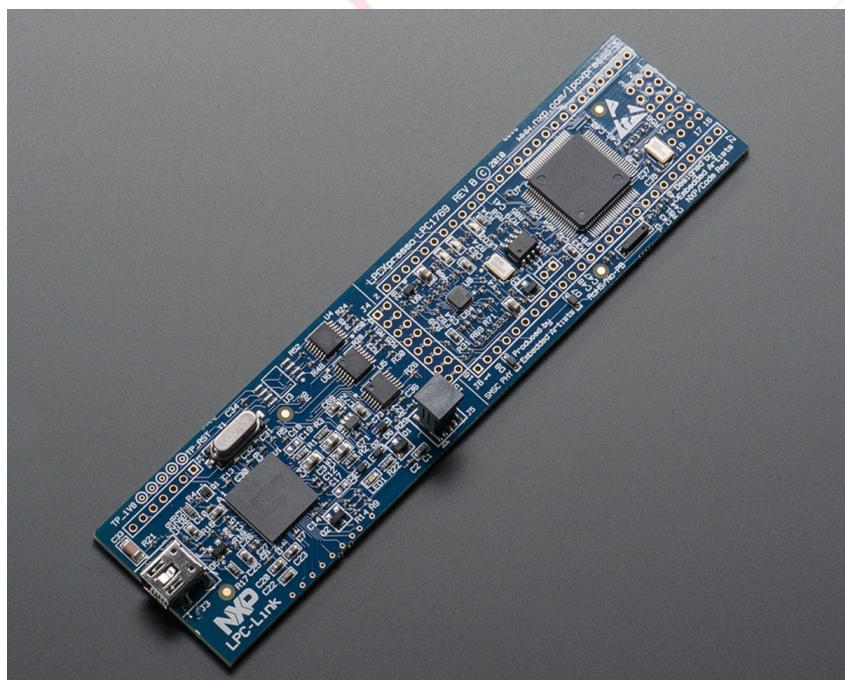




2016

Interfacing 2x16 LCD with NXP LPC1769 using LPCXpresso



Author: Gurudatta Palankar

Reviewers:

Version: 1.0

Introduction:

LPCXpresso™ is a new, low-cost development platform available from NXP supporting NXP's ARM-based micro-controllers. The platform is comprised of a simplified Eclipse-based IDE and low-cost target boards which include an attached JTAG debugger. LPCXpresso™ is an end-to-end solution enabling engineers to develop their applications from initial evaluation to final production.

Step 1: Open LPCXpresso IDE

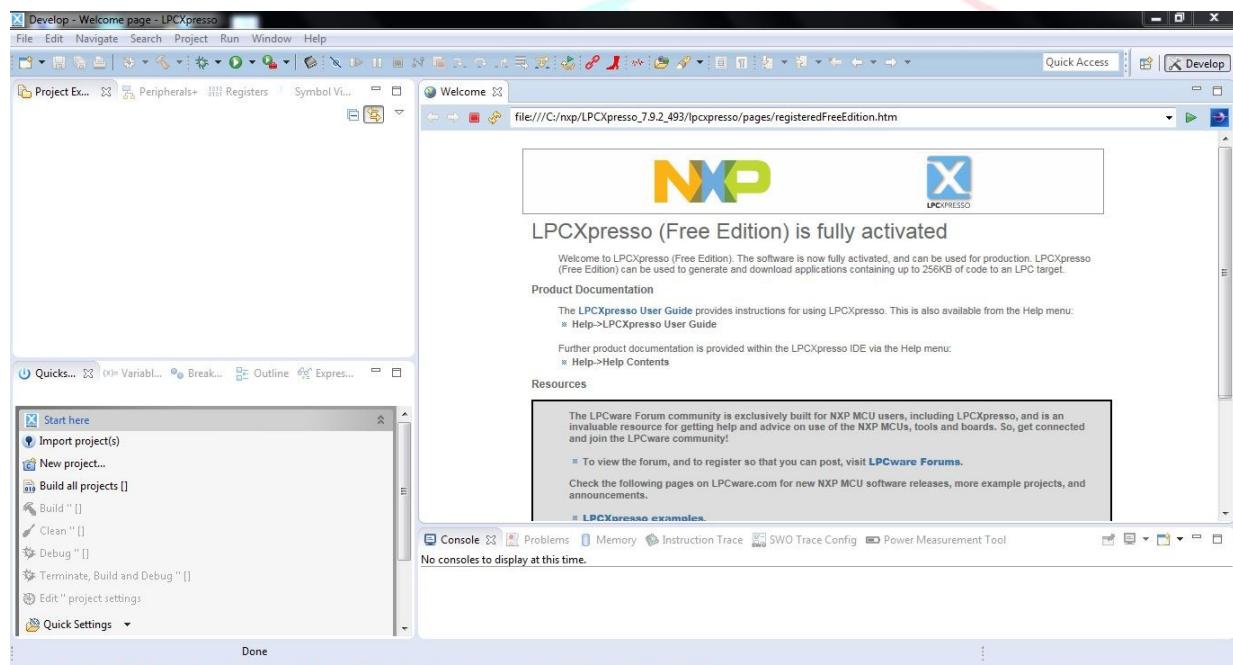


Figure 1

TENET
TECHNETRONICS

Step 2: Before writing a code, we have to Import some Library Files to the Workspace. Click on Import projects on Quickstart Panel on the bottom left of the window.

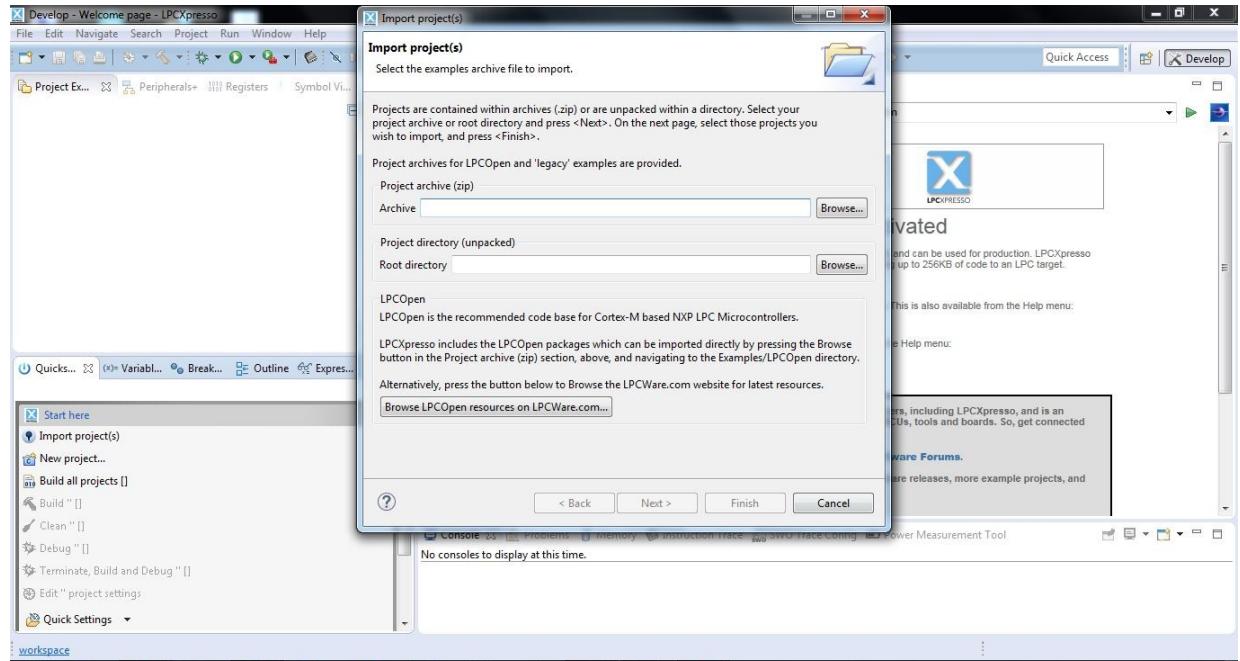


Figure 2

Step 3: Browse file, open the LPC1000 folder.

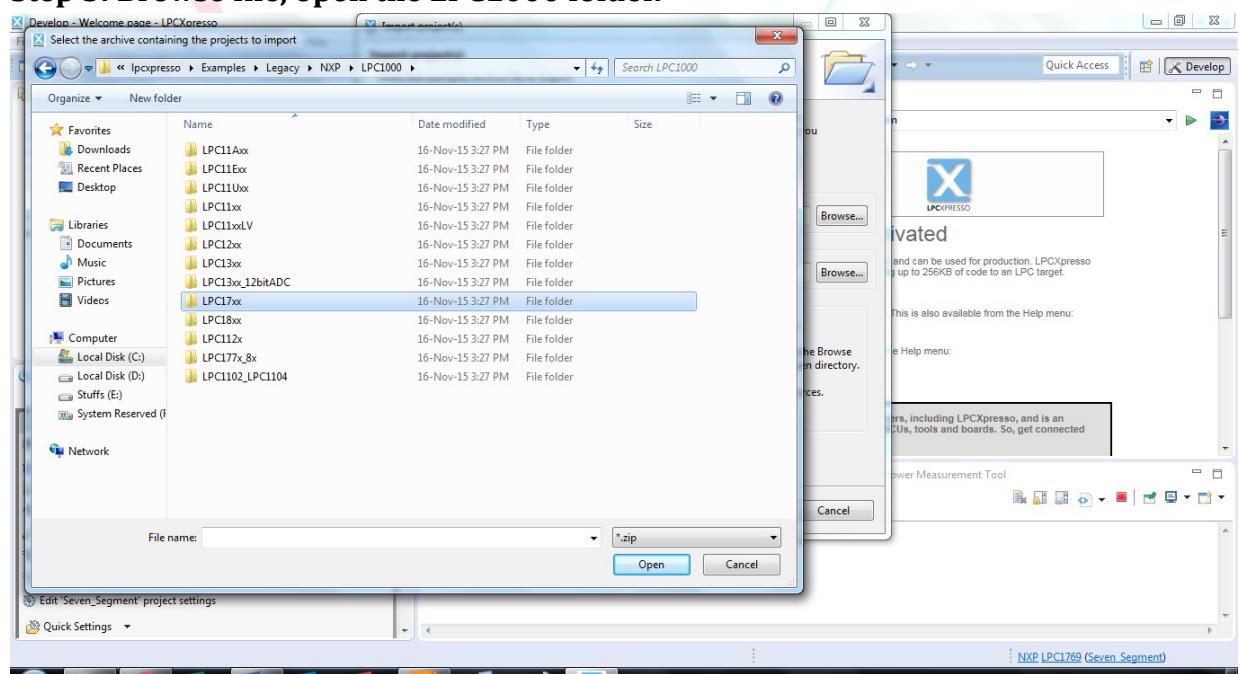


Figure 3

Step 4: Select the appropriate archive file. Let us select LPCXpresso176x_cmsis2. We can select CMSIS CORE library that include LPC17xx.h header file.

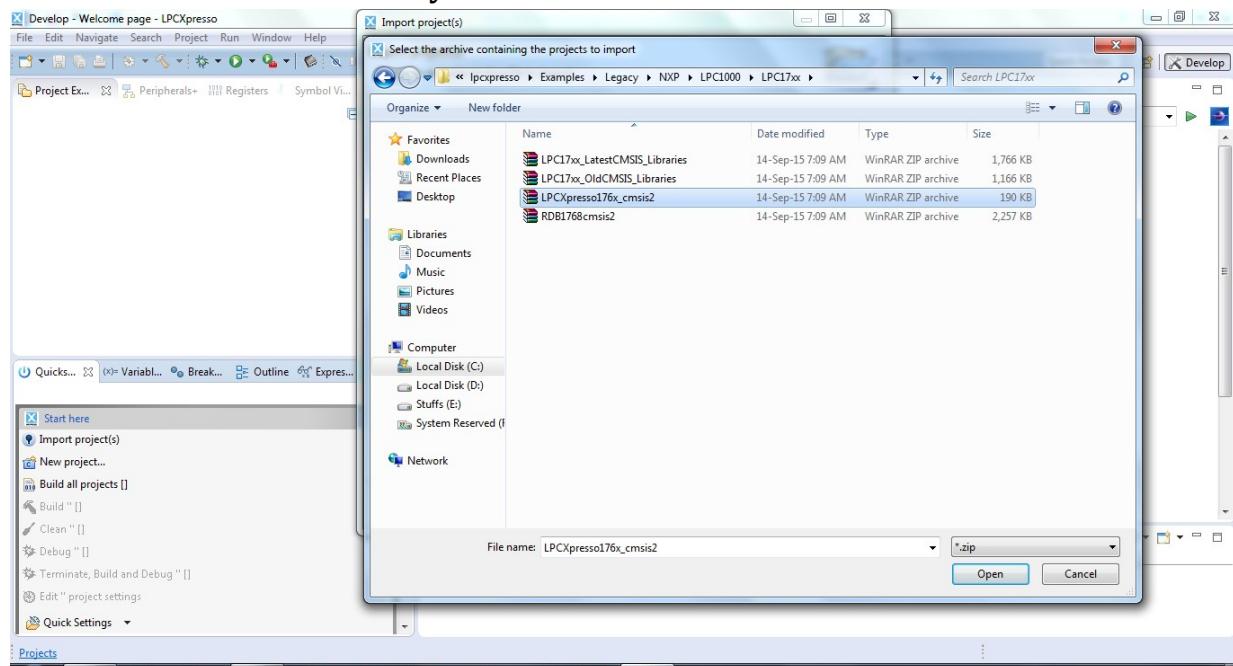


Figure 4

Step 5: After selecting you will be able to see the following libraries files. Let us select specific one.

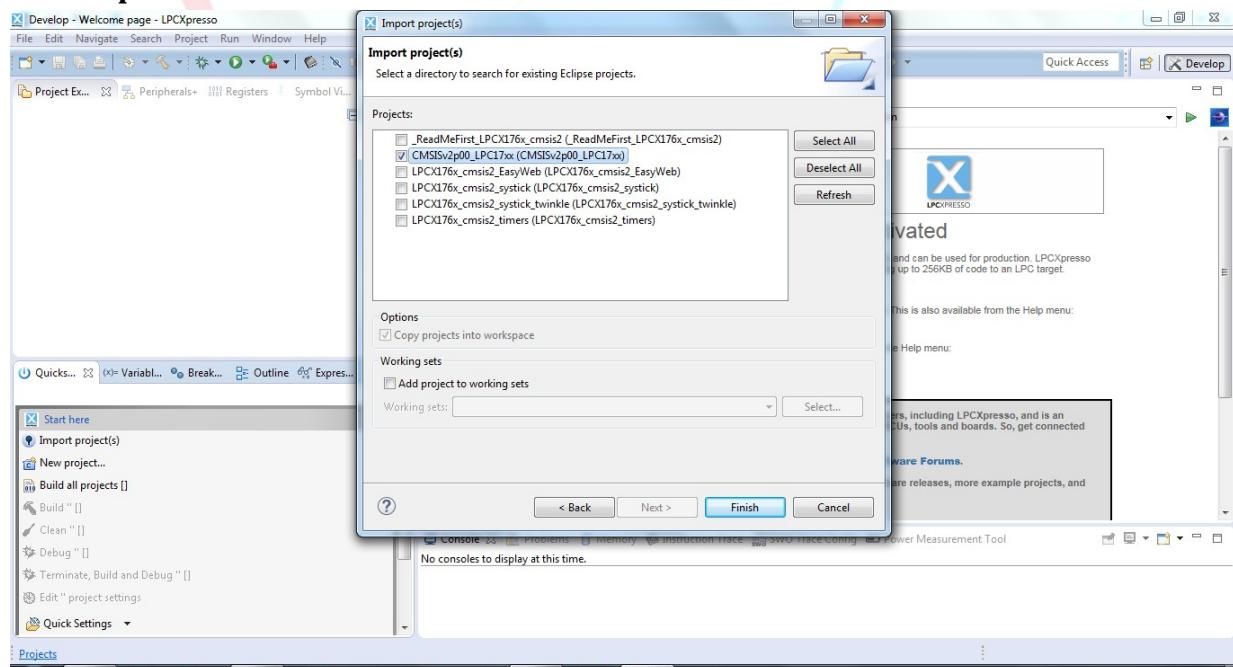


Figure 5

Step 6: Now we will be able to see those libraries in the workspace.

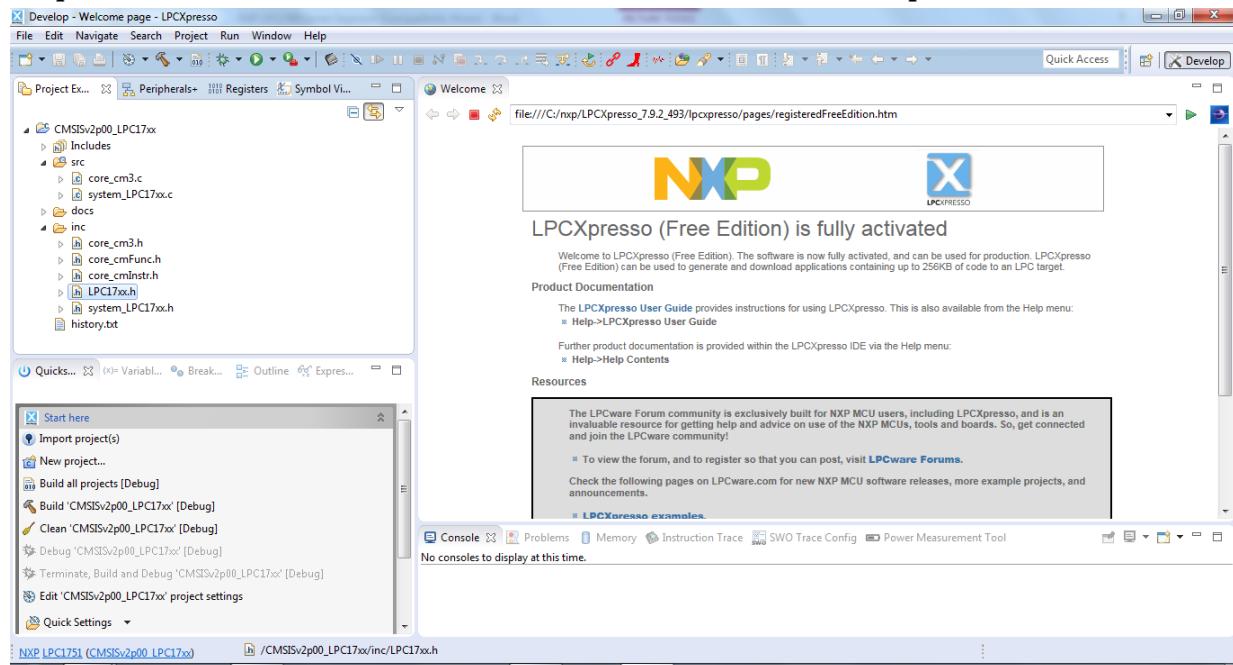


Figure 6

Step 7: Now we can start creating our new project. Goto File >> New >> Project. Select LPCXpresso C project.

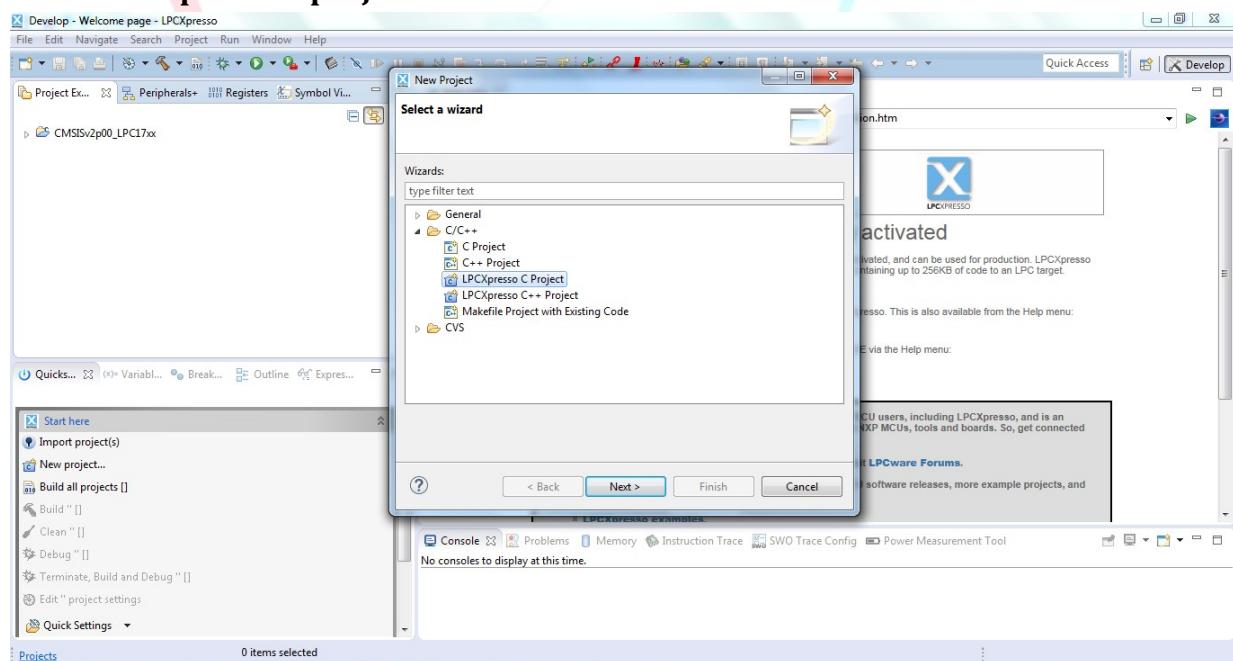


Figure 7

Step 8: Select LPC1769, C Project and give name to your project. Select target MCU as LPC1769.

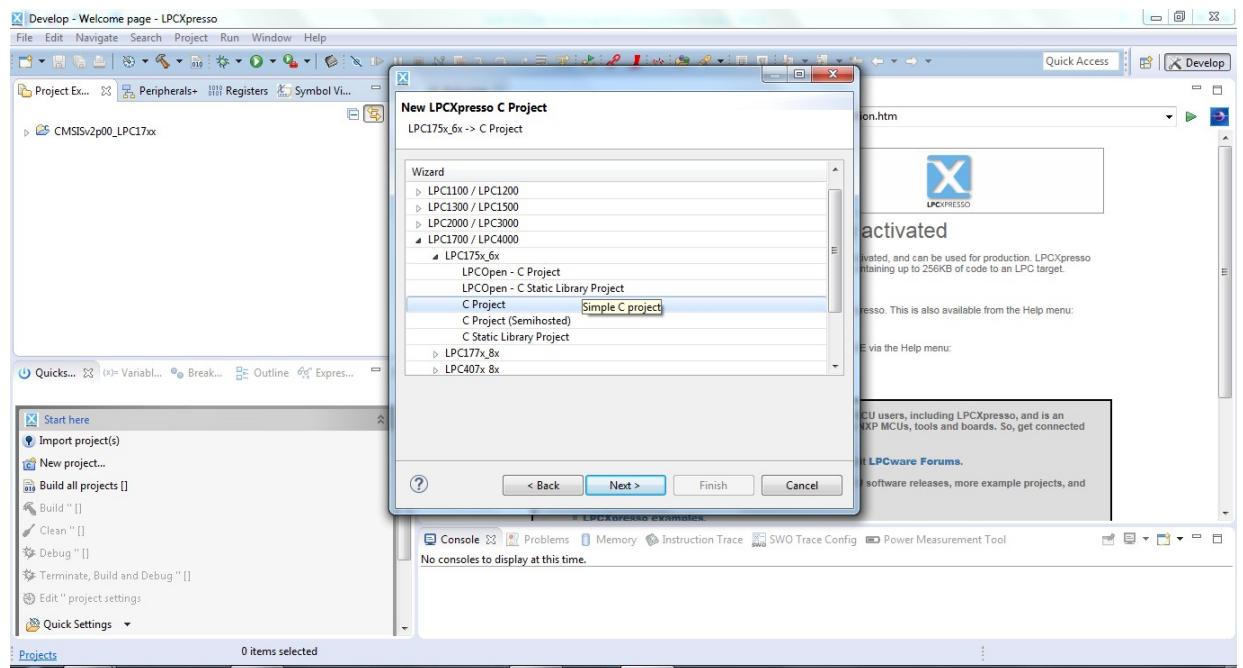


Figure 5

Step 9: Now select CMSIS Core library. Click on Next and keep all the other configurations as default and Finish.

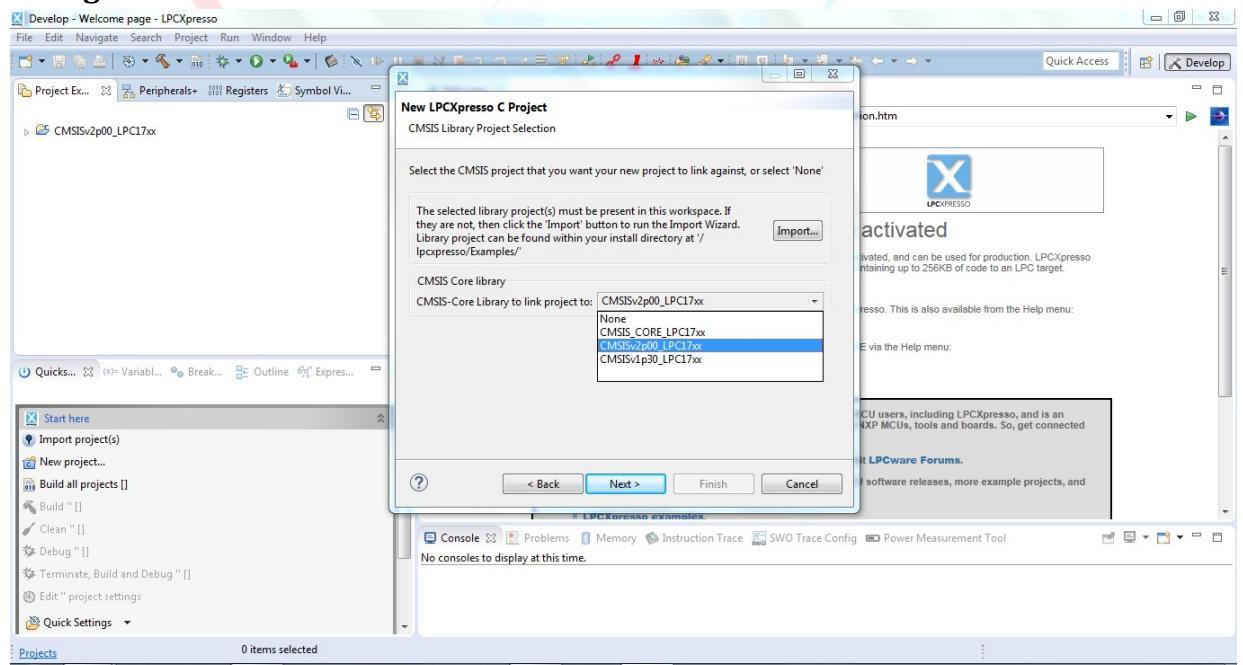


Figure 9

Step 10: Now we can see our project onto the workspace. Now by double clicking on LCD_display.c file, we can start writing code.

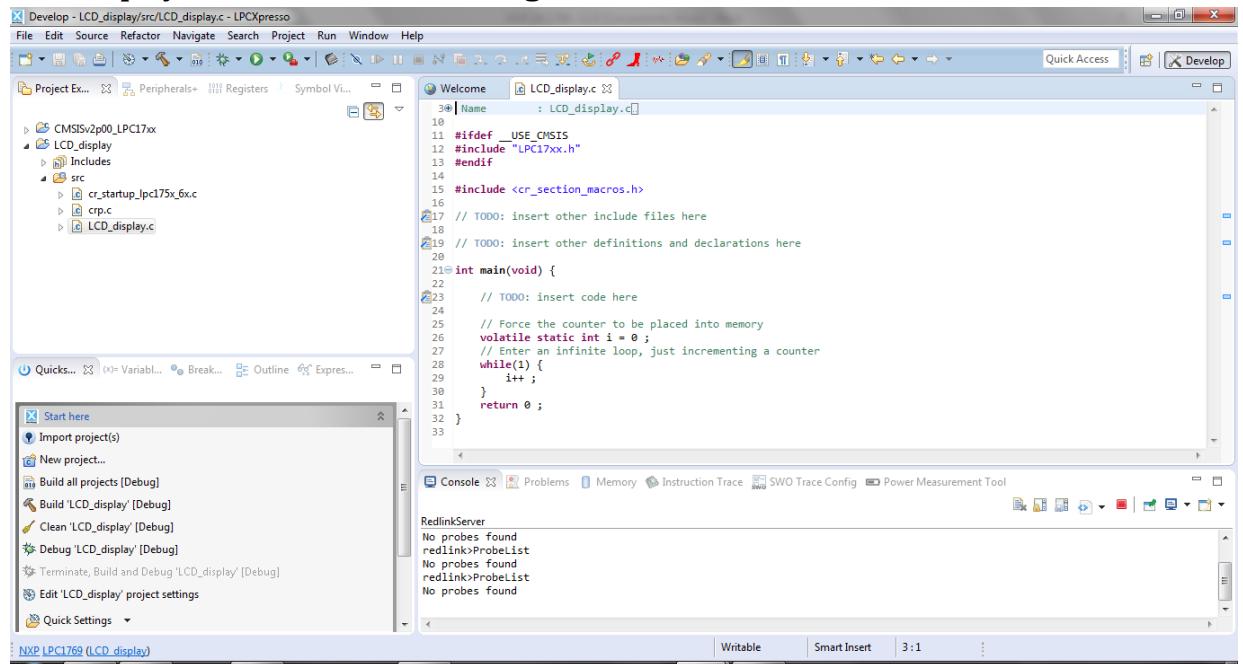


Figure 10

Step 11: Write a code as shown below.

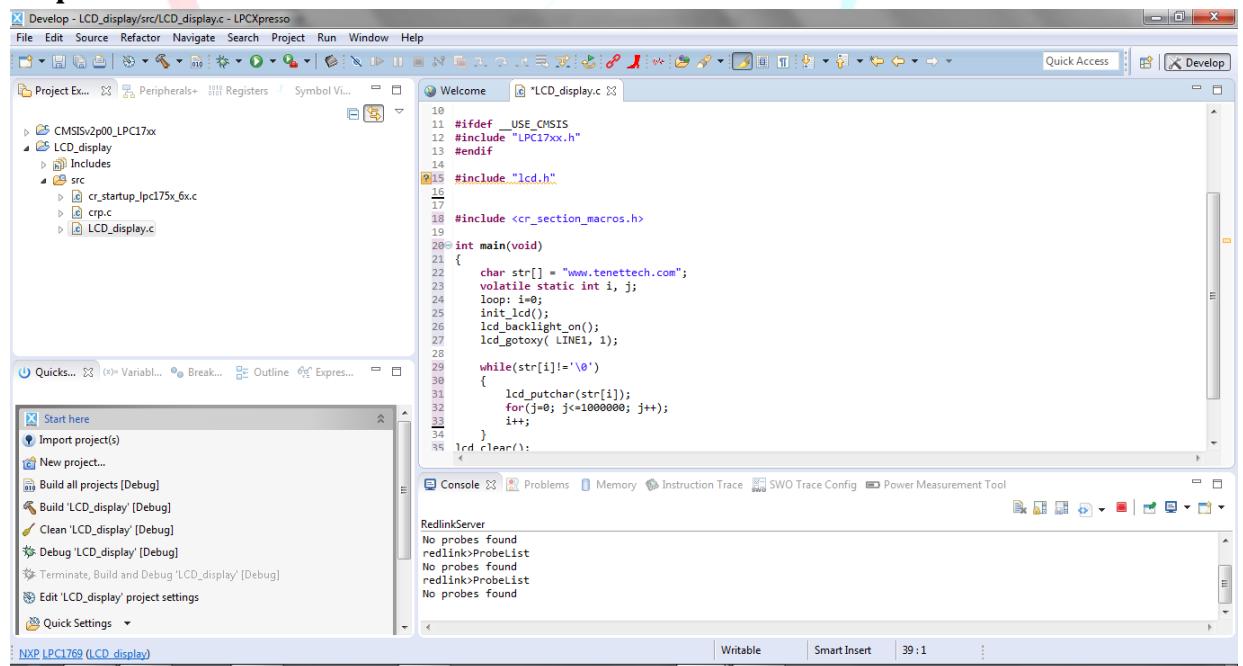


Figure 11

CODE:

```
#ifdef __USE_CMSIS
#include "LPC17xx.h"
#endif

#include "lcd.h"

int main(void)
{
    char str[] = "www.tenettech.com";
    volatile static int i, j;

loop: i=0;

    init_lcd();
    lcd_backlight_on();
    lcd_gotoxy( LINE1, 1);

    while(str[i]!='\0')
    {
        lcd_putchar(str[i]);
        for(j=0; j<=1000000; j++);
        i++;
    }

    lcd_clear();
    goto loop;

    return 0 ;
}
```

NOTE: The above code will not work until and unless we add header file ([lcd.h](#)) and Source File ([lcd.c](#)),

Step 12: To create or add library files, right click on src file of your project file, then New > Source File.

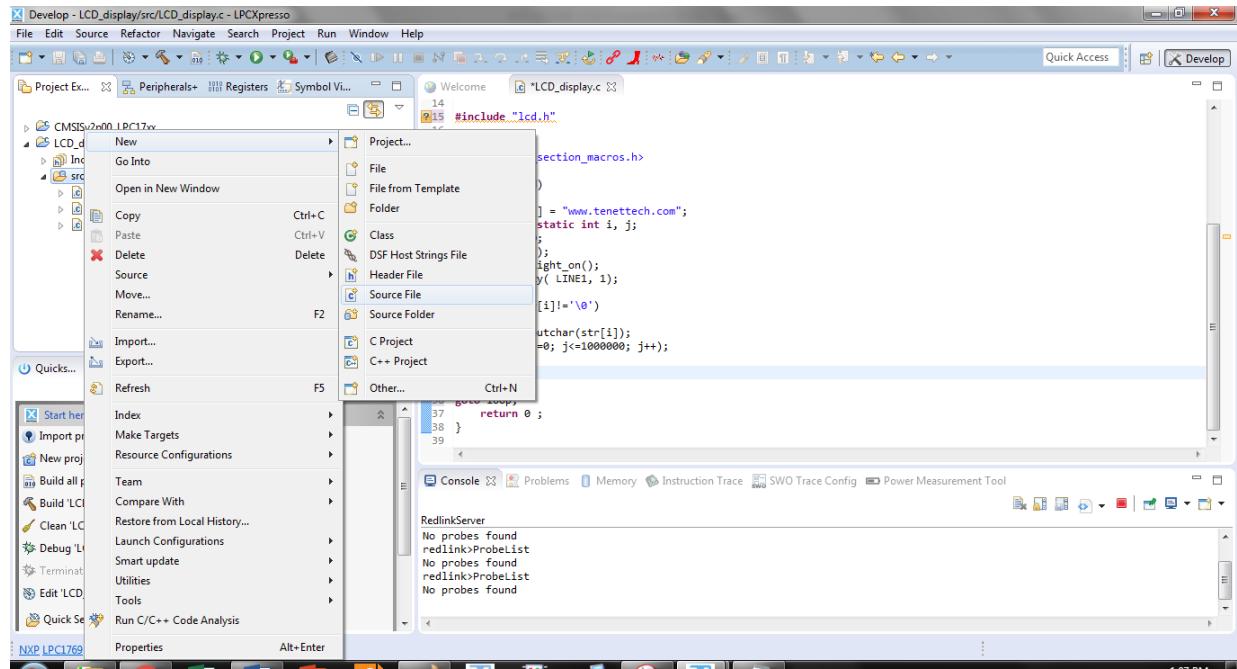


Figure 12

Step 13: Save the Source File name with .c file extension.

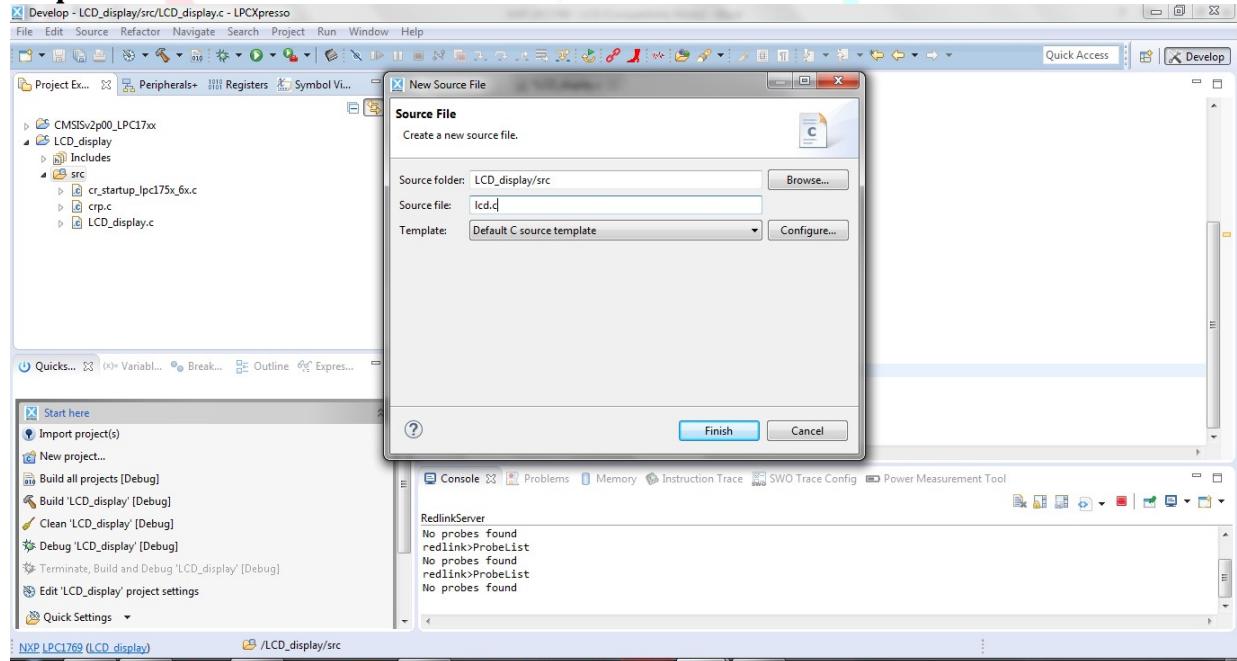


Figure 13

Step 14: Write or paste the Source File (Library).

The screenshot shows the LPCXpresso IDE interface. The top menu bar includes File, Edit, Source, Refactor, Navigate, Project, Run, Window, and Help. The left sidebar displays the project structure under 'Project Explorer' with nodes like CMSISv2p00_LPC17xx, LCD_display, and src. The main workspace shows the source code for 'lcd.c'. The code defines two functions: 'delay' and 'wait_lcd'. The 'delay' function suspends tasks for specified ticks in multiples of 1 us. The 'wait_lcd' function introduces a delay of 10 us at 60MHz. The bottom status bar shows 'NXP LPC1769 (LCD_display)'.

```

1 #include "lcd.h"
2
3 Function Name : delay()
4 Description : This function suspends the tasks for specified ticks.
5 Input : ticks: no of ticks in multiple of 1 us
6 Output : task: task to be suspended
7 Note :
8
9 ****
10 ****
11 ****
12 void delay(int count)
13 {
14     int j=0,i=0;
15     for(j=0;j<count;j++)
16     {
17         /* At 60MHz, the below loop introduces delay of 10 us */
18         for(i=0;i<35;i++);
19     }
20 }
21
22 ****
23 Function Name : wait_lcd()
24 Description :
25 Input :
26 Output : Void

```

Figure 14

Step 15: Similarly add the Header File.

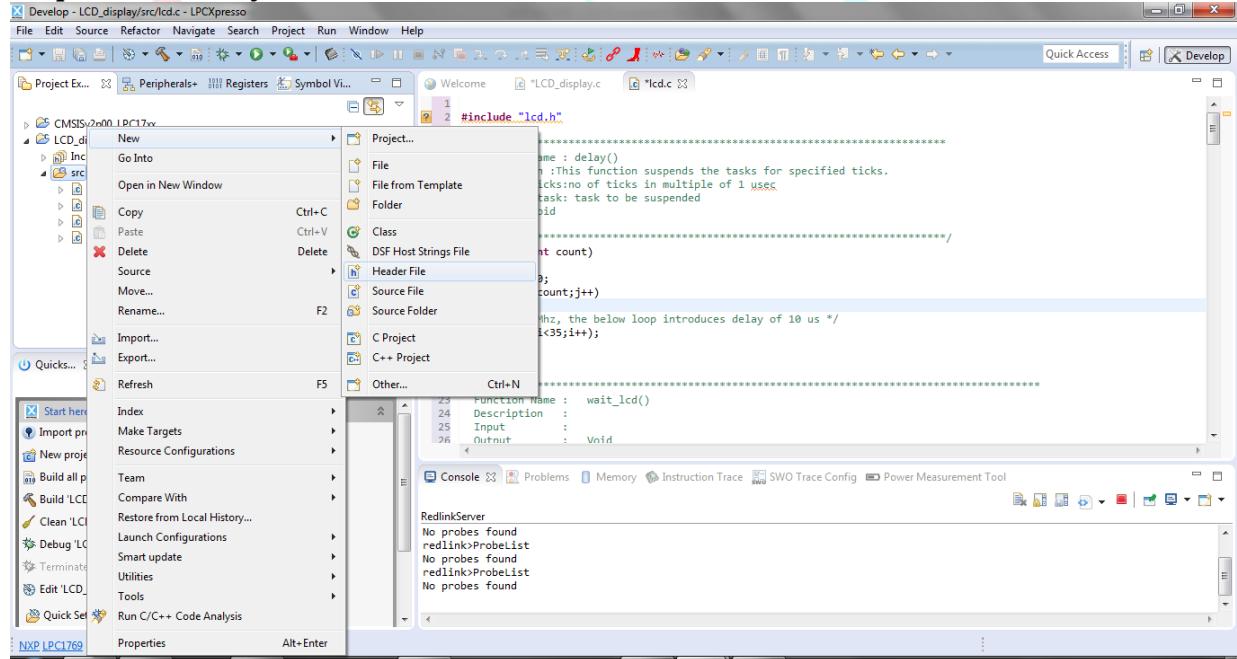


Figure 15

Step 16: Save the Header File name with .h file extension.

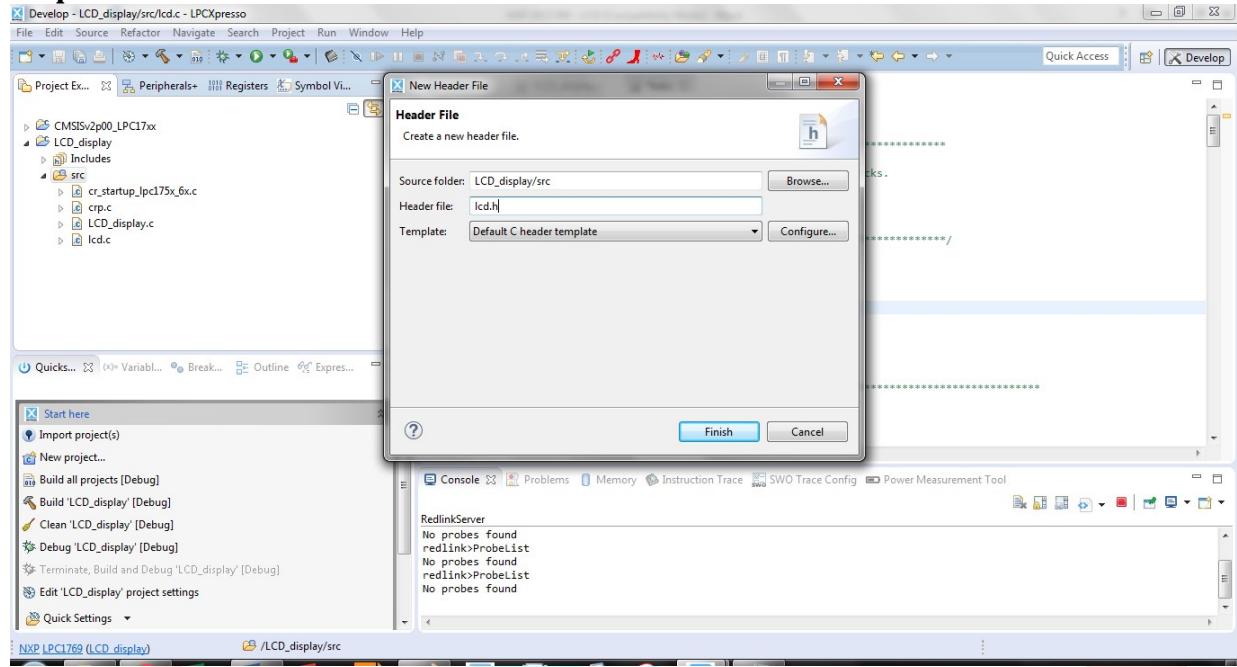


Figure 16

Step 17: Write or paste the Header File (Library).

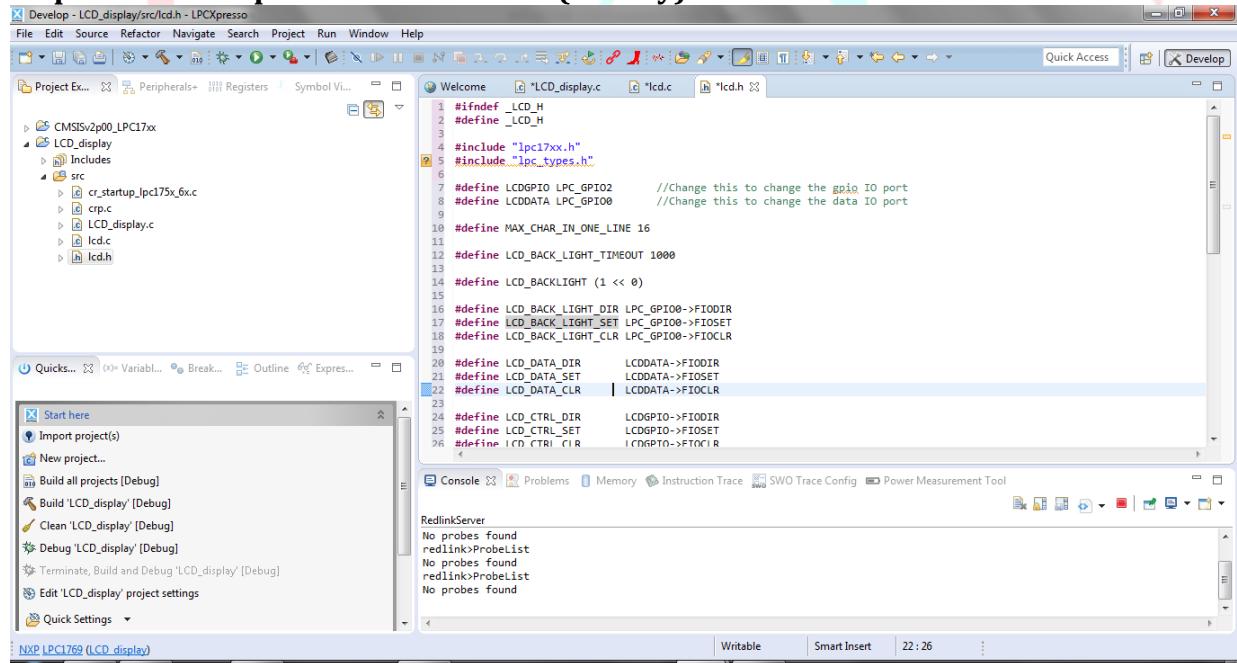


Figure 17

Step 18: After writing code and adding libraries, Build the project by clicking on Build LCD_display on the Quickstart Panel on the bottom left of the window.

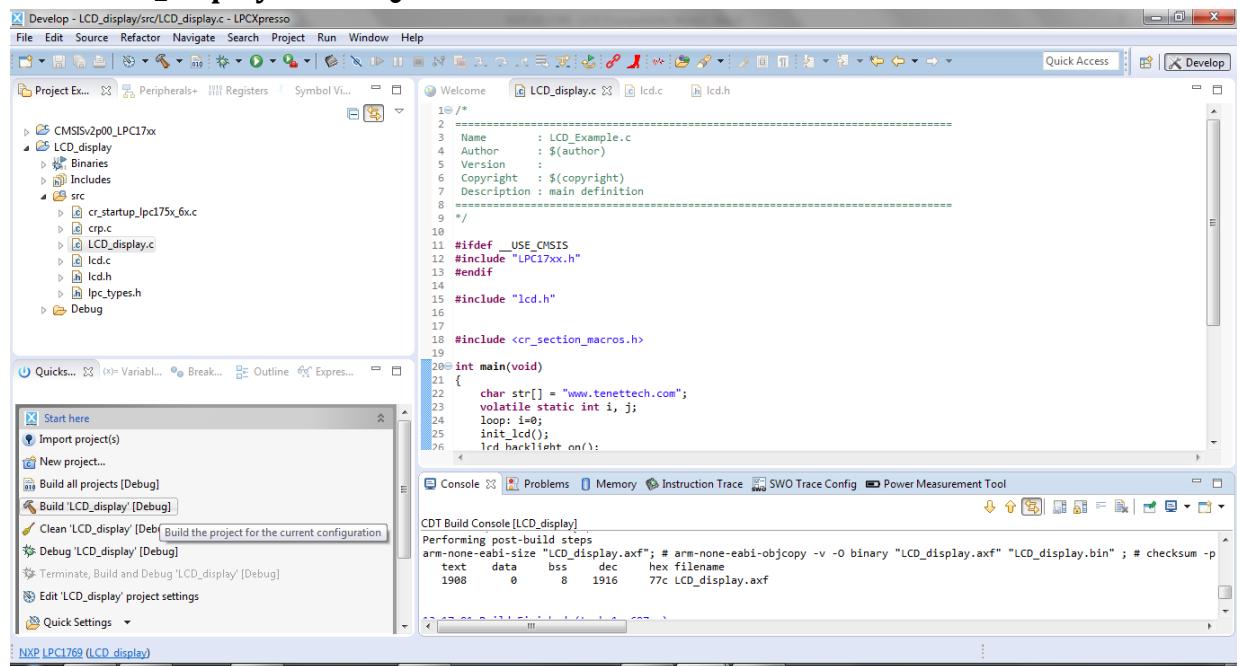


Figure 18

Step 19: Now, if all goes well connect the Micro B cable to LPC1769 and connect it to your computer. To upload the project file, click on the Program flash.

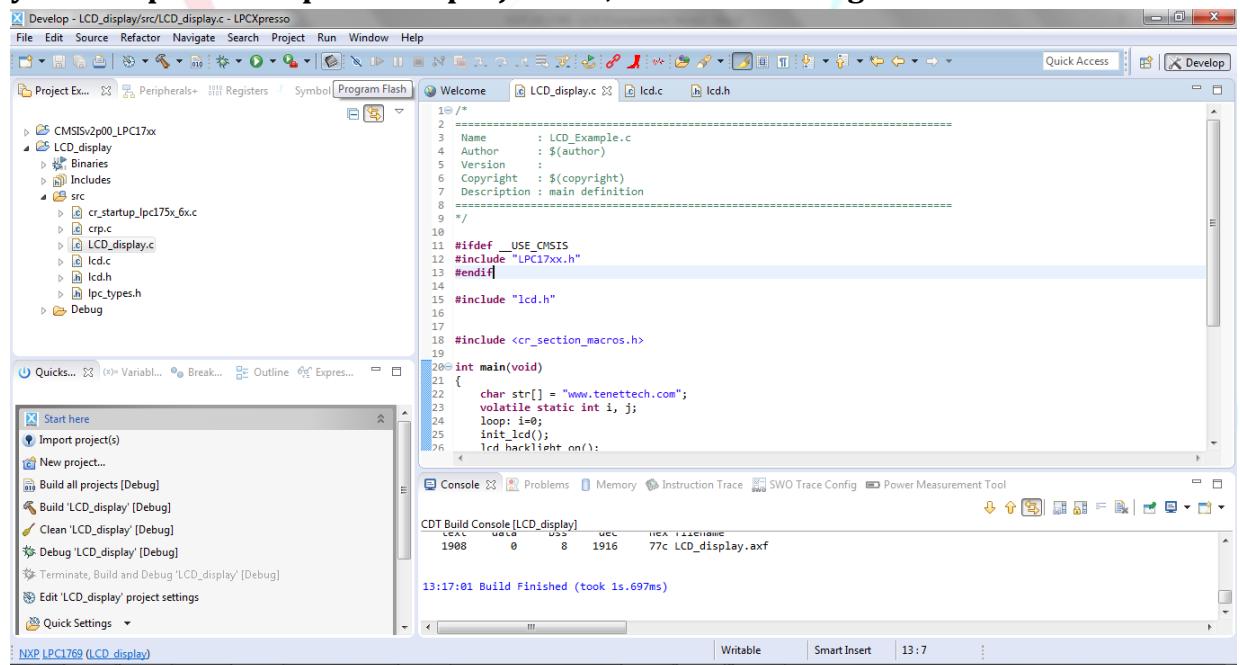


Figure 19

Step 20: Now select the Project file LCD_display.axf. We can find it in our project folder.

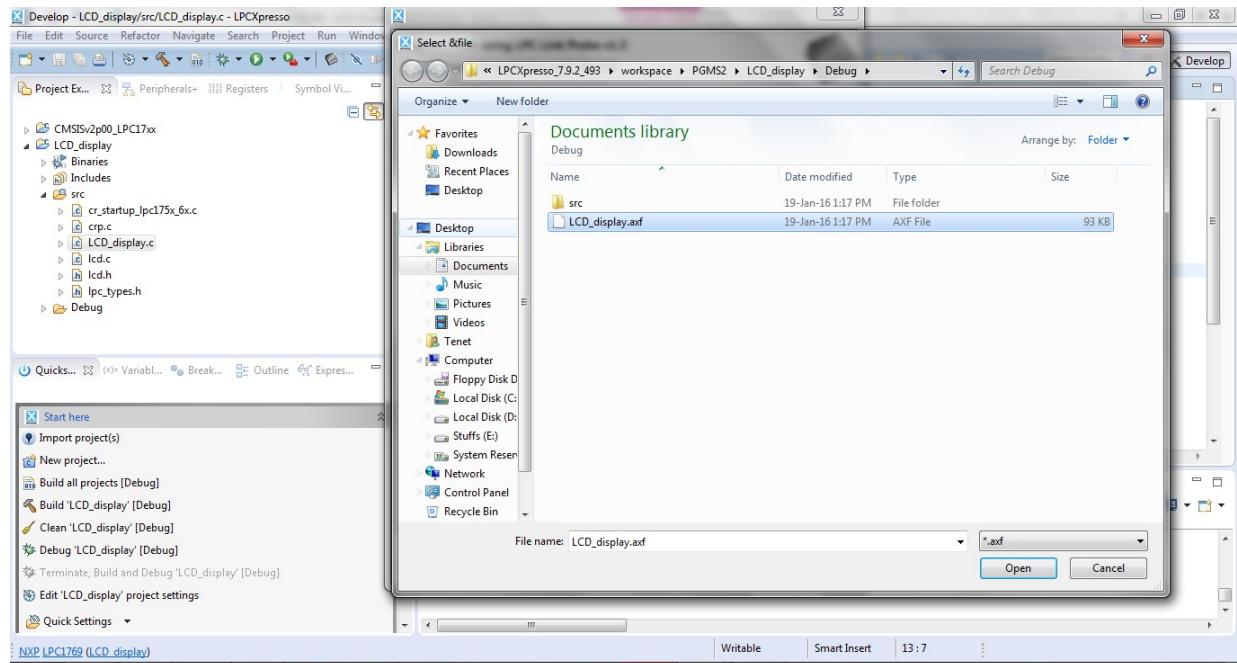


Figure 20

Step 21: Now this window shows we have finally dumped our project onto LPC1769.

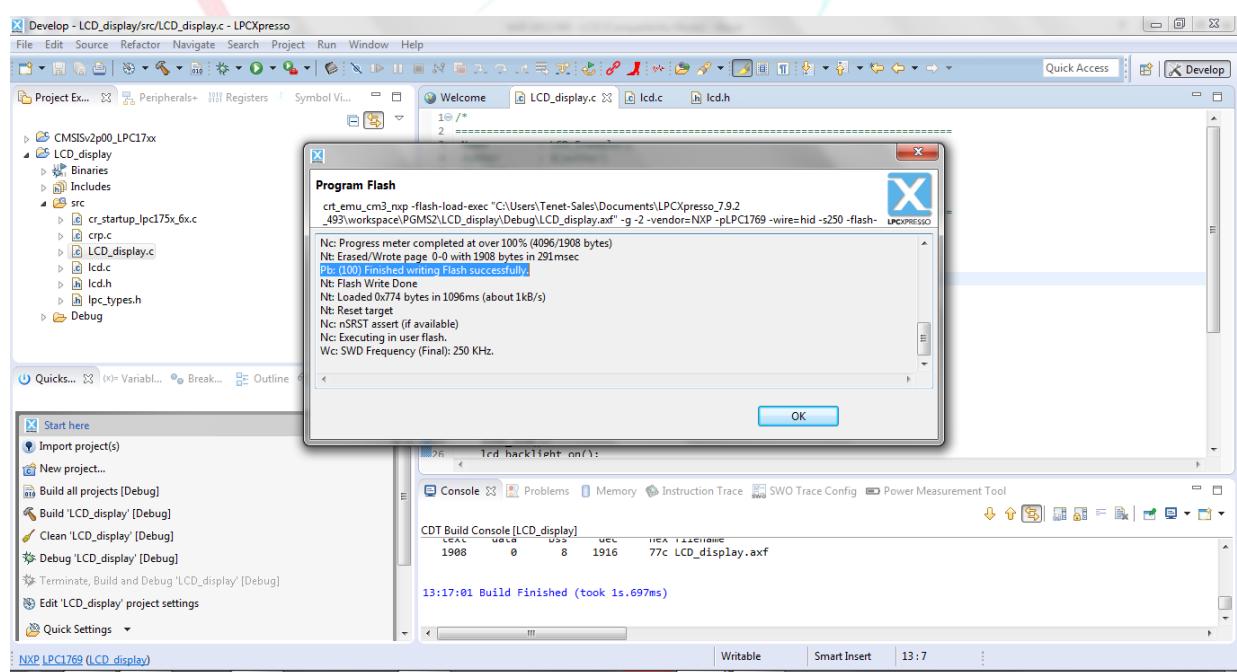


Figure 21

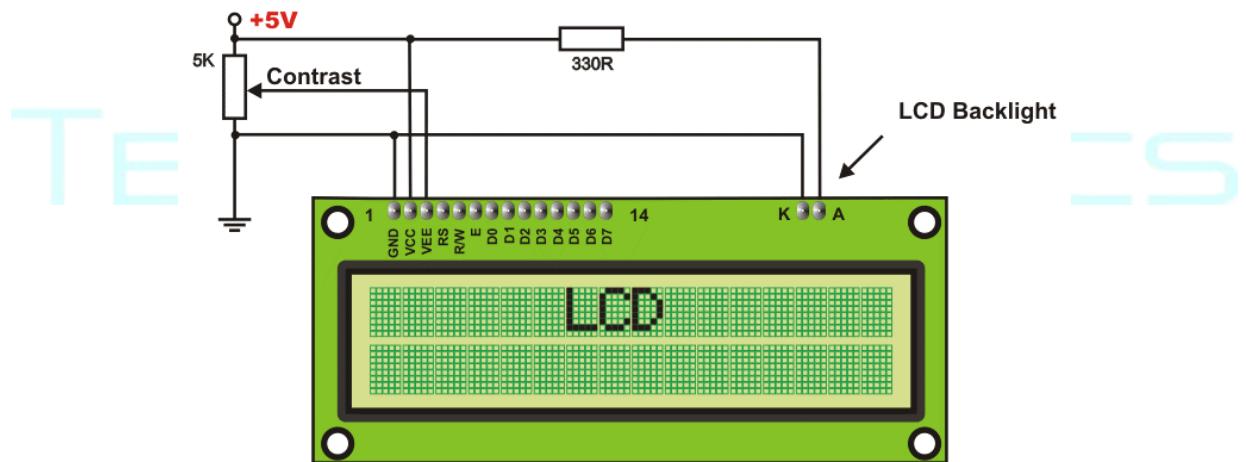
Hardware Requirements:

- **LPC1769 Board**
- **2x16 Character LCD**
- **breadboard**
- **Jumper wires**

16x2 character LCD pin-out



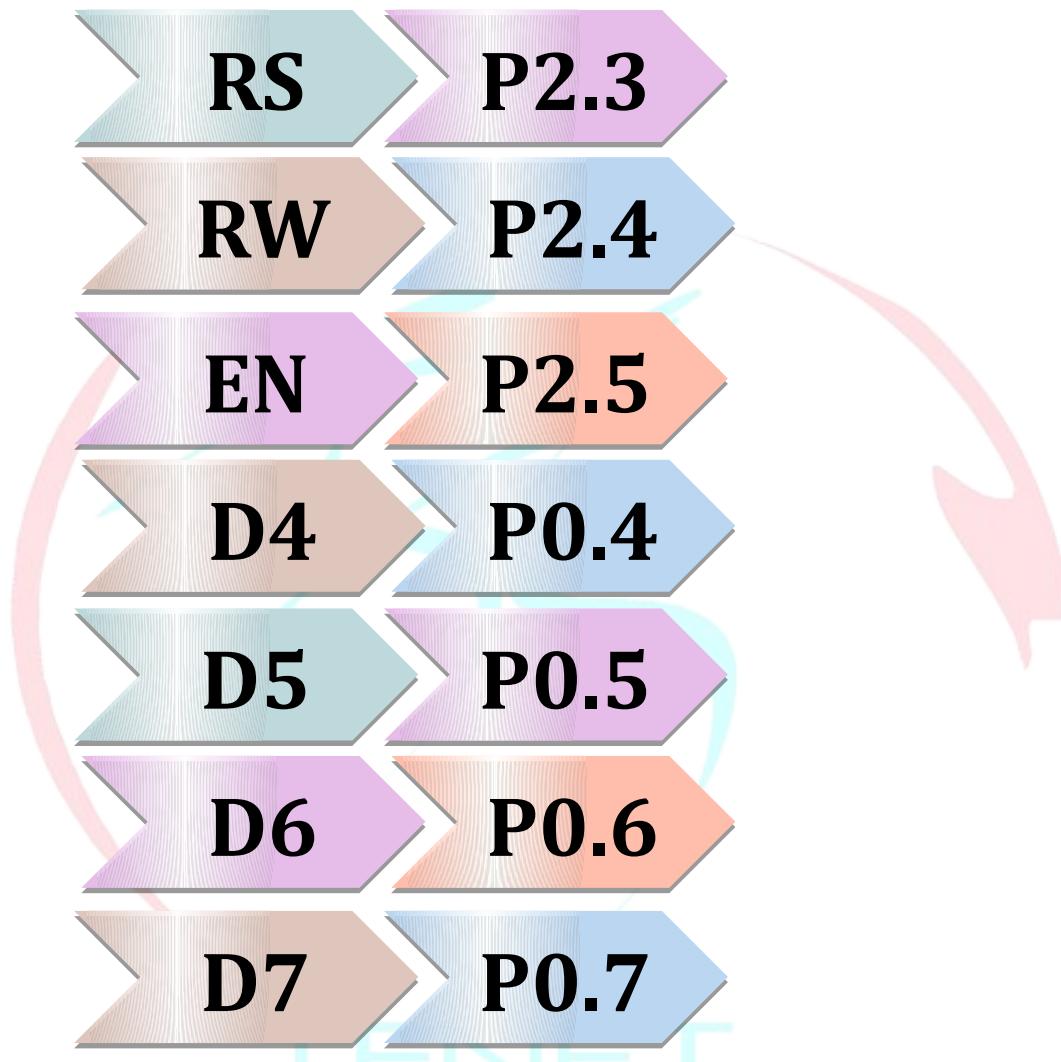
Minimum connection required



Commands for LCD

<i>Sl. no.</i>	<i>Instruction</i>	<i>Hex</i>	<i>Decimal</i>
1	Function Set: 8-bit, 1 Line, 5x7 Dots	0x30	48
2	Function Set: 8-bit, 2 Line, 5x7 Dots	0x38	56
3	Function Set: 4-bit, 1 Line, 5x7 Dots	0x20	32
4	Function Set: 4-bit, 2 Line, 5x7 Dots	0x28	40
5	Entry Mode	0x06	6
6	Display off Cursor off (clearing display without clearing DDRAM content)	0x08	8
7	Display on Cursor on	0x0E	14
8	Display on Cursor off	0x0C	12
9	Display on Cursor blinking	0x0F	15
10	Shift entire display left	0x18	24
11	Shift entire display right	0x1C	30
12	Move cursor left by one character	0x10	16
13	Move cursor right by one character	0x14	20
14	Clear Display (also clear DDRAM content)	0x01	1
15	Set DDRAM address or cursor position on display	0x80 + address *	128 + address*
16	Set CGRAM address or set pointer to CGRAM location	0x40 + address **	64 + address**

Interfacing 16x2 character LCD with LPC1769 pin-out



TENET
TECHNETRONICS

OUTPUT:

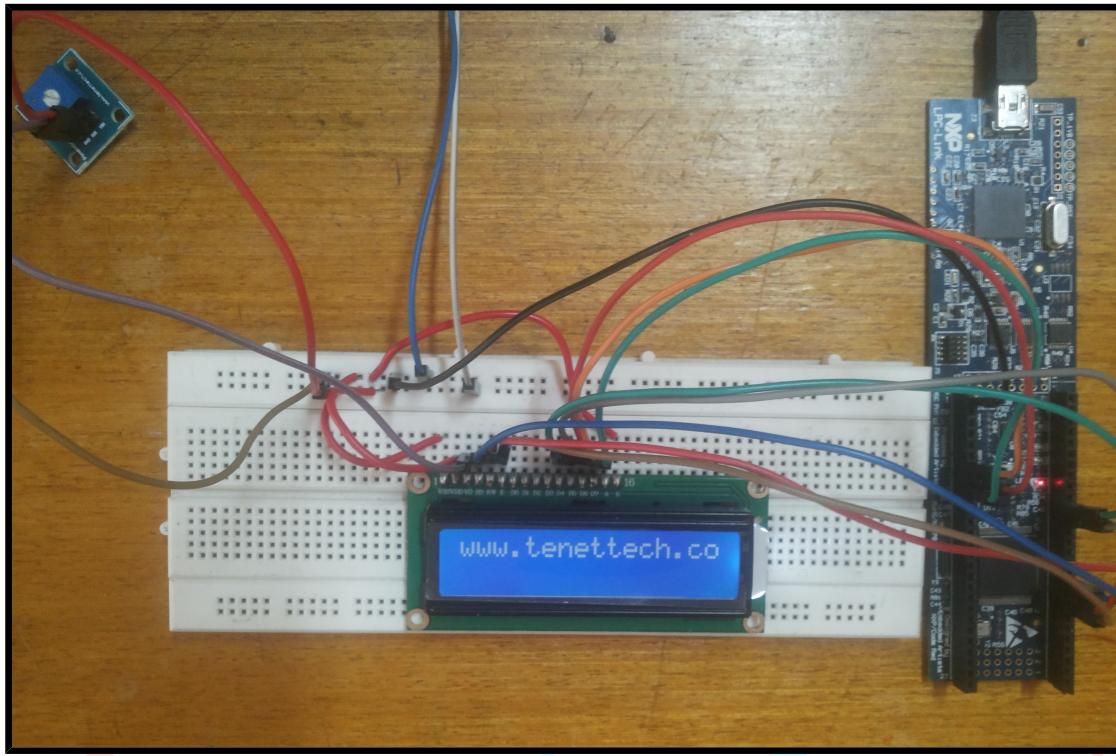


Figure 18

For product link:

1. <http://www.tenettech.com/product/1548/lpc1769-lpcxpresso-board>
2. <http://www.tenettech.com/product/2442/16-x-2-character-lcd-display-with-backlight-jhd162a-green>
3. <http://www.tenettech.com/product/6068/power-supply-breakout-board>

TECHNETRONICS

For more information please visit: www.tenettech.com

For technical query please send an e-mail: info@tenettech.com