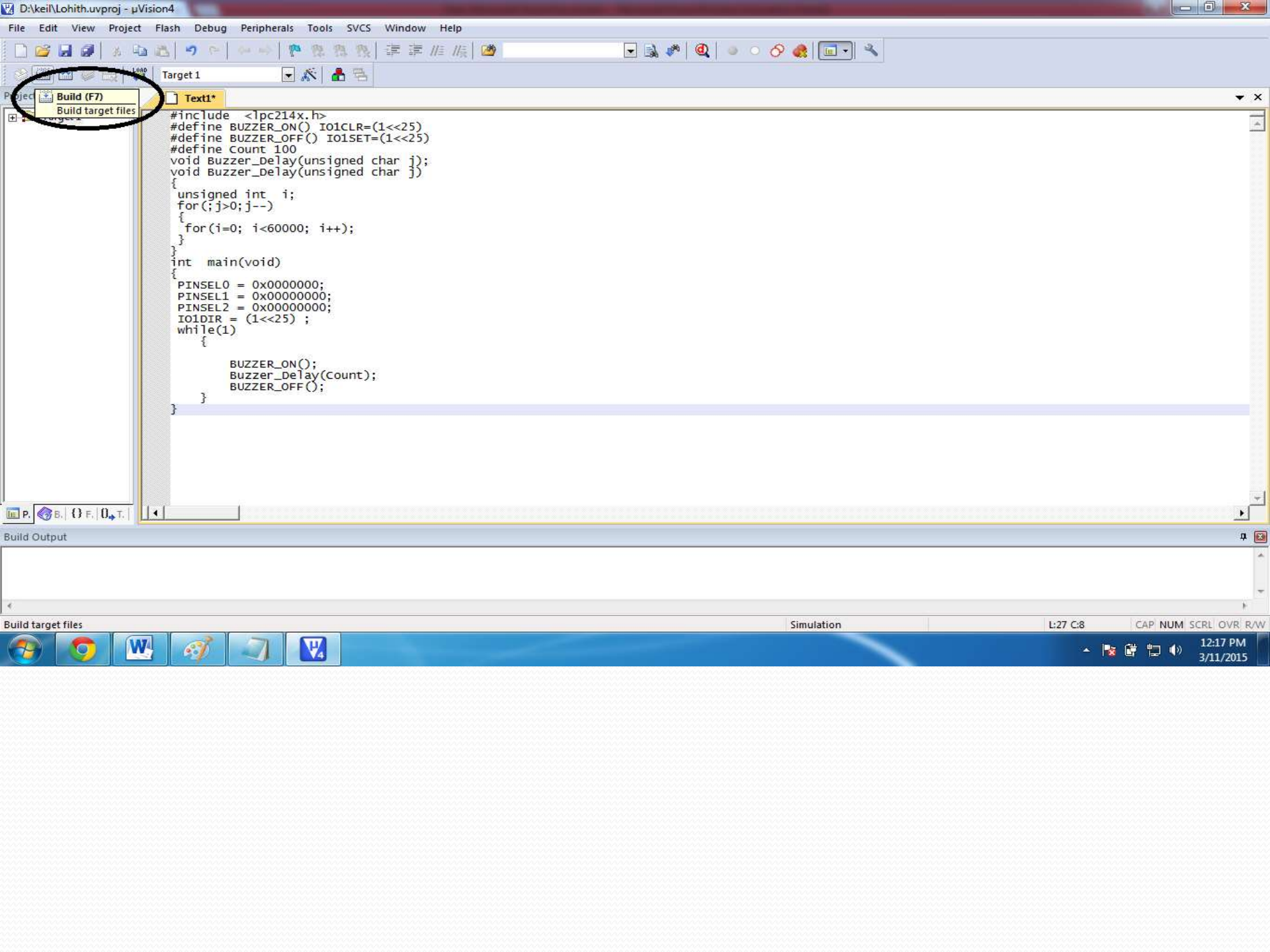


expand the file and you will find one more file called source Group1. Pls expand the Source group 1 and right click on source group file you will get the option called Add file to group Source group1



After you have add file to group source Group1 It will ask you to select you file
Thn select your file from the new tab
add and close the window, now type your program and save it . Once the file is
saved you need to press F7 or press the below mentioned marked button in the
application to build the program.

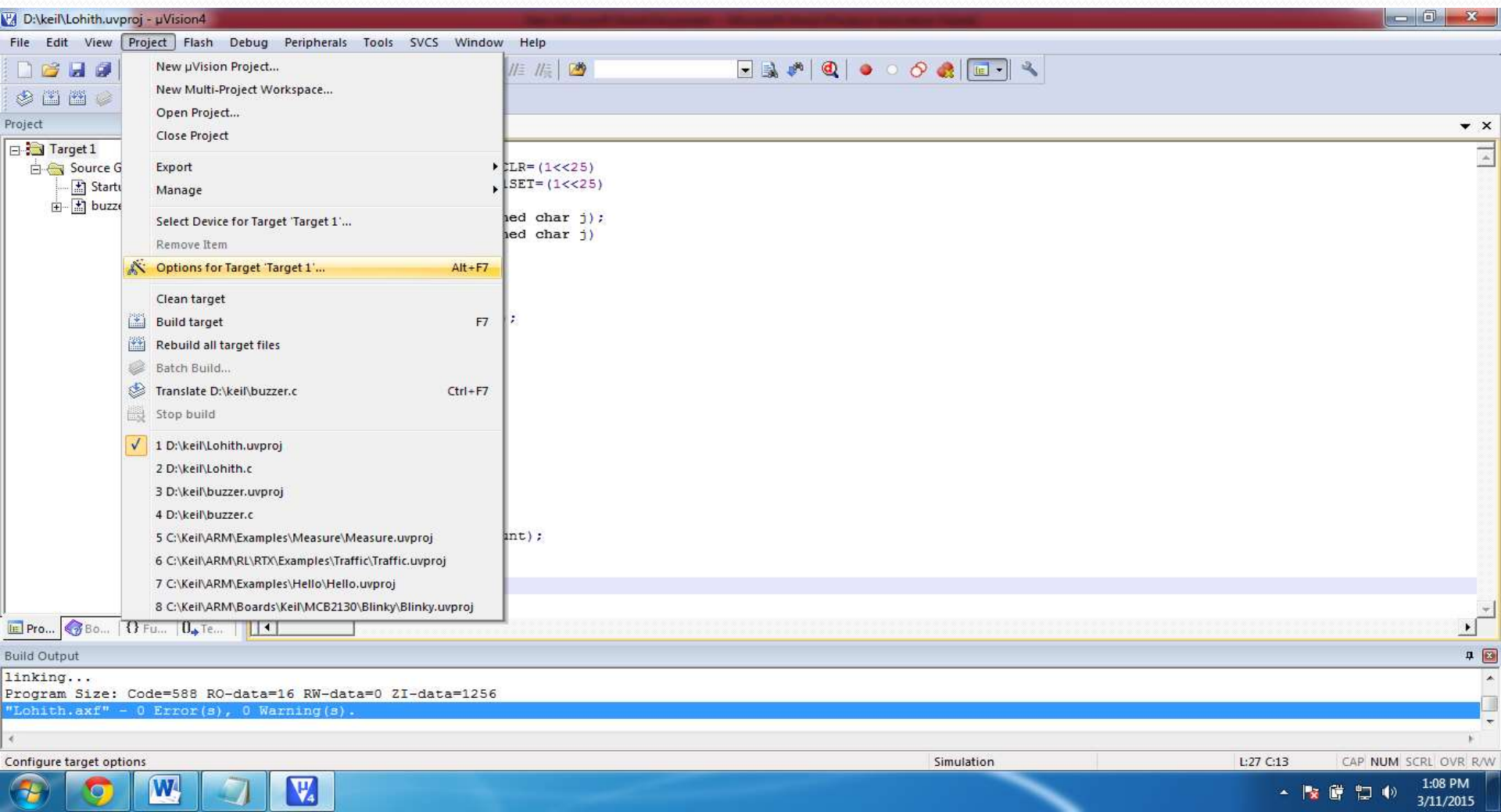


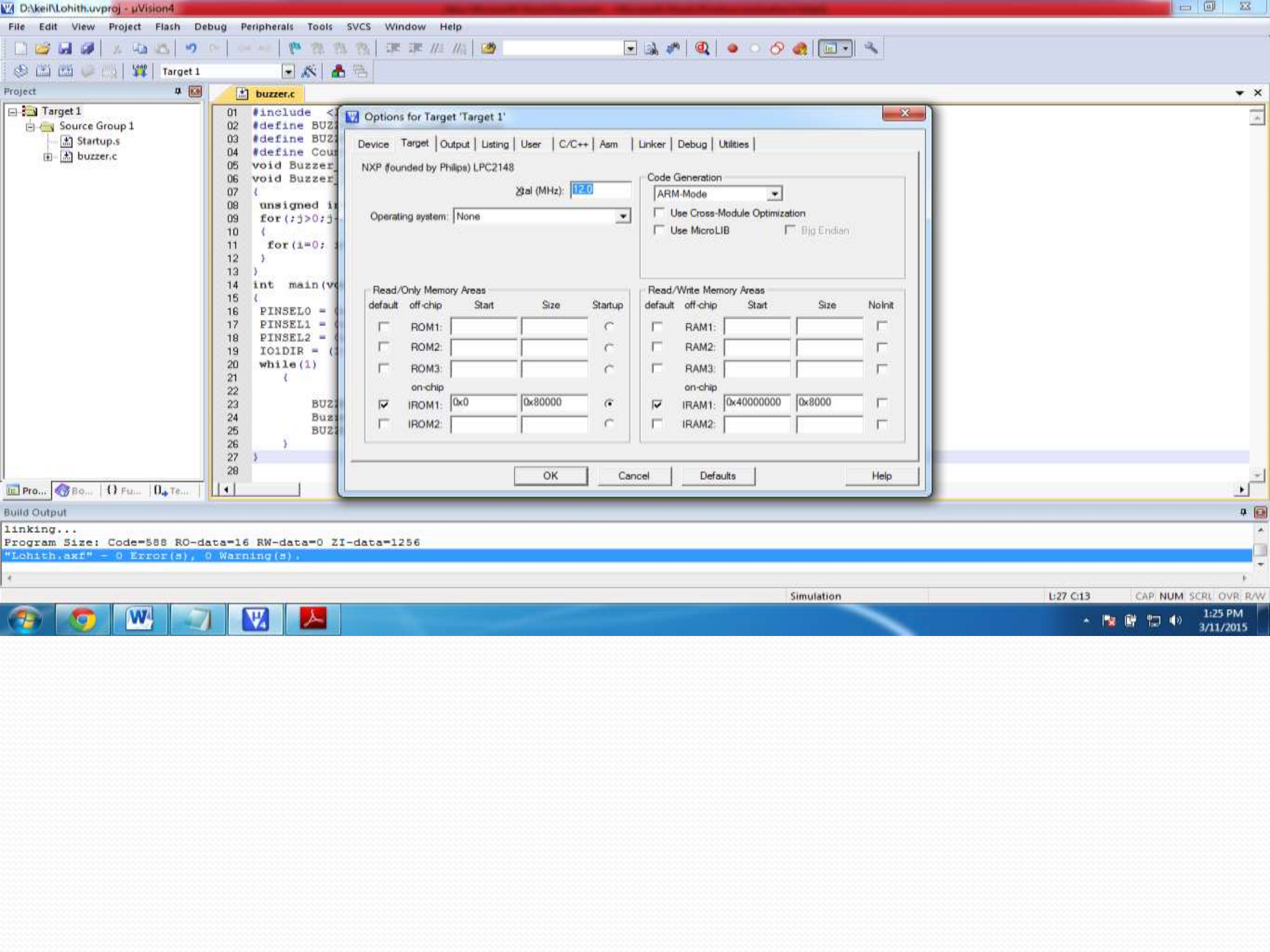


```
Project
  Build (F7)
  Build target files
  Text1*
#include <ipc214x.h>
#define BUZZER_ON() IO1CLR=(1<<25)
#define BUZZER_OFF() IO1SET=(1<<25)
#define count 100
void Buzzer_Delay(unsigned char j);
void Buzzer_Delay(unsigned char j)
{
    unsigned int i;
    for(;j>0;j--)
    {
        for(i=0; i<60000; i++);
    }
}
int main(void)
{
    PINSEL0 = 0x00000000;
    PINSEL1 = 0x00000000;
    PINSEL2 = 0x00000000;
    IO1DIR = (1<<25) ;
    while(1)
    {
        BUZZER_ON();
        Buzzer_Delay(count);
        BUZZER_OFF();
    }
}
```

Pls save the program in same folder which you have created earlier. Once you rebuild the program it will display the warnings and the error, pls clear the errors.

Now you are in the last step that is go to menu and in the menu you have the option called project under the project you have the option called “Option for target source group1” pls select it will take you to one more window called option for target.





Project

- Target 1
 - Source Group 1
 - Startup.s
 - buzzer.c

```

01 #include <
02 #define BUZZER
03 #define BUZZER_FREQ
04 #define BUZZER_DURATION
05 void Buzzer
06 void Buzzer
07 {
08     unsigned int i;
09     for (;j>0;j--)
10     {
11         for (i=0; i<BUZZER_DURATION; i++)
12         {
13             // Buzzer on
14             int main(void)
15             {
16                 PINSEL0 = 0;
17                 PINSEL1 = 0;
18                 PINSEL2 = 0;
19                 IO1DIR = 0;
20                 while (1)
21                 {
22                     // Buzzer on
23                     BUZZER_DURATION;
24                     BUZZER_DURATION;
25                     BUZZER_DURATION;
26                 }
27             }
28         }

```

Options for Target 'Target 1'

Device: Target | Output | Listing | User | C/C++ | Asm | Linker | Debug | Utilities |

NXP (founded by Philips) LPC2148

Operating system: None

Code Generation

ARM-Mode

Use Cross-Module Optimization

Use MicroLIB

Big Endian

Read/Only Memory Areas

default	off-chip	Start	Size	Startup
<input type="checkbox"/>	ROM1:			<input type="checkbox"/>
<input type="checkbox"/>	ROM2:			<input type="checkbox"/>
<input type="checkbox"/>	ROM3:			<input type="checkbox"/>
	on-chip			
<input checked="" type="checkbox"/>	IROM1:	0x0	0x80000	<input checked="" type="checkbox"/>
<input type="checkbox"/>	IROM2:			<input type="checkbox"/>

Read/Write Memory Areas

default	off-chip	Start	Size	NoInit
<input type="checkbox"/>	RAM1:			<input type="checkbox"/>
<input type="checkbox"/>	RAM2:			<input type="checkbox"/>
<input type="checkbox"/>	RAM3:			<input type="checkbox"/>
	on-chip			
<input checked="" type="checkbox"/>	IRAM1:	0x40000000	0x8000	<input type="checkbox"/>
<input type="checkbox"/>	IRAM2:			<input type="checkbox"/>

OK Cancel Defaults Help

Build Output

linking...

Program Size: Code=588 RO-data=16 RW-data=0 ZI-data=1256

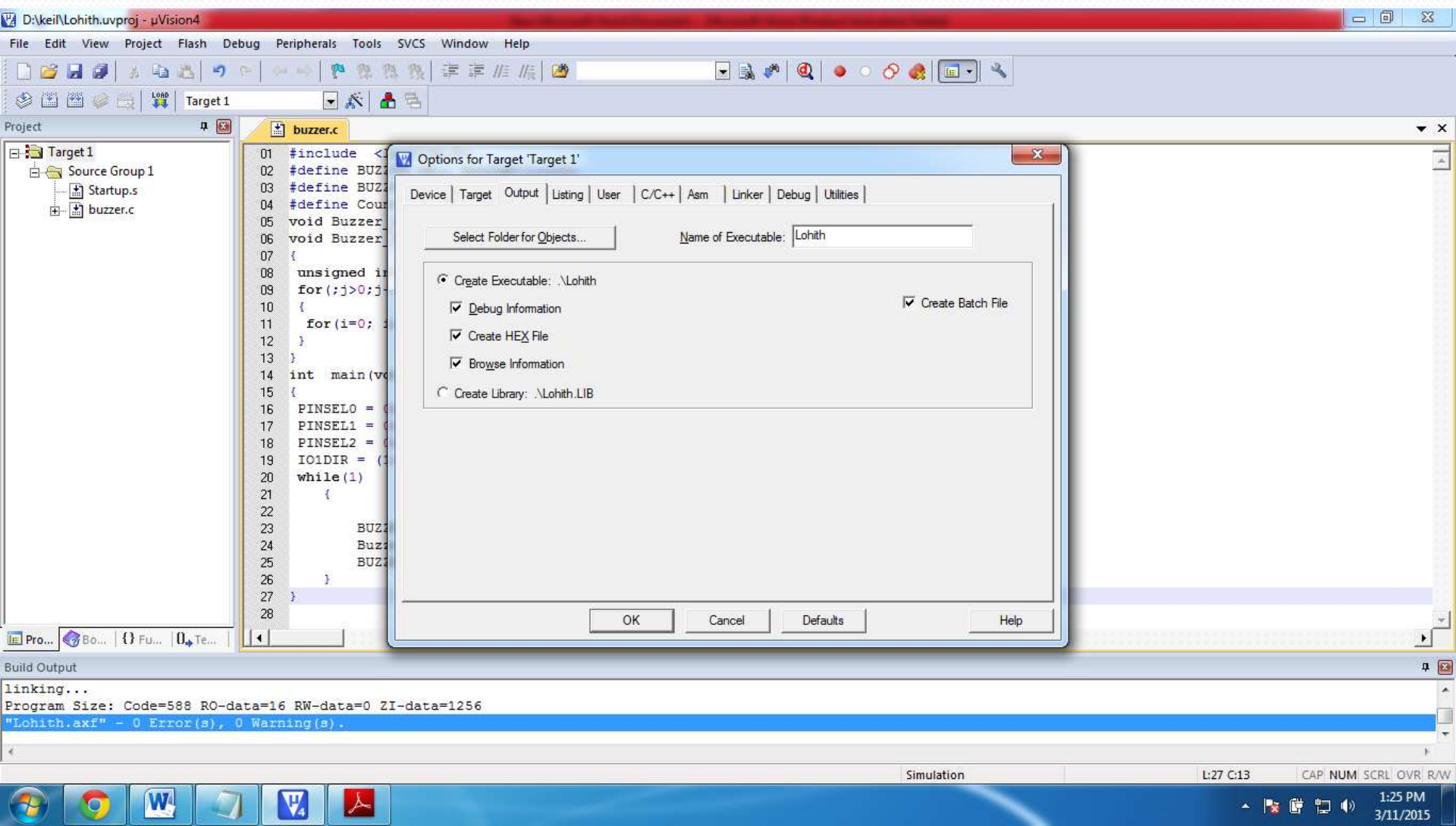
"Lohith.axf" - 0 Error(s), 0 Warning(s).

Simulation

L:27 C:13

CAP NUM SCRL OVR RAW

In the option target menu you have to select output option in the menu to create HEX file , in the output Pls tick the HEX file option and tick the create batch file option also and say ok.



And in the same menu you have to enable one more tab that is memory layout, in the same menu pls go to linker option in that you have memory layout tab pls enable the memory layout than press ok

The screenshot displays the uVision4 IDE interface. The main window shows the source code for 'buzzer.c' with the following content:

```
01 #include <lpc214x.h>
02 #define BUZZER_ON() IO1CLR=(1<<25)
03 #define BUZZER_OFF() IO1SET=(1<<25)
04 #define Count 100
05 void Buzzer_Delay(unsigned char j);
06 void Buzzer_De
07 {
08 unsigned int
09 for (;j>0;j--)
10 {
11 for(i=0; i<6
12 }
13 }
14 int main(void
15 {
16 PINSEL0 = 0x0
17 PINSEL1 = 0x0
18 PINSEL2 = 0x0
19 IO1DIR = (1<<
20 while(1)
21 {
22
23 BUZZER
24 Buzzer
25 BUZZER
26 }
27 }
28
```

The 'Options for Target' dialog box is open, with the 'Linker' tab selected. The 'Use Memory Layout from Target Dialog' checkbox is checked. Other options include:

- Make RW Sections Position Independent
- Make RO Sections Position Independent
- Don't Search Standard Libraries
- Report 'might fail' Conditions as Errors

Additional fields include R/O Base (0x00000000), R/W Base (0x40000000), and a 'disable Warnings' field. The 'Scatter File' field is empty, and the 'Misc controls' field is also empty. The 'Linker control string' field contains the following text:

```
-cpu ARM7TDMI -o -strict -scatter "Lohith.sct"
-autoat -summary_stderr -info summarysizes -map -xref -callgraph -symbols
```

The dialog box has 'OK', 'Cancel', 'Defaults', and 'Help' buttons at the bottom.

Finally pls cross check the HEX file is created or not, if not created than you have made some mistake in the above mentioned steps, pls cross check once.

If you found the HEX file in your created folder than that will be the input file for one more application called Flash Magic.

The Flash magic is the one of the application which will take the HEX file as a input and which will give the output for buzzer device.

Now let us see how the flash magic will work ? Go to flash magic application

File ISP Options Tools Help



Step 1 - Communications

Select... LPC2148

Flash Bank:

COM Port: COM 1

Baud Rate: 9600

Interface: None (ISP)

Oscillator (MHz): 12

Step 2 - Erase

Erase block 0 (0x000000-0x000FFF)
Erase block 1 (0x001000-0x001FFF)
Erase block 2 (0x002000-0x002FFF)
Erase block 3 (0x003000-0x003FFF)
Erase block 4 (0x004000-0x004FFF)
Erase block 5 (0x005000-0x005FFF)

- Erase all Flash+Code Rd Prot
 Erase blocks used by Hex File

Step 3 - Hex File

Hex File: D:\keil\buzzer.hex

Browse...

Modified: Wednesday, March 11, 2015, 11:49:04 AM

[more info](#)

Step 4 - Options

- Verify after programming
 Fill unused Flash
 Gen block checksums
 Execute
 Activate Flash Bank

Step 5 - Start!

Start

On-Line training classes for microcontrollers and embedded networking and
Internetworking

www.esacademy.com/faq/classes

Not programmed

0



Pls select the input as per the picture shown above and connect the buzzer board to P C and port should be com1.

Now browse the .Hex file and press start button once the program is uploaded to the board in the left hand side corner you will get the indication called finish , once it finish pl remove the com1 cable from the board as soon as you removed the cable from the board buzzer make sound and as per your program buzzer will work.