GECA

(An Autonomous Institute of Government of Maharashtra)

Government College of Engineering, Aurangabad

शासकीय अभियांत्रिकी महाविद्यालय, औरंगाबाद

In pursuit of Technical Excellence

महाराष्ट्र शासन

Department of Electronics and Telecommunication Engineering

ET3014: Lab Embedded Systems

Name : Ritik Nandanwar

Roll No. : BE18F04F033

Batch : T1

Experiment 1: Introduction to IDE software and universal programmer to program microcontrollers.

GOVERNMENT COLLEGE OF ENGINEERING, AURNAGABAD

ET3014 E&TC DEPARTMENT EMBEDDED SYSTEM LAB

EXPERIMENT NO. 1

Aim: Introduction to IDE software and universal programmer to program microcontrollers.

Apparatus: Keil µ vision 5

Theory:

Getting started with ARM LPC2148 using Keil uVision IDE

There are various development environments available in the market for ARM processors. Some of these are mentioned below:

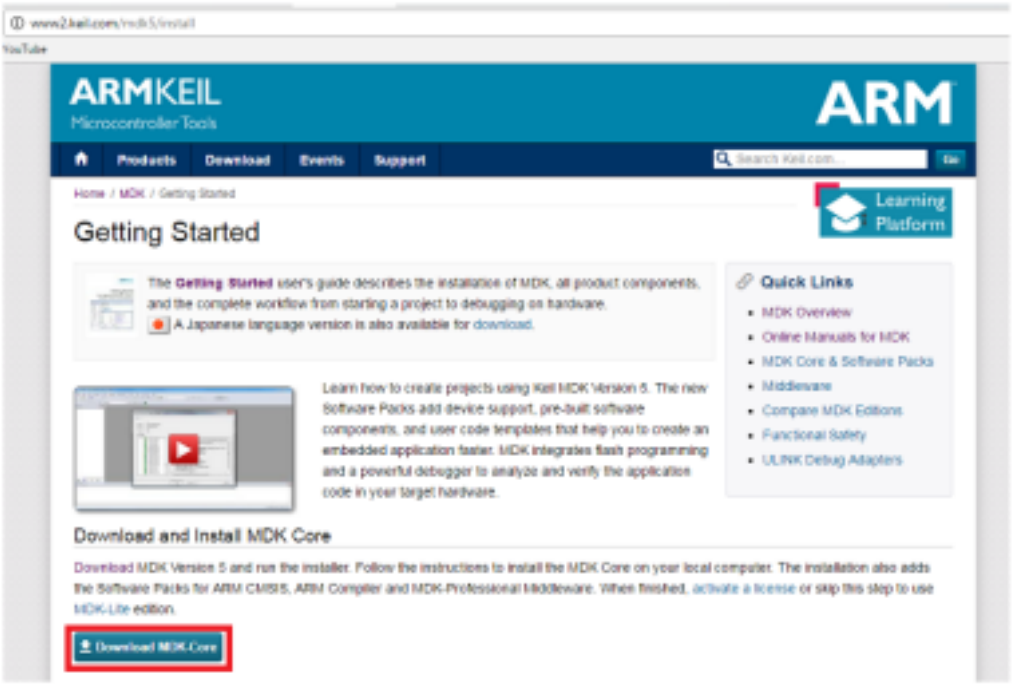
- Cross Works for Arm
- Keil µVision
- IAR Embedded Workbench

We will see how to install and setup the µVision IDE by Keil.
We will see the steps that need to be followed for installing this software correctly. When this is done, we will setup the environment for LPC2148 and write a basic code for LED blinking.

Downloading and installation

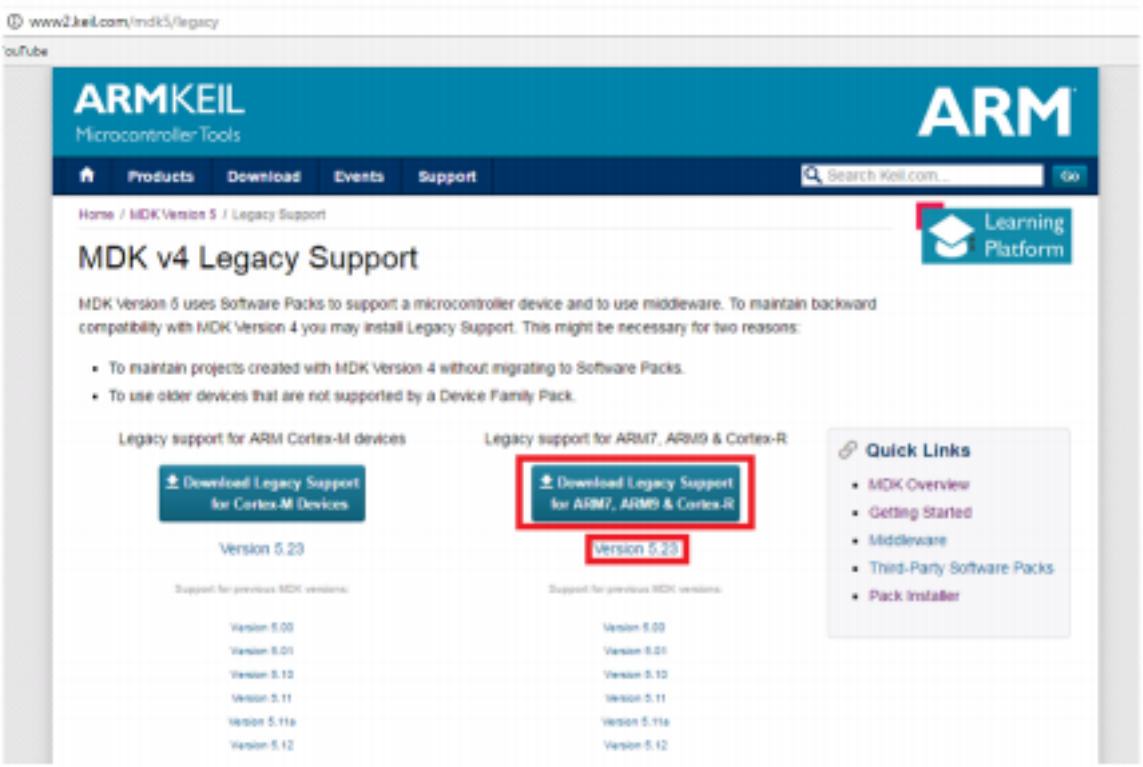
Follow the steps given below:

1. Download the MDK-lite (Microcontroller Development Kit) by Keil from their website. Here is the link to the page from where you can download this : http://www2.keil.com/mdk5/install Click on **Download MDK-Core**.



Install the software by following the simple instructions provided during installation process.

2. The new version µVision5 does not support many of the devices that were supported in the older versions yet. LPC2148 is one of the devices that are not supported. Hence, we need to add this device after successfully installing µVision5. To do this, go to the following link and download the executable file for Legacy Support for ARM7, ARM9, and Cortex-R: http://www2.keil.com/mdk5/legacy Download the Legacy support for the version of MDK downloaded and installed.

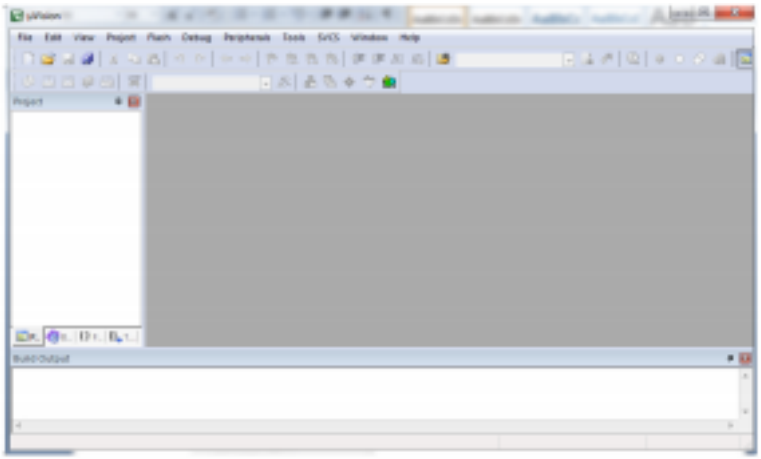


Install the executable file that will be downloaded. Follow the simple instructions provided during the installation process.
When the above described steps are completed, we will have the IDE installed and ready to use with support for the device we intend to use, i.e. LPC2148.

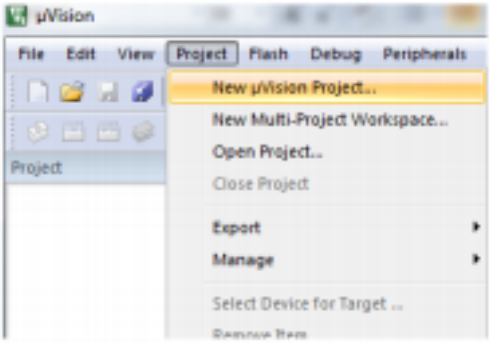
Using µVision IDE

We will create a simple LED blinking project. Following are steps which show how to create and built project using the Keil uVision IDE:

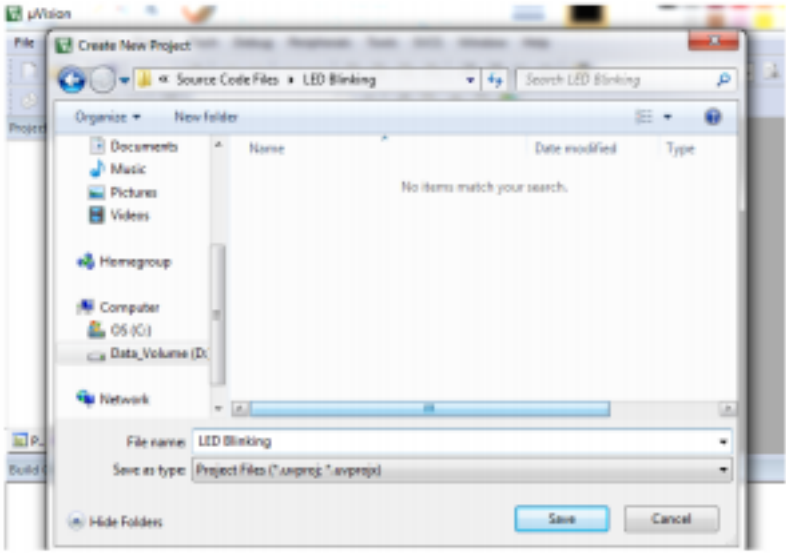
1. Open Keil µVision from the icon created on your desktop.



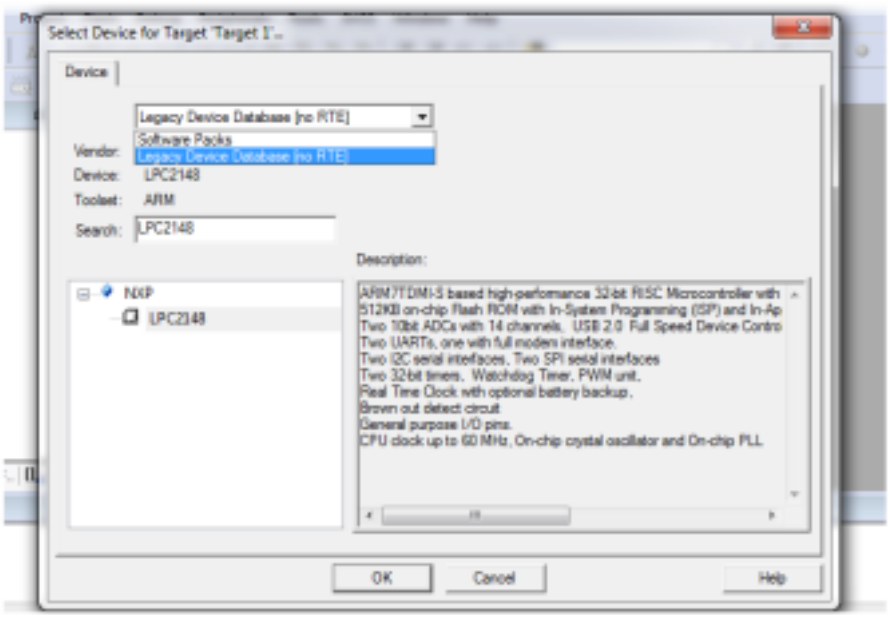
2. Go to the **Project** tab. Select New **µVision Project ...**from that menu.



3. **Create New Project** window will pop up. Select the folder where you want to create project and give a suitable name to the project. Then click on **Save**.

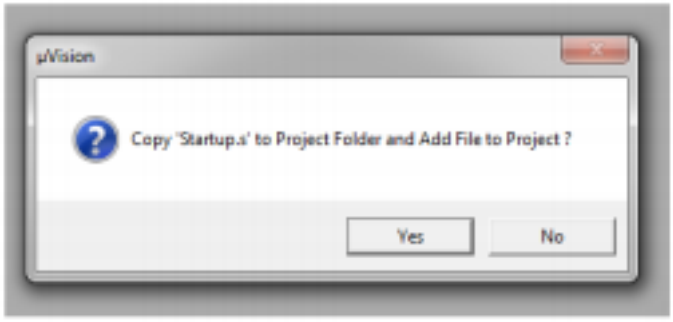


4. **Select Device for Target: 'Target1'...**window will pop up next. It has a select window to choose between Software Packs or Legacy Device Database. As LPC2148 is in Legacy Device Database, choose Legacy Device Database.

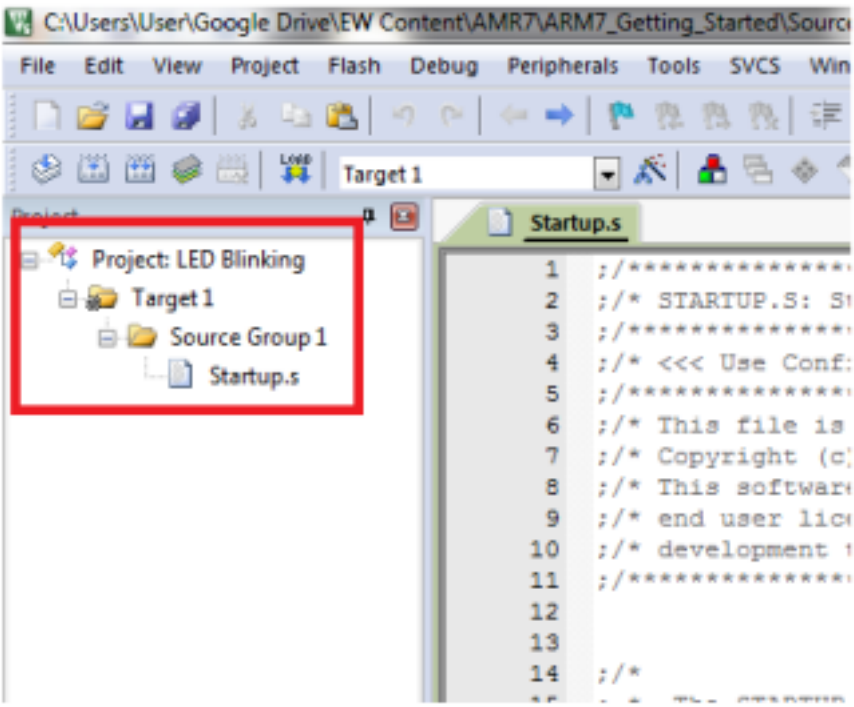


Type in LPC2148 in search and select the device under NXP with the name LPC2148 and click on OK.

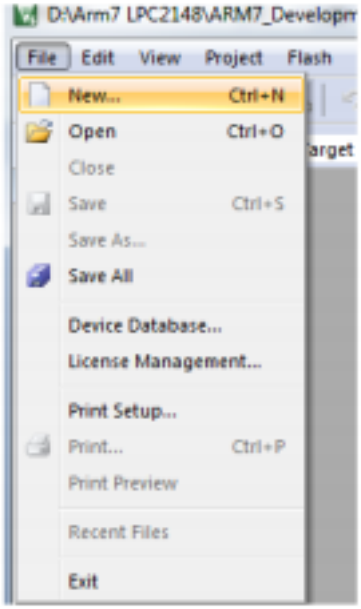
5. A window will pop up asking whether to copy Startup.s to project folder and add file to project. Click on **Yes**.



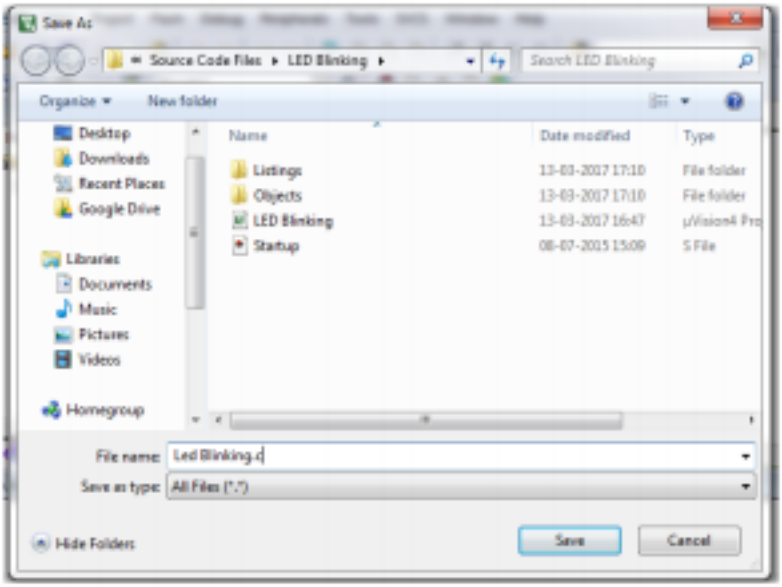
6. The project name and its folders can be seen on the left side in the project window after the previous step is completed as shown below.



7. Now go to File tab and add **New** file from the menu.

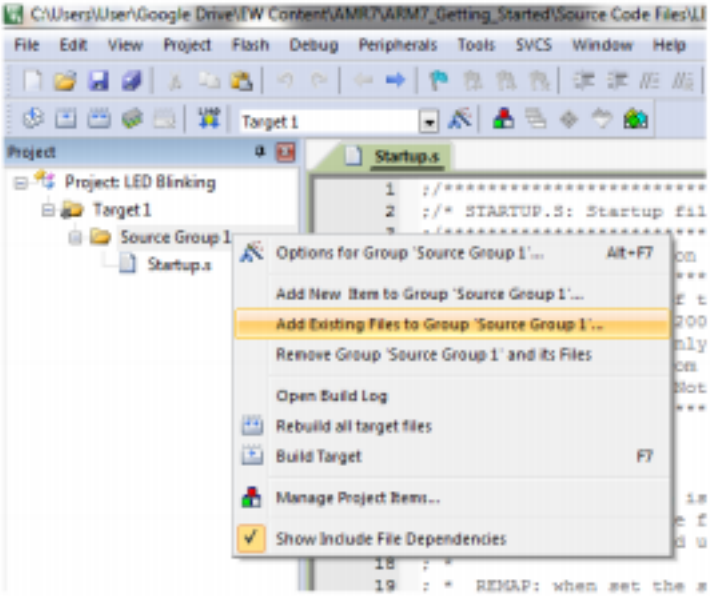


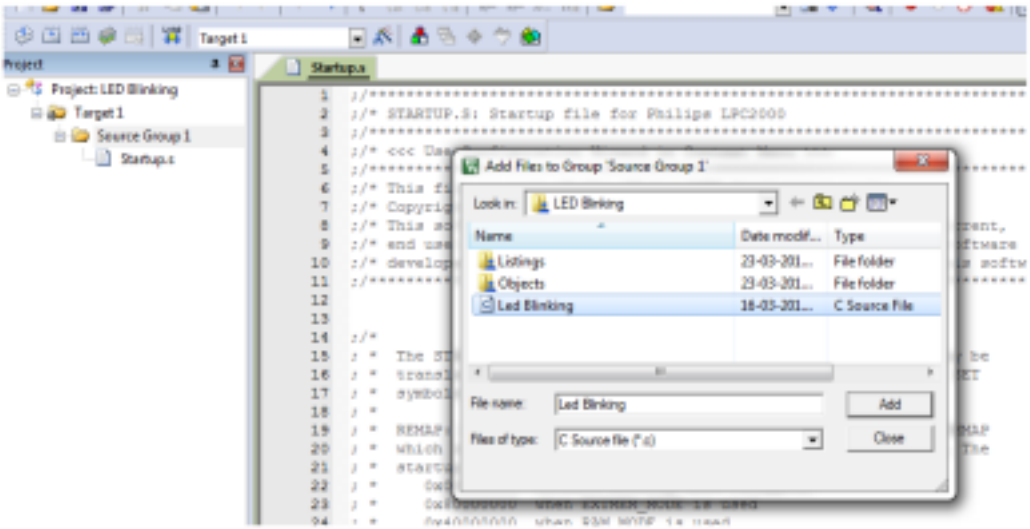
8. Save the file from the previous step with a specific name. Add .c extension to the



file name.

9. Add this file to Source Group folder in the project window by right clicking on Source Group1 folder and selecting **Add Existing Files to Group 'Source Group1'...**

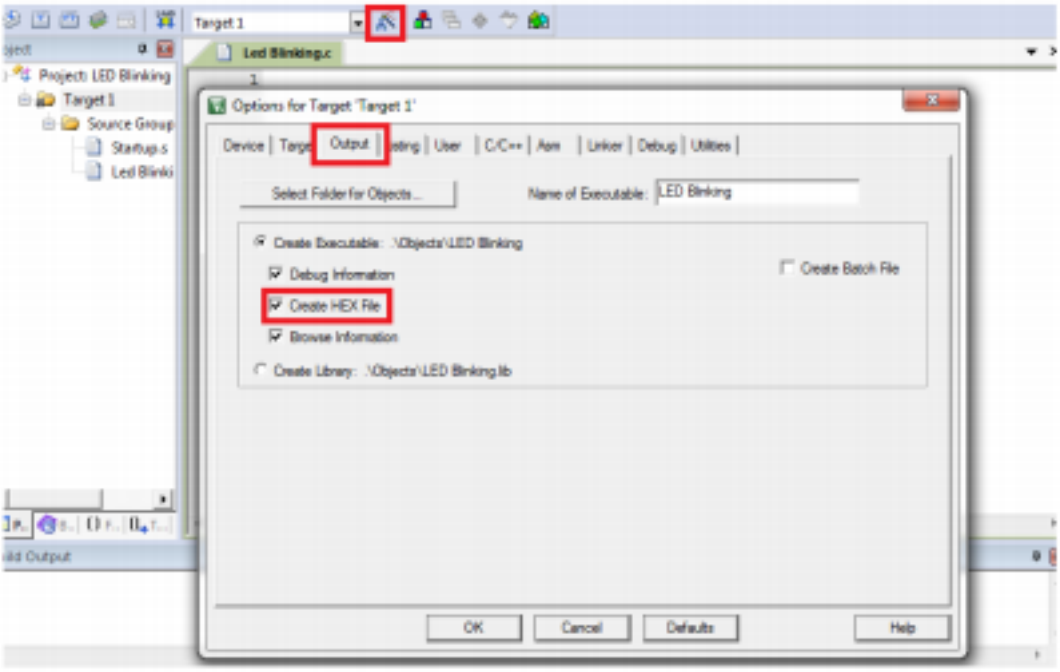




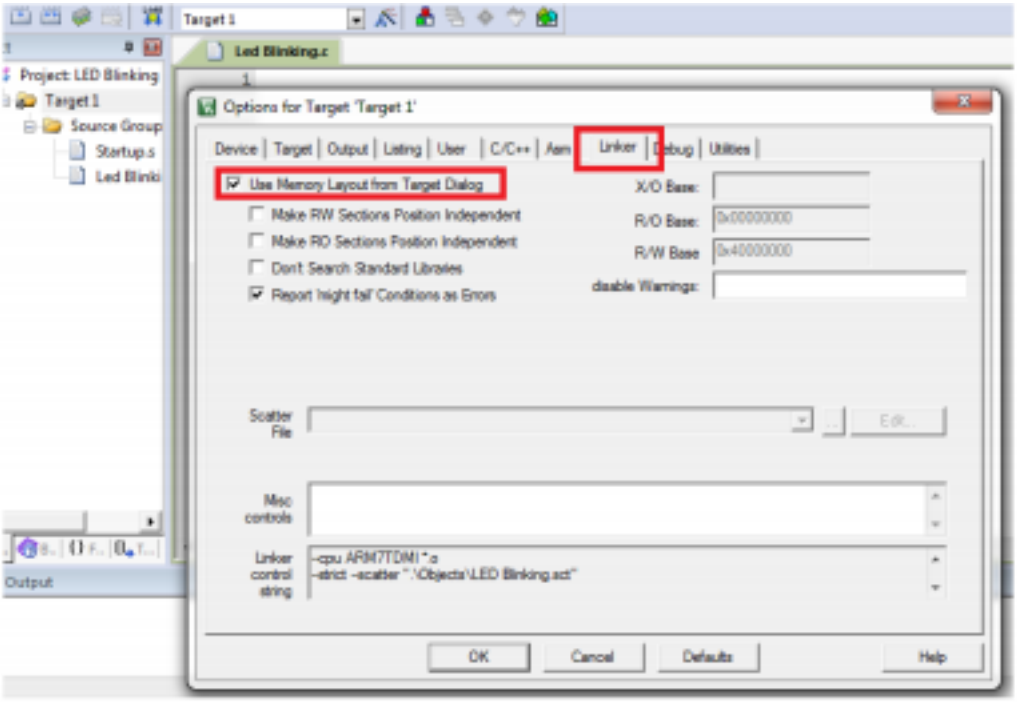
Select the previously saved file from the window that pops up and add it to the Source Group1. In our case, LED Blinking.c

10. Now click on the **Options for Target 'Target1'...** symbol shown in red box in the image below or press **Alt+F7** or right click on Target1 and click on **Options for Target 'Target1'...**.

Options for target window will open. Go to the **Output** tab in that window. Tick '√' **Create HEX File** option. We need to produce HEX file to burn it into the microcontroller.



In the options for target window, go to the **Linker** tab. Select the **Use Memory Layout from Target Dialogue** option.



Then click on OK.

11. Now write the code for LED Blinking.

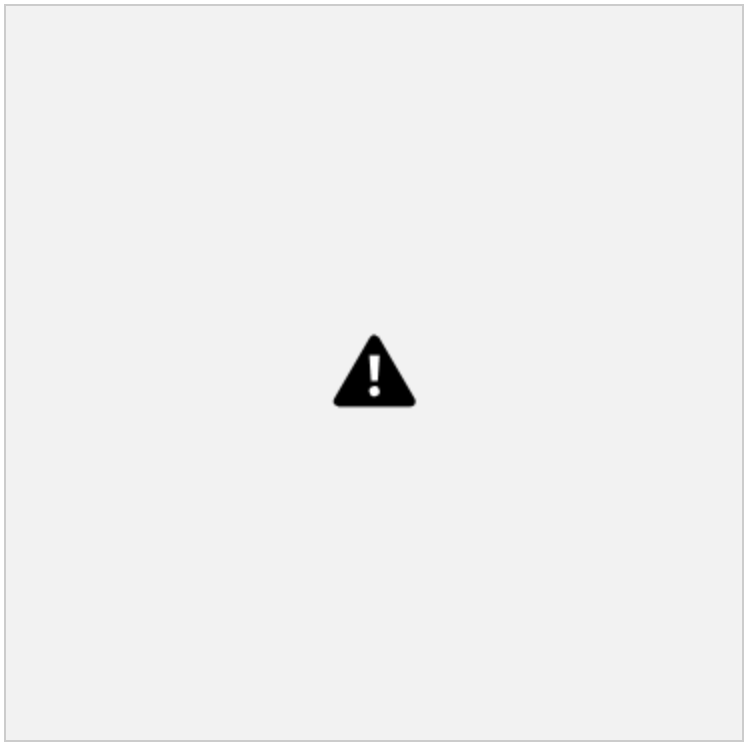
```
#include <lpc214x.h>
#include <stdint.h>

Void delay_ms(uint16_t j) /* Function for delay in milliseconds */
{
 uint16_t x,i;
for(i=0;i<j;i++)
{
 for(x=0; x<6000; x++); /* loop to generate 1 millisecond delay with 12MHz Fosc. */ }
}

int main(void)
{
IO0DIR = 0x000000FF; /* Set P0.0 to P0.7 bits as output bits by writing 1 in IO0DIR register corresponding to those bits. */
while(1)
{
IO0PIN = IO0PIN | 0x000000FF; /* Make P0.0 to P0.7 HIGH while keeping other bits unchanged. */
delay_ms(300);
IO0PIN = IO0PIN & 0xFFFFFF00; /* Make P0.0 to P0.7 LOW while keeoing other bits
```

```
unchanged. */
delay_ms(300);
}
}
```

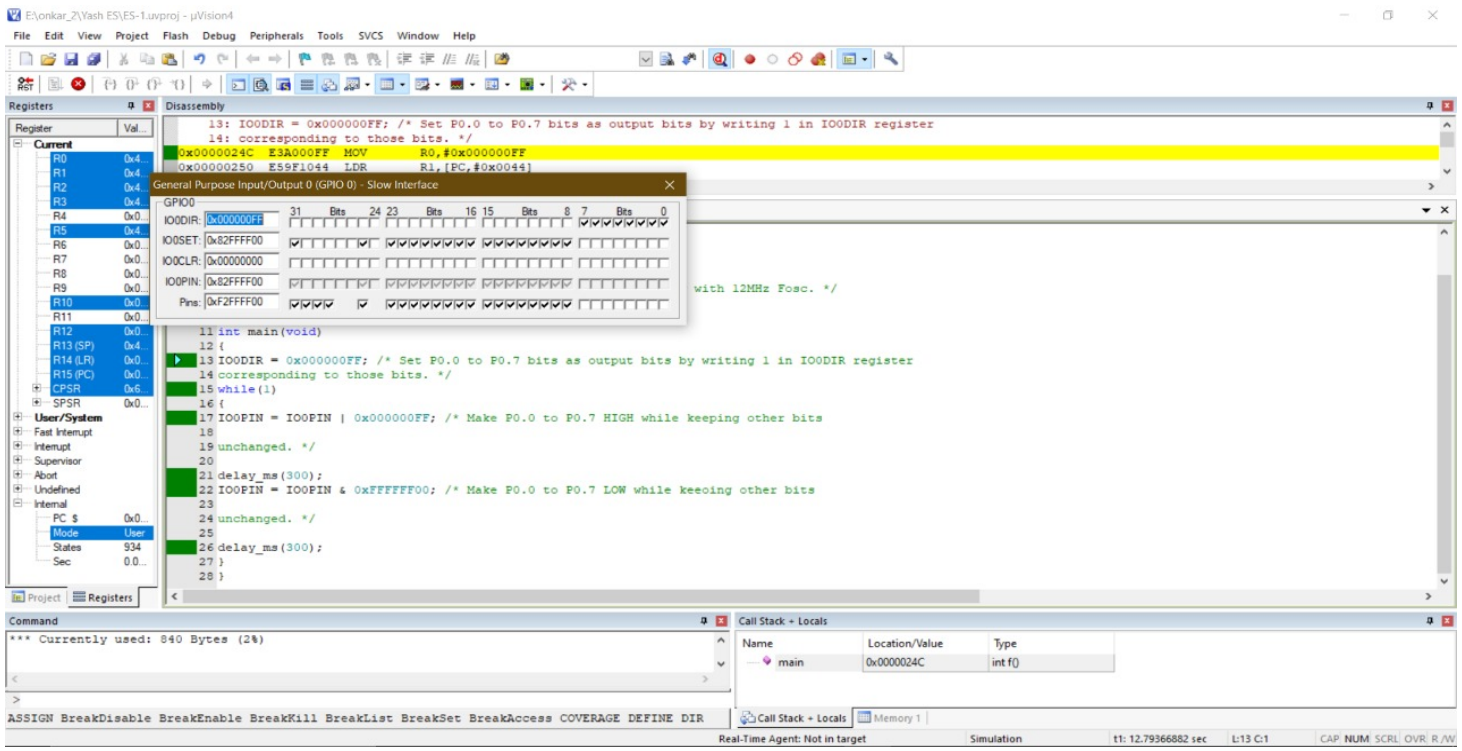
12. Once the code is written, **Build** the code by clicking on the button shown in red in the image below. You can also build the project from the **Build Target** option in the Project tab or by pressing **F7** on the keyboard.

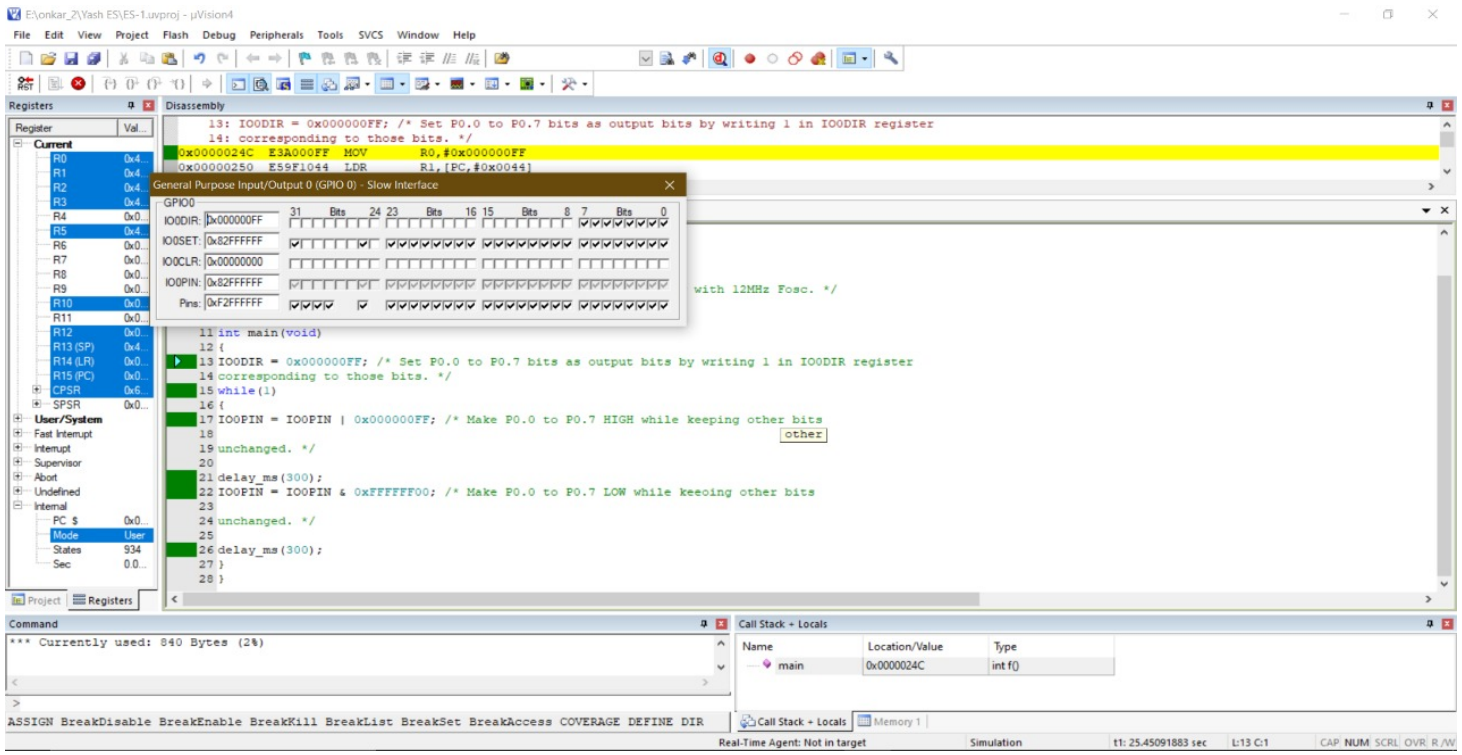


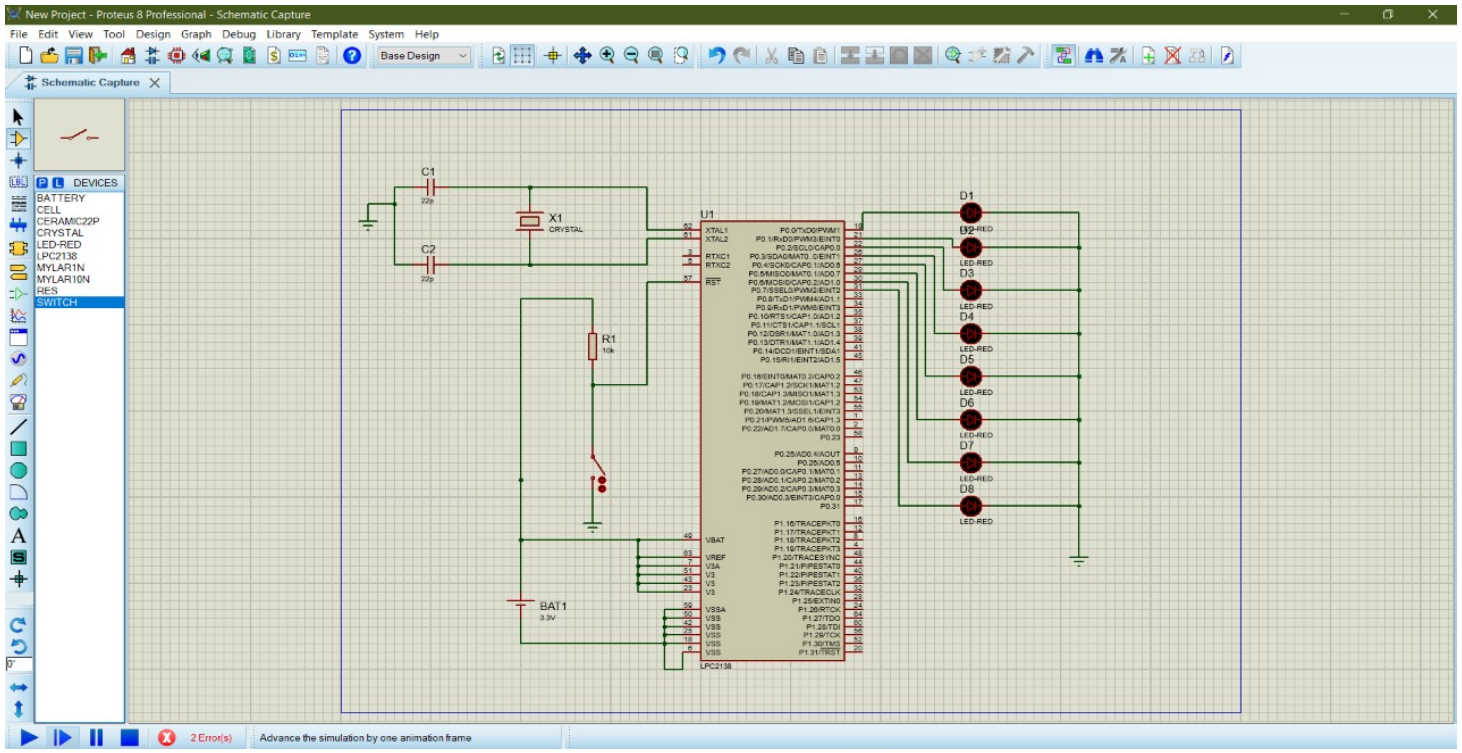
You can see **creating hex file ...** in the Build Output window as shown in the image.

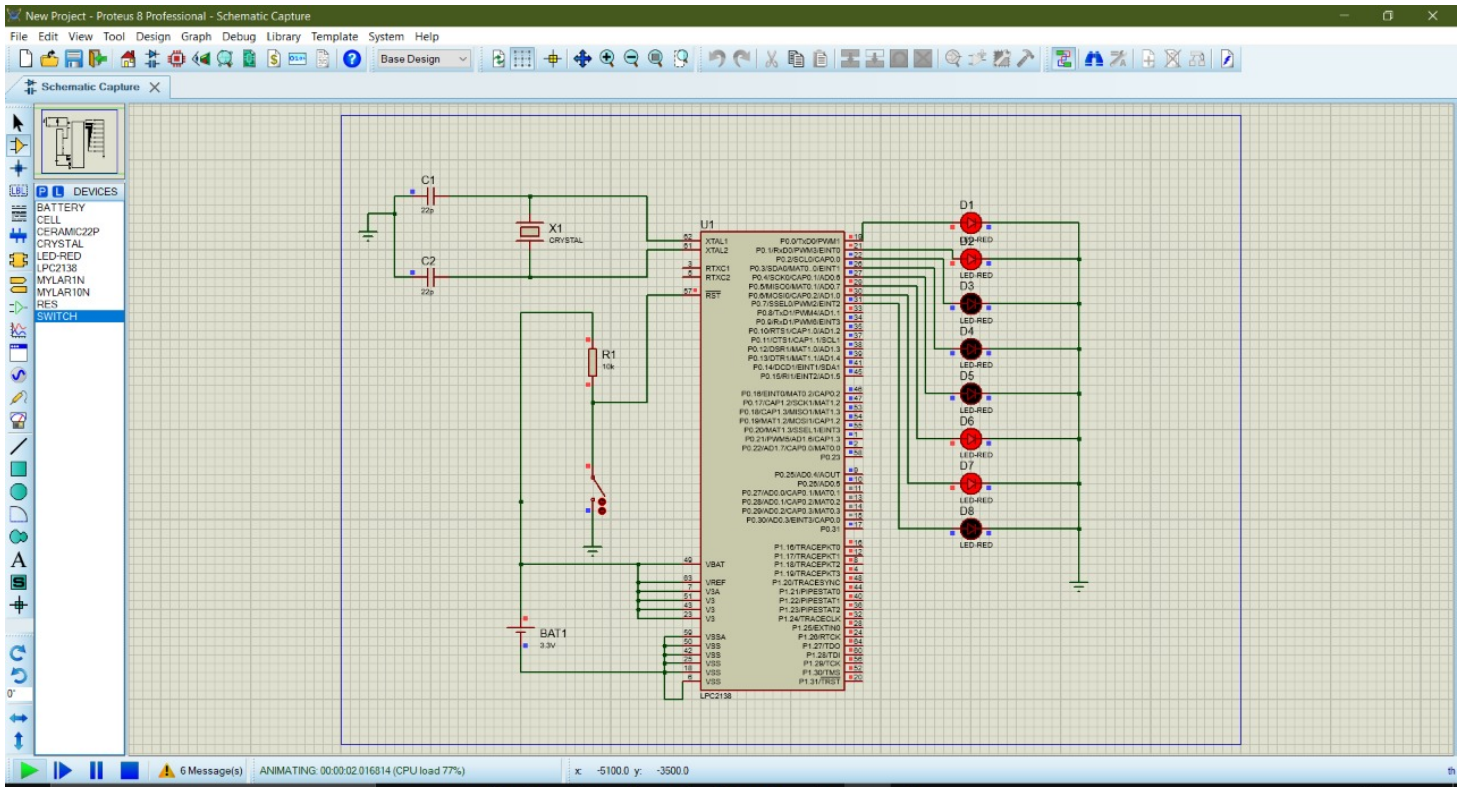
13. Once the project is built, a **hex file** is created in the **Objects** folder inside the folder of your project. Use **Flash Magic** software to burn this hex file in your microcontroller.

Output:


E:\onkar_2\Yash ES\ES-1.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
Register
Current
R0 R1 R2 R3 R4 R5 R6 R7 R8 R9 R10 R11 R12 R13 (SP) R14 (LR) R15 (PC) CPSR SPSR
User/System
Fast Interrupt
Interrupt
Supervisor
Abort
Undefined
Internal
PC $
Mode User
States 934
Sec
Project Registers
Disassembly
13: IO0DIR = 0x000000FF; /* Set P0.0 to P0.7 bits as output bits by writing 1 in IO0DIR register
14: corresponding to those bits. */
0x0000024C E3A000FF MOV R0,#0x000000FF
0x00000250 E59F1044 LDR R1,[PC,#0x0044]
General Purpose Input/Output 0 (GPIO 0) - Slow Interface
GPIO0
IO0DIR: 0x000000FF
IO0SET: 0x82FFFF00
IO0CLR: 0x00000000
IO0PIN: 0x82FFFF00
Pins: 0xF2FFFF00
31 Bits 24 23 Bits 16 15 Bits 8 7 Bits 0
with 12MHz Fosc. */
11 int main(void)
12 {
13 IO0DIR = 0x000000FF; /* Set P0.0 to P0.7 bits as output bits by writing 1 in IO0DIR register
14 corresponding to those bits. */
15 while(1)
16 {
17 IO0PIN = IO0PIN | 0x000000FF; /* Make P0.0 to P0.7 HIGH while keeping other bits
18
19 unchanged. */
20
21 delay_ms(300);
22 IO0PIN = IO0PIN & 0xFFFFFF00; /* Make P0.0 to P0.7 LOW while keeoing other bits
23
24 unchanged. */
25
26 delay_ms(300);
27 }
28 }
Command
*** Currently used: 840 Bytes (2%)
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE DEFINE DIR
Call Stack + Locals
Name Location/Value Type
main 0x0000024C int f()
Call Stack + Locals Memory 1
Real-Time Agent: Not in target
Simulation
t1: 12.79366882 sec
L:13 C:1
CAP NUM SCRL OVR R/W


E:\onkar_2\Yash ES\ES-1.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
Disassembly
13: IO0DIR = 0x000000FF; /* Set P0.0 to P0.7 bits as output bits by writing 1 in IO0DIR register
14: corresponding to those bits. */
0x0000024C E3A000FF MOV R0,#0x000000FF
0x00000250 E59F1044 LDR R1,[PC,#0x0044]
General Purpose Input/Output 0 (GPIO 0) - Slow Interface
GPIO0
IO0DIR: 0x000000FF
IO0SET: 0x82FFFFFF
IO0CLR: 0x00000000
IO0PIN: 0x82FFFFFF
Pins: 0xF2FFFFFF
11 int main(void)
12 {
13 IO0DIR = 0x000000FF; /* Set P0.0 to P0.7 bits as output bits by writing 1 in IO0DIR register
14 corresponding to those bits. */
15 while(1)
16 {
17 IO0PIN = IO0PIN | 0x000000FF; /* Make P0.0 to P0.7 HIGH while keeping other bits
18 other
19 unchanged. */
20
21 delay_ms(300);
22 IO0PIN = IO0PIN & 0xFFFFFF00; /* Make P0.0 to P0.7 LOW while keeoing other bits
23
24 unchanged. */
25
26 delay_ms(300);
27 }
28 }
Command
*** Currently used: 840 Bytes (2%)
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE DEFINE DIR
Call Stack + Locals
Name Location/Value Type
main 0x0000024C int f()
Real-Time Agent: Not in target
Simulation
t1: 25.45091883 sec
L:13 C:1


New Project - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
DEVICES
BATTERY
CELL
CERAMIC22P
CRYSTAL
LED-RED
LPC2138
MYLAR1N
MYLAR10N
RES
SWITCH
C1
22p
C2
22p
X1
CRYSTAL
R1
10k
BAT1
3.3V
U1
LPC2138
D1
D2
D3
D4
D5
D6
D7
D8
LED-RED
2 Error(s)
Advance the simulation by one animation frame



Conclusion:Hence we have performed the LED blinking program using Keil Software and Proteus simulation