

# Roboversity

---



# HOW TO USE KEIL SOFTWARE FOR CREATING HEX FILE TO THE CORRESPONDING C FILE

## KEIL µVISION:

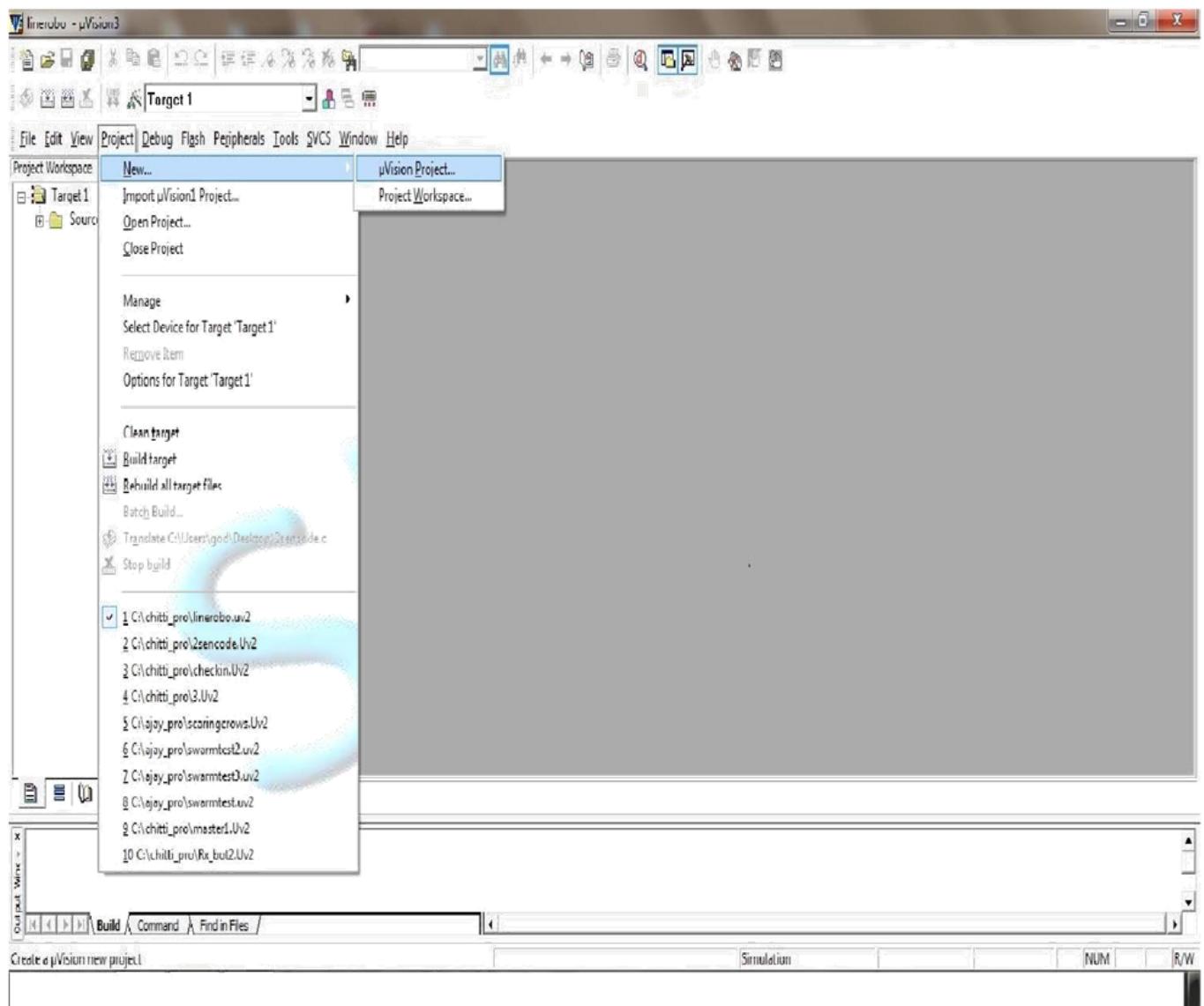
Keil µvision is a highly compatible programming tool used for coding, compiling and debugging any of C, Assembly language programs and also for translating that program code file into a HEX file. Microcontrollers can receive/transmit data only in digital (HEX) language and so it can understand the files which are in HEX format only. Hence this sort of conversion is done in order to burn them to the microcontroller. So the necessity of this software is increasing with the advancement of the Embedded Technology. It can also be used for project management, simulation and handling embedded programs.

This software also simplifies the conversion task and enables easy access of the program code, as it makes the debugging process easy by pointing the position of errors and warnings after the compiling process. This tool has an additional option for analyzing the simulation of the C code by fixing the desired input and checking the output. The Full version of this tool enables to access and convert program codes whose size is more than 2KB. Numerous example programs are also available in the µvision tool, which help in assisting beginners.

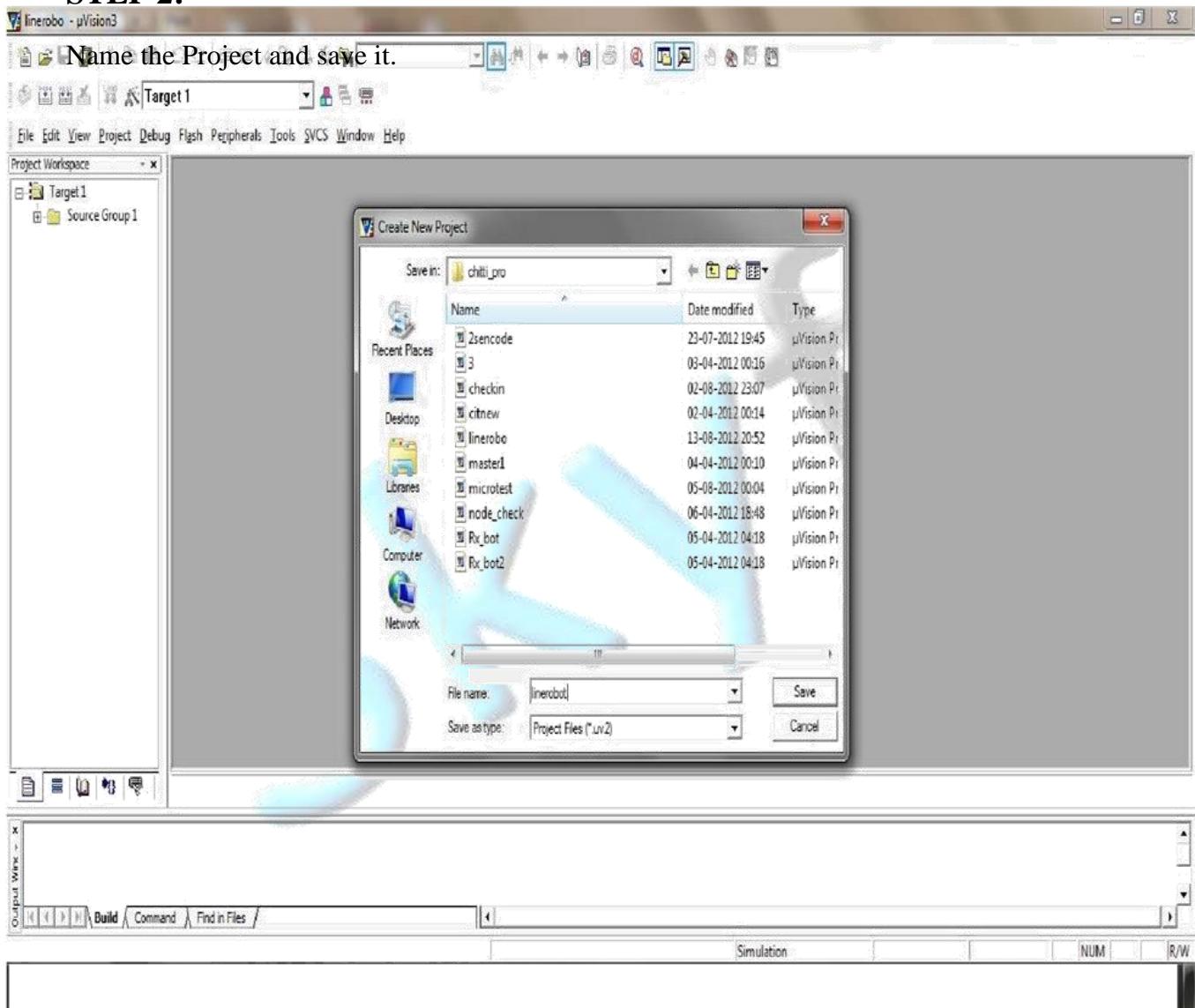


## STEP 1:

Create a new µVision Project for performing the process of converting a C file into Hex file.



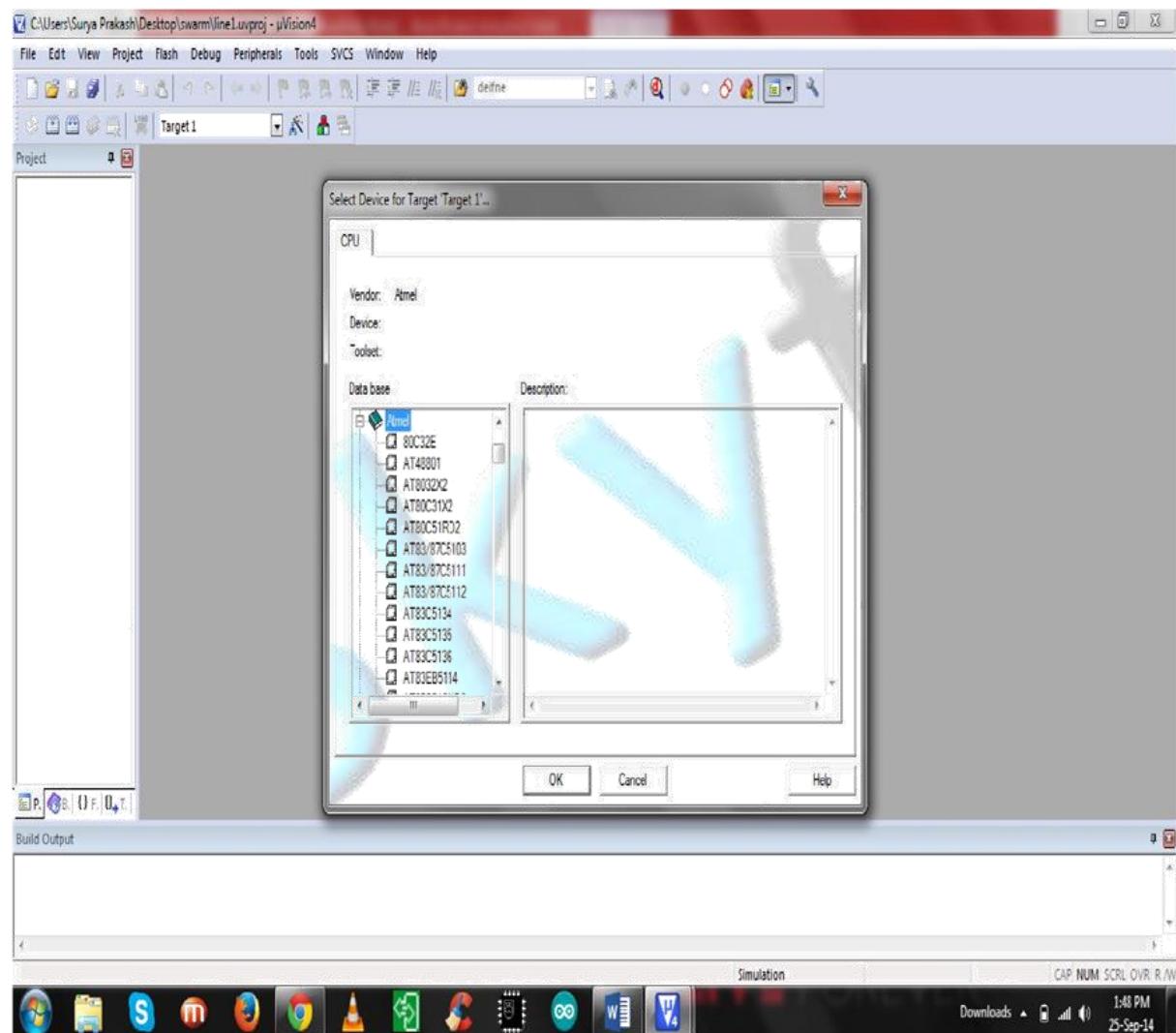
## STEP 2:



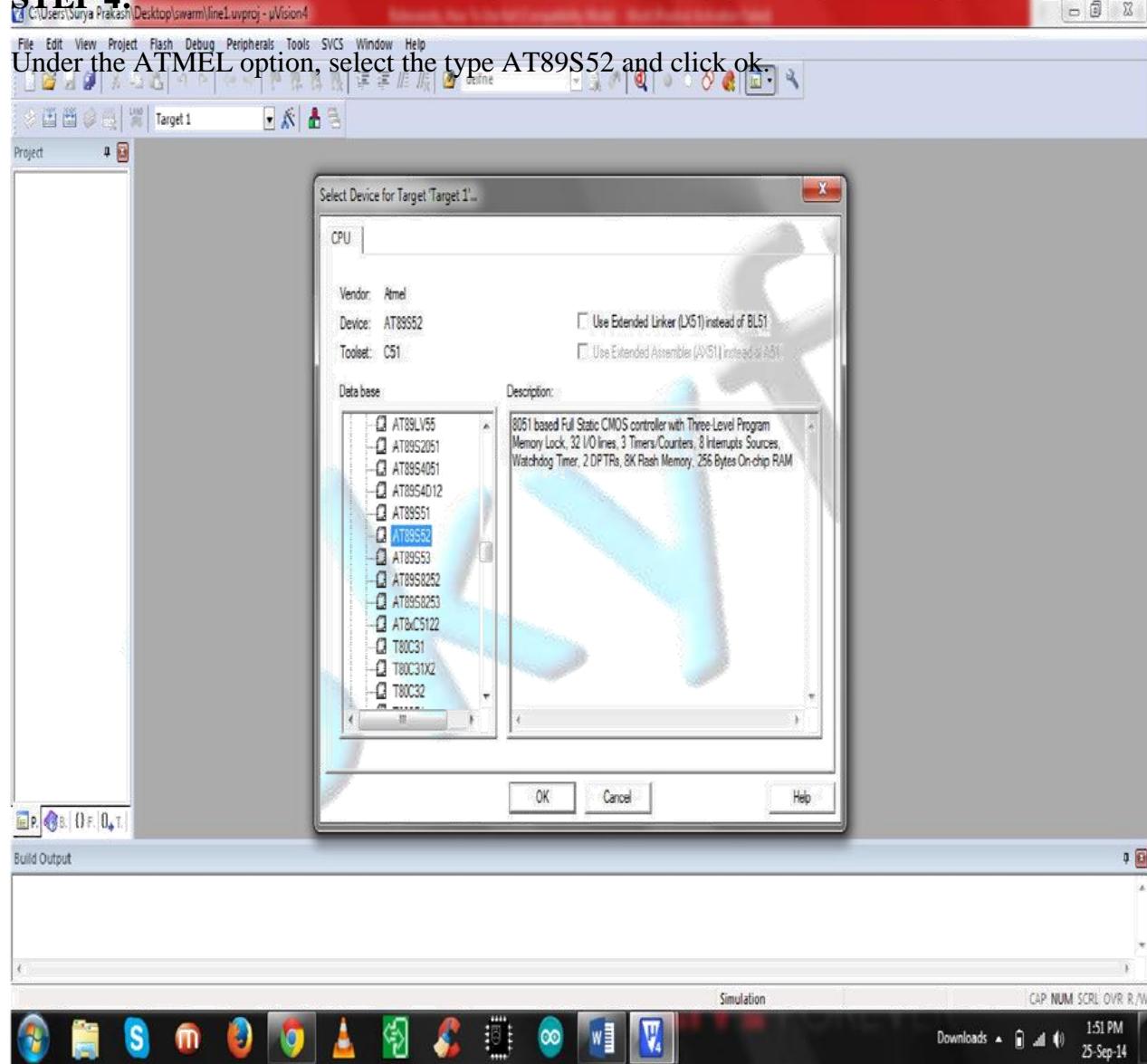


### STEP 3:

Select the name of the 8051 manufacturer, which is being used in the PCB. It is ATMEL in our case.



#### STEP 4:

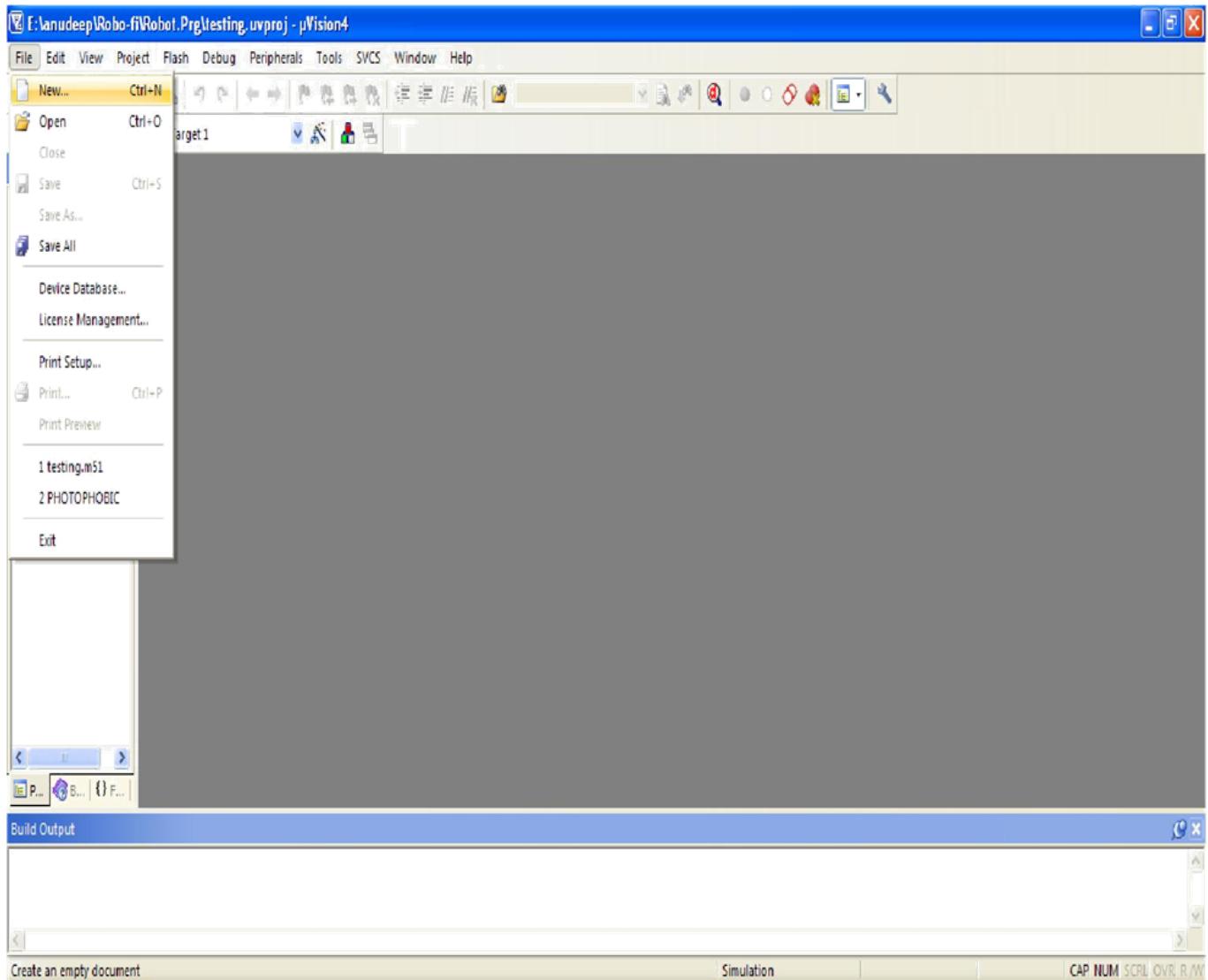


#### NOTE:

- A dialog box appears on the screen regarding copying the microcontroller code to the project folder. Click ‘Yes’.
- If another dialog box, mentioning that the file is already present in the folder, appears, click ‘Yes’.

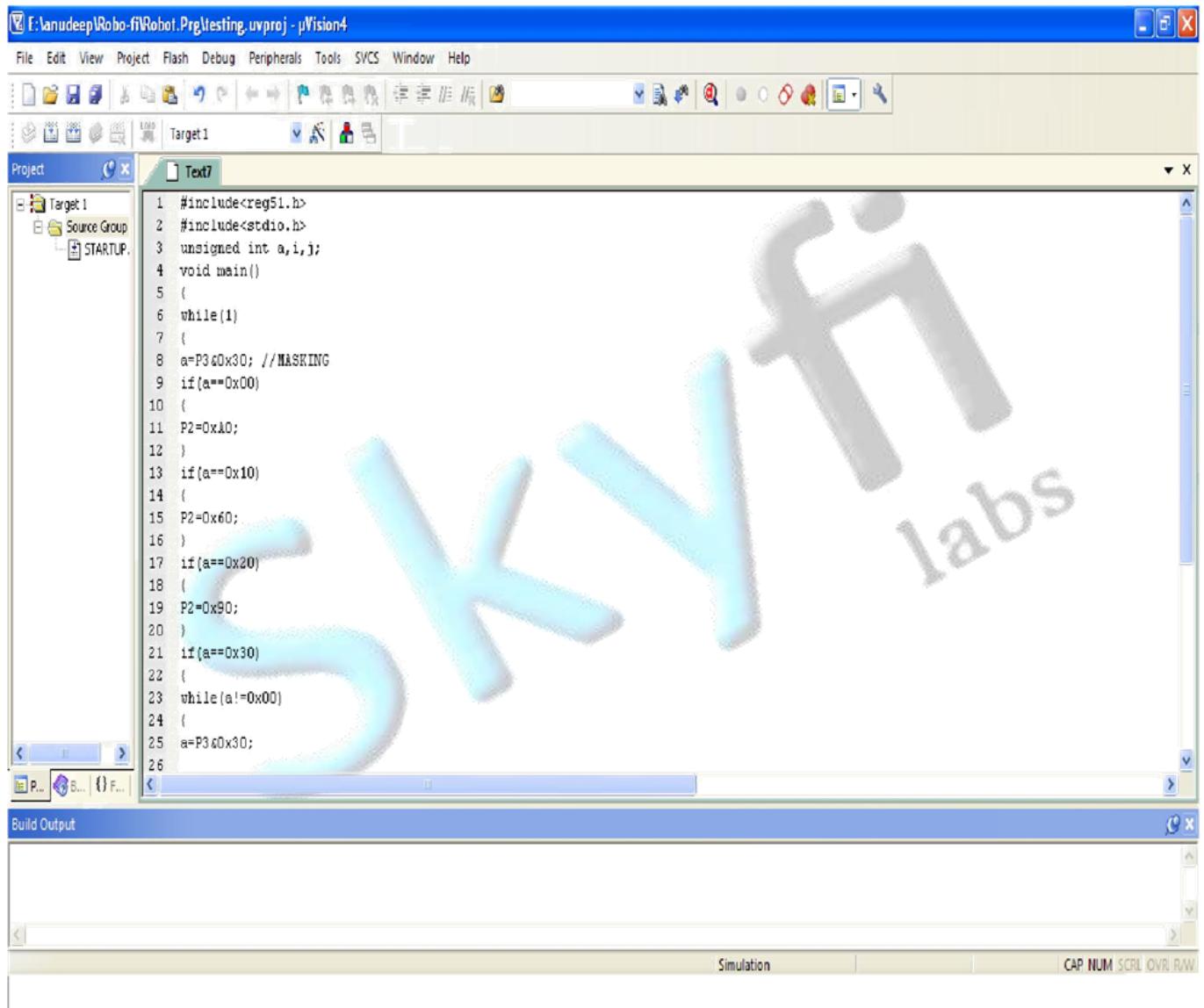
## STEP5:

To write a program code, select **File** and then **New** option



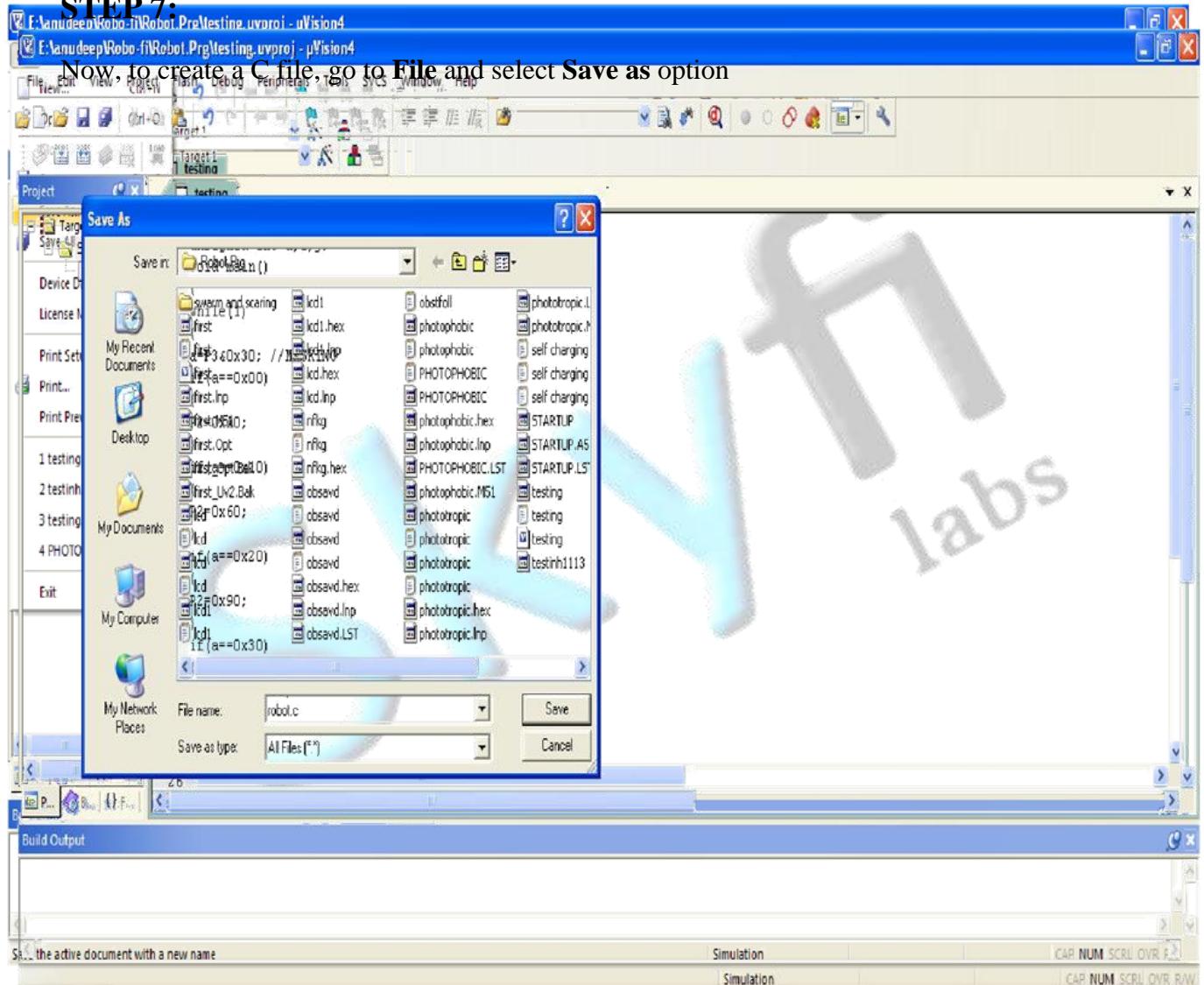
## STEP 6:

Enter the program code in the editor page as shown below



```
1 #include<reg51.h>
2 #include<stdio.h>
3 unsigned int a,i,j;
4 void main()
5 {
6     while(1)
7     {
8         a=P3&0x30; //MASKING
9         if(a==0x00)
10        {
11             P2=0x10;
12         }
13         if(a==0x10)
14        {
15             P2=0x60;
16         }
17         if(a==0x20)
18        {
19             P2=0x90;
20         }
21         if(a==0x30)
22        {
23             while(a!=0x00)
24            {
25                 a=P3&0x30;
26             }
27         }
28     }
29 }
```

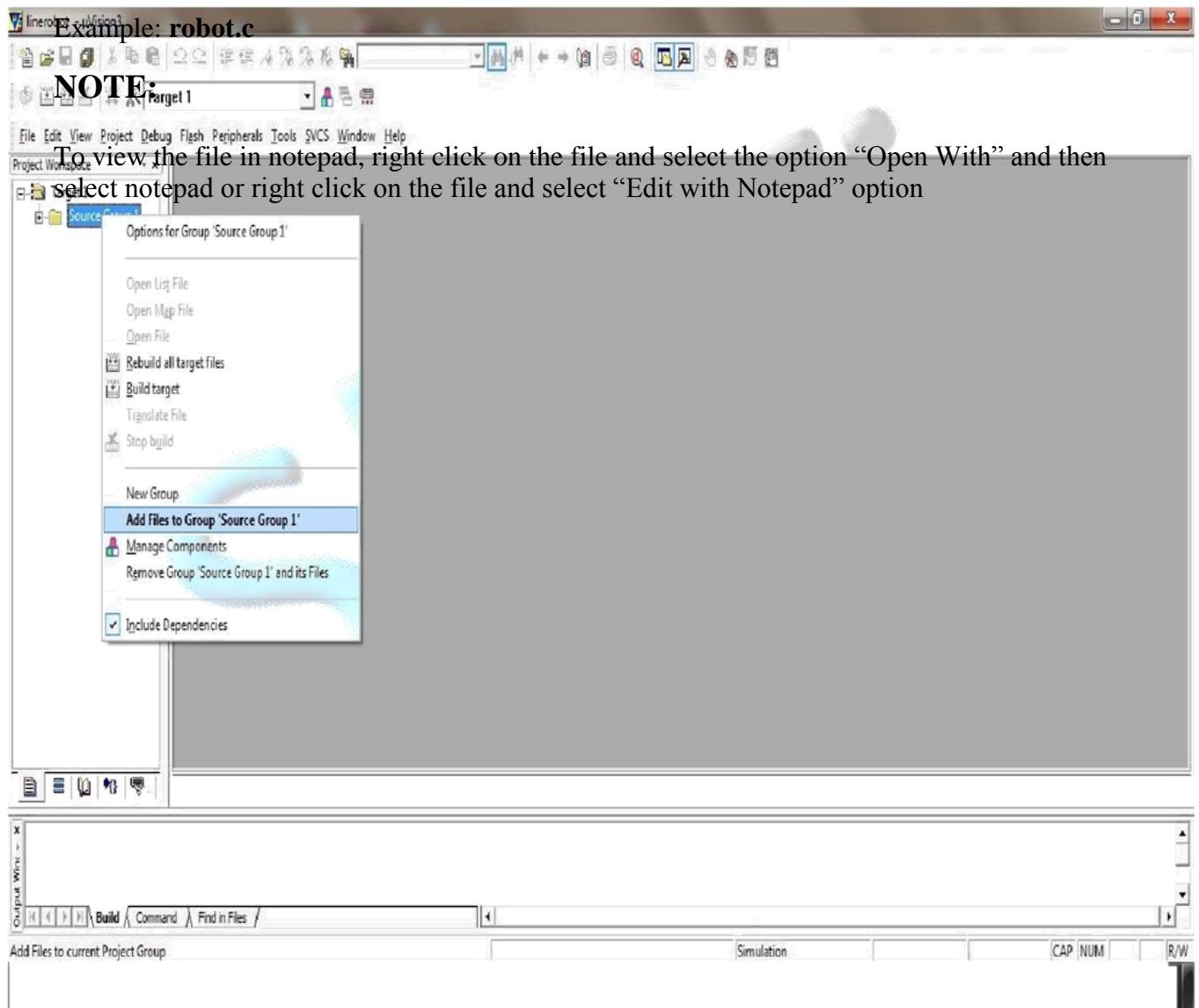
### STEP 7:



Now, to create a C file, go to File and select Save as option

## STEP 8:

Type the file name with an extension of “.c”



## **STEP 9:**

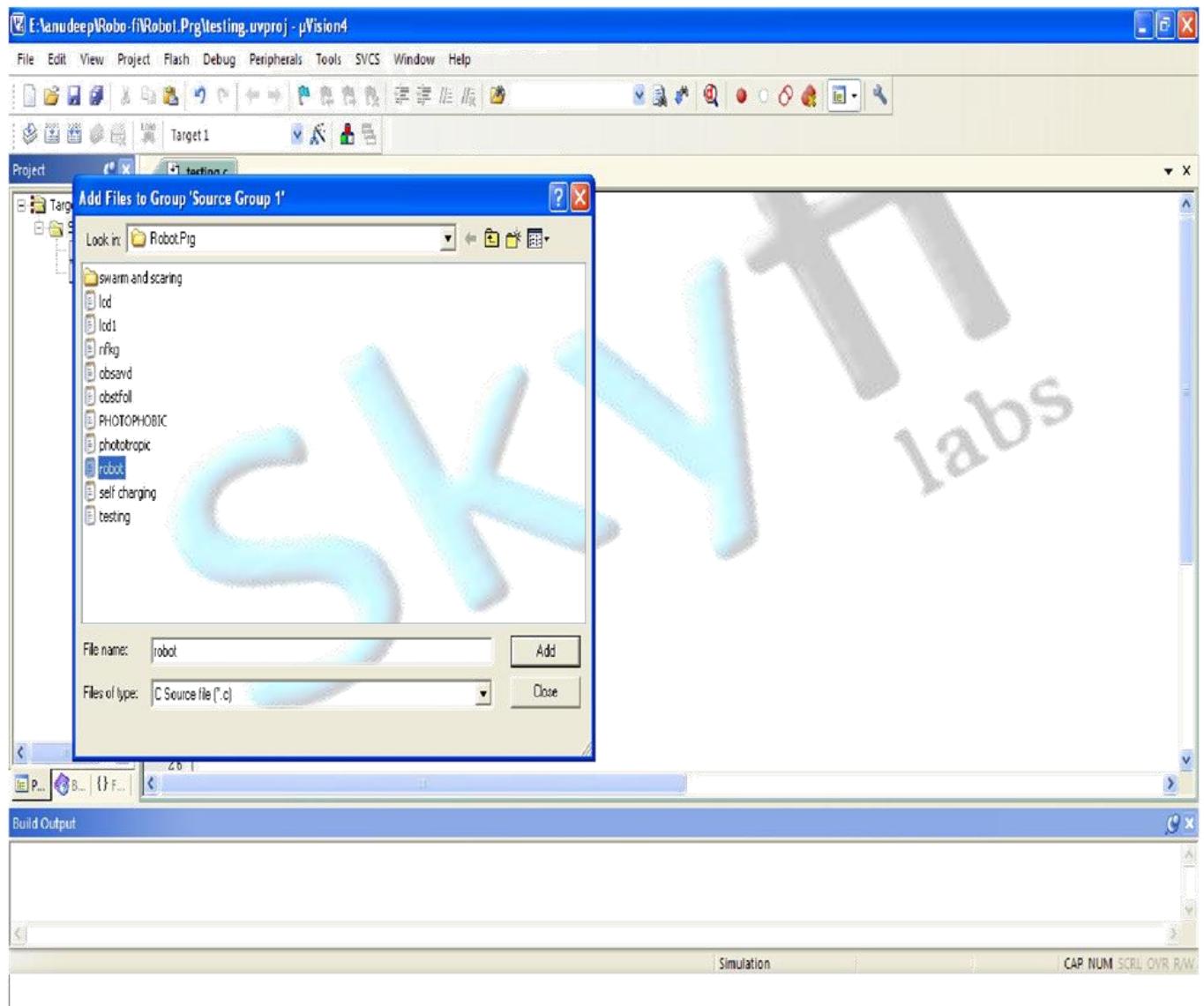
Add the C file to the Source group,

Go to **Source Group 1**, which will appear on the left toolbar when “**Target 1**” is clicked;  
Right click on it and select option **Add Files to Group ‘Source Group1’**.



## STEP 10:

Select the C file from the location where you have saved it and click on **Add** option and **Close**.

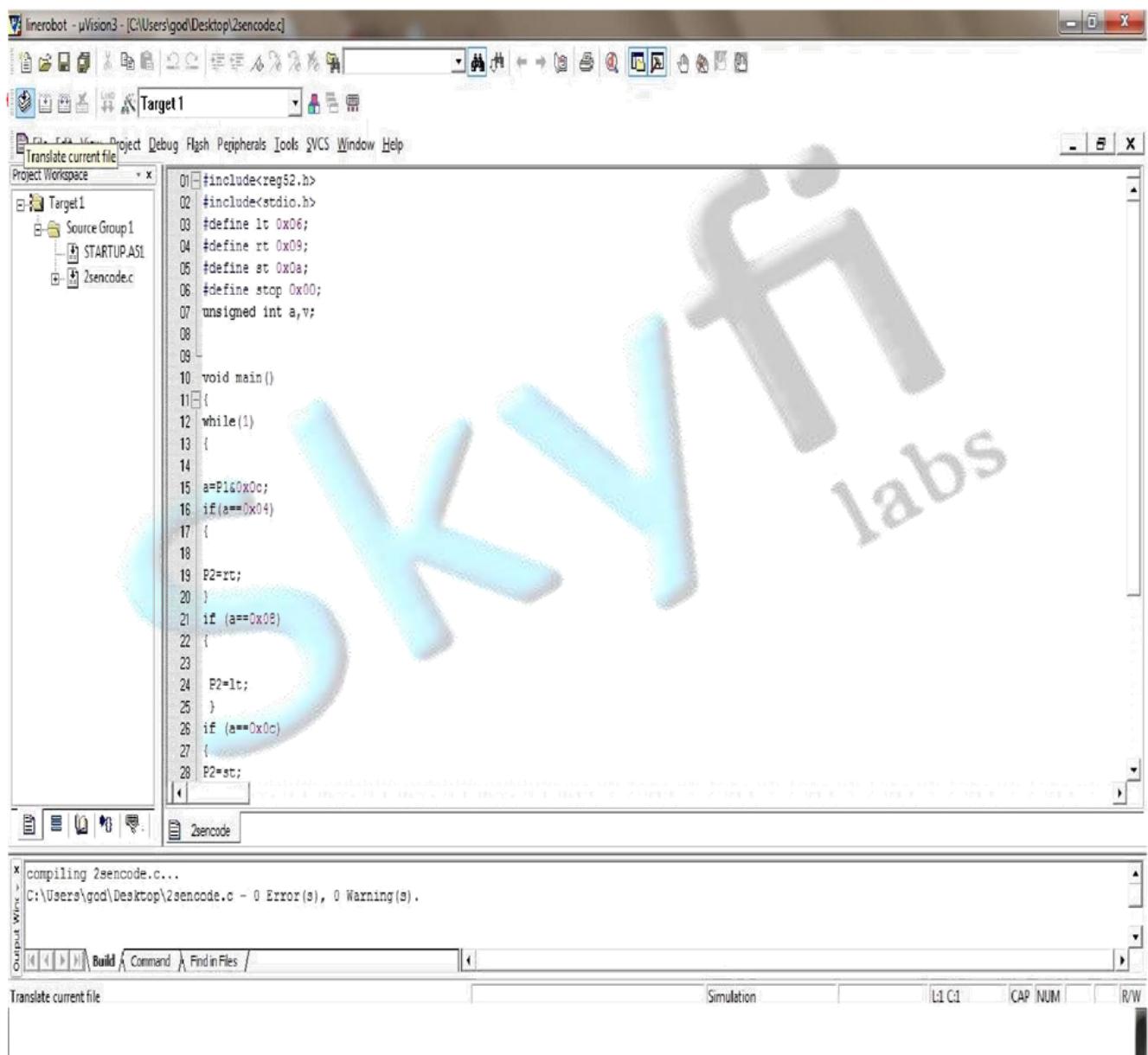


## NOTE:

Make sure that the file just added is displayed in the drop down menu of 'Source Group 1'.

## STEP 11:

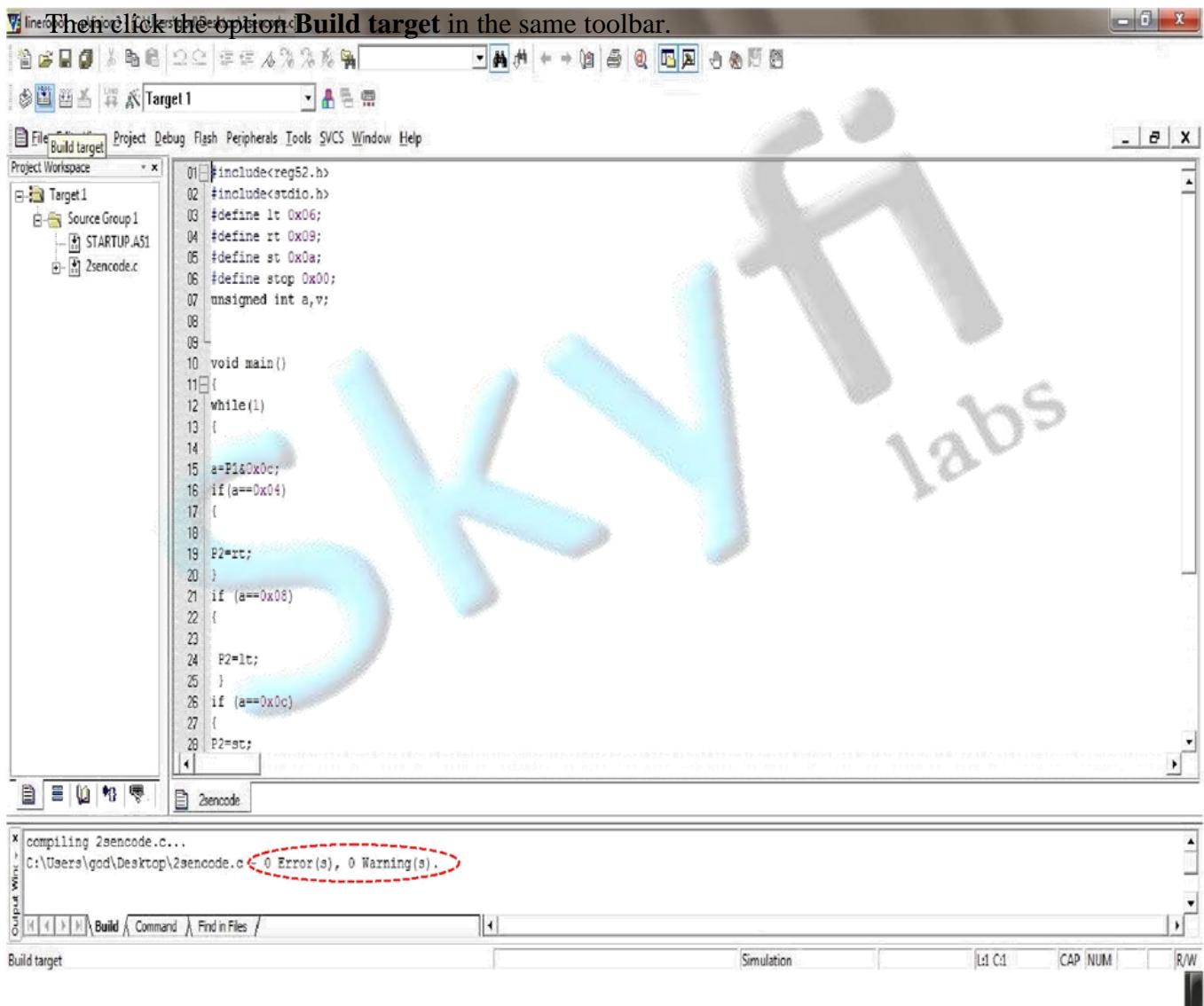
Click the option **Translate current file** in the left corner of the toolbar.



## NOTE:

- The above message regarding errors and warnings will be displayed after each step. Make sure there are no errors/warnings at each step.
- Make Error Free Program for proper execution and compilation.

## STEP 14:





**STEP 15:**

Then click on the button **Rebuild all target files** as shown in the figure in the same toolbar.

```

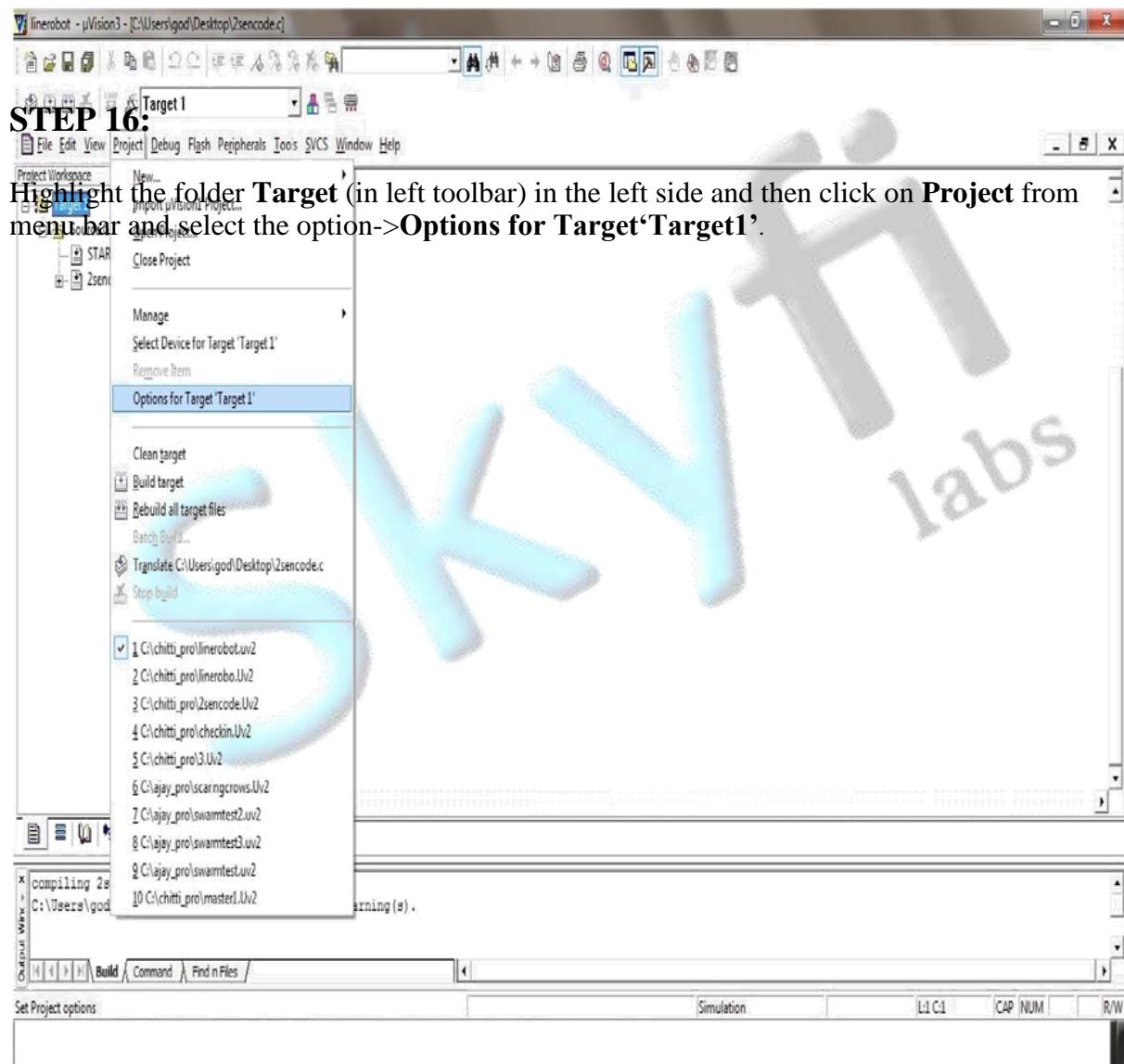
12 while(1)
13 {
14
15     a=P1&0x0c;
16
17     if(a==0x04)
18     {
19
20         P2=rt;
21
22     if (a==0x08)
23     {
24
25         P2=lt;
26
27     if (a==0x0c)
28     {
29         P2=st;
30
31     if (a==0x00)
32     {
33
34         P2=lt;
35
36     }
37 }
38

```

compiling 2sencode.c...  
C:\Users\god\Desktop\2sencode.c - 0 Error(s), 0 Warning(s).

Output Window: Build, Command, Find in Files

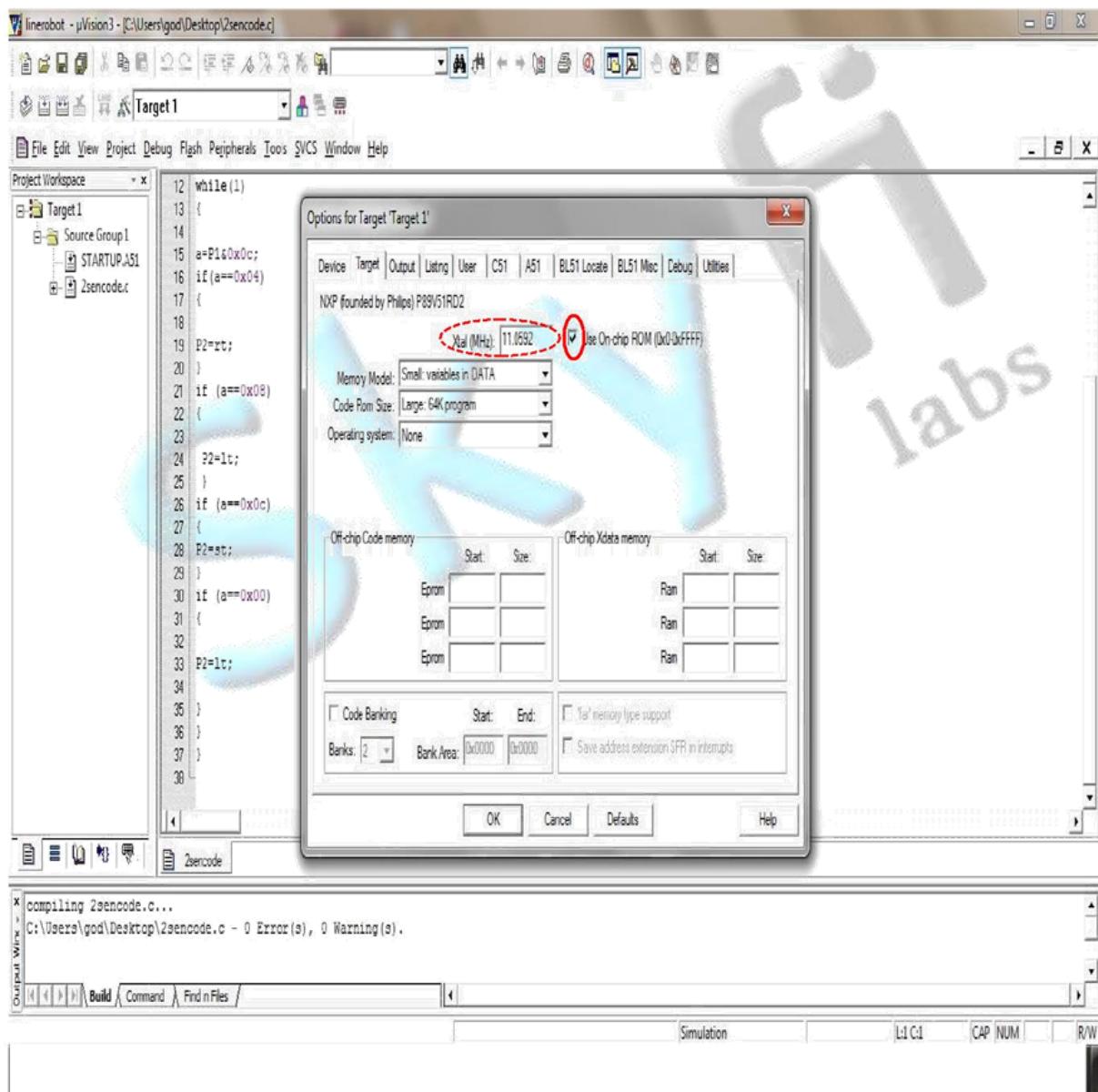
Rebuild all target files      Simulation      L1 C1      CAP NUM      R/W





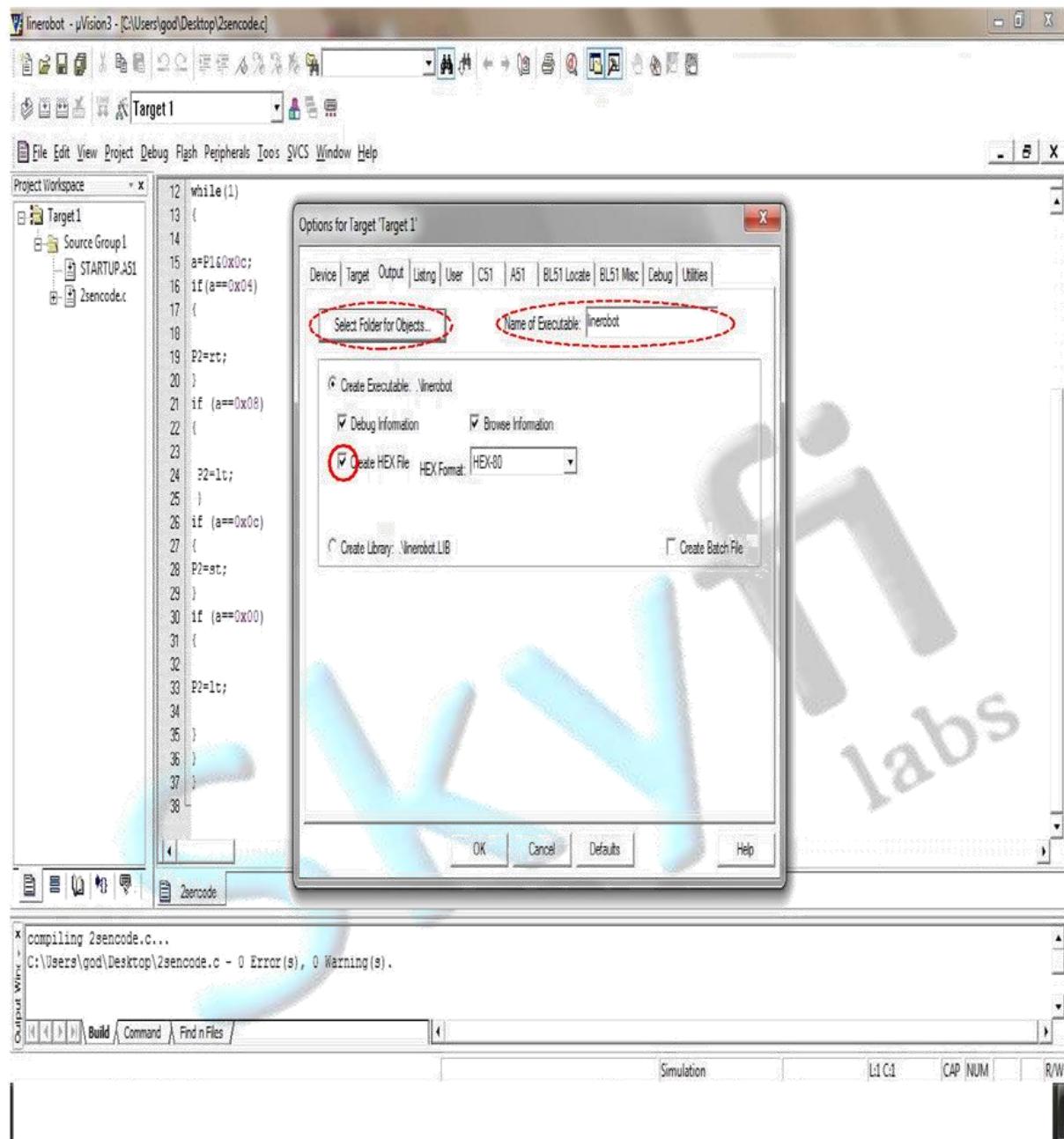
## STEP 17:

Set the Crystal frequency value in **XTAL (MHz)**: as 11.0592 and check the box beside **Use On-chip ROM**.



## STEP 18:

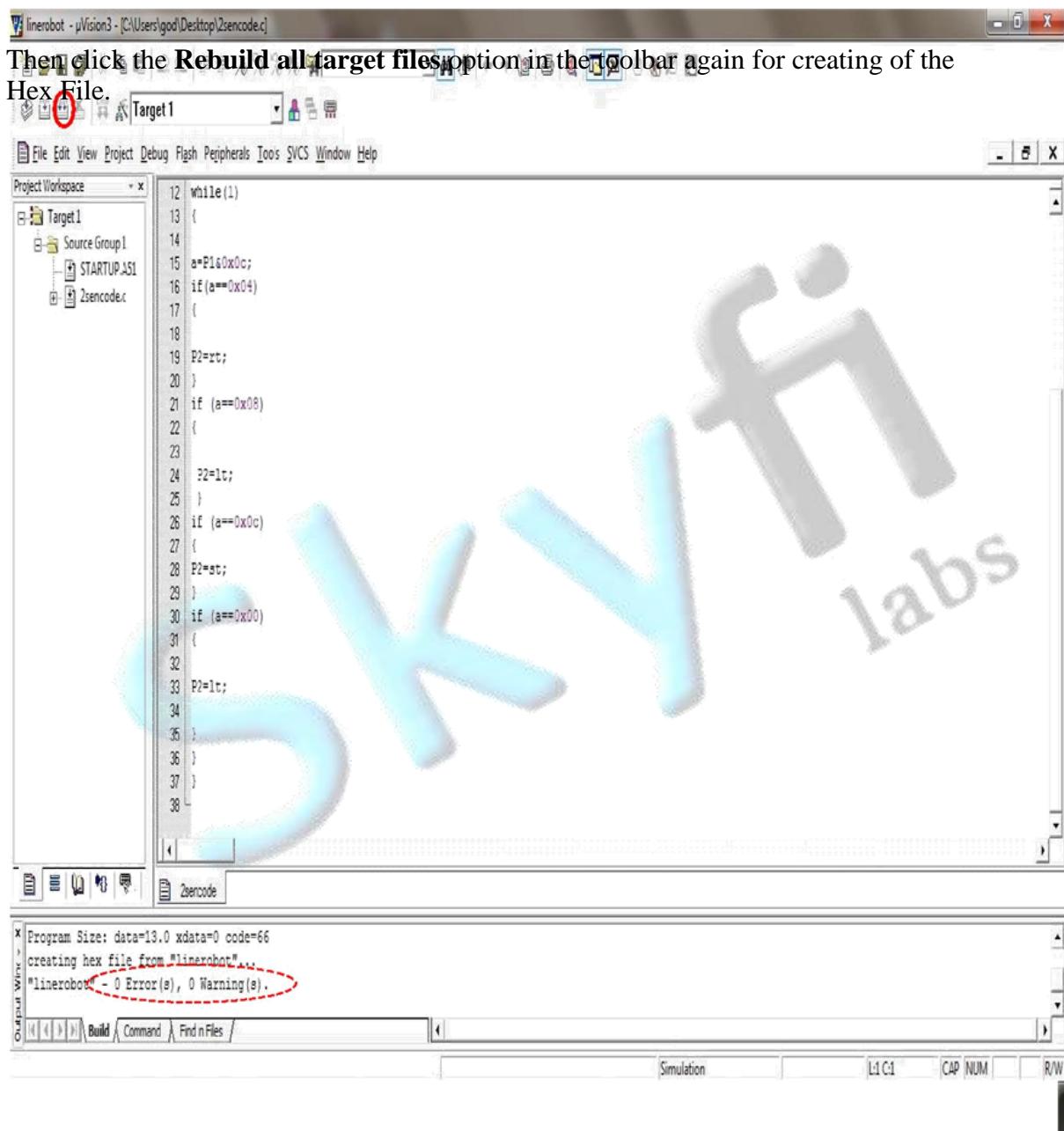
Then go to output option from toolbar of same popup menu and highlight **Create HEX File** Option as shown below and enter the HEX-file name in **Name of Executable** field and also select the destination folder in **Select Folder for Objects...**option.



## NOTE:

Make sure you note the destination folder of the Hex file and name of the executable.

## STEP 19:



## STEP 20:

After finishing the above process, verify the creation of the Hex file in the status window below the workspace.

The screenshot shows a software interface with a menu bar (File, Edit, View, Project, Debug, Flash, Peripherals, Tools, SVCS, Window, Help) and a toolbar. On the left is a 'Project Workspace' window showing a project structure with 'Target1' and 'Source Group1' containing 'STARTUP.AS1' and '2sencode.c'. The main area is a code editor with the following C code:

```

12 while(1)
13 {
14
15     a=P1&0x0c;
16     if(a==0x04)
17     {
18
19         P2=rt;
20     }
21     if (a==0x08)
22     {
23
24         P2=lt;
25     }
26     if (a==0x0c)
27     {
28
29         P2=st;
30     }
31     if (a==0x00)
32     {
33
34         P2=lt;
35     }
36 }
37
38

```

Below the code editor is a status window with the following message:

```

Program Size: data 10.0 xdata=0 code=66
creating hex file from "linerobot"...
"linerobot" - 0 Error(s), 0 Warning(s)

```

A red oval highlights the status message in the status window.

- Now the HEX file is created and ready to be flashed onto the microcontroller using Flash Magic Software (Refer to ‘How to Detect COM Port’ and ‘How to Use Flash Magic’ documents).

