**LCD COMMANDS**

InitLCD(LS\_BLINK|LS\_ULINE);

Initialize the LCD with blinking cursor and underline cursor on.

InitLCD(LS\_BLINK);

Initialize LCD with blinking cursor on.

InitLCD(LS\_ULINE);

Initialize LCD with underline cursor on.

InitLCD(LS\_NONE);

Initialize LCD with cursor off.

LCDClear();

Clear total LCD and bring the cursor back to Home position (0, 0).

LCDGotoXY(11, 1);

Move the cursor to (11, 1) i.e. 12th column of second line. And then anything you write to LCD will be printed at (11, 1).

LCDWriteString("Text");

Print string(Here “Text”)on home position (Column0, Row0).

LCDWriteStringXY(0, 1,"Text");

Print a string(Here “Text”) on Column0, Row1 (2nd line).

LCDWriteInt(i, 4);

Print value of an integer (Here i) at Home position as four digits (If i = 50, it will print 0050).

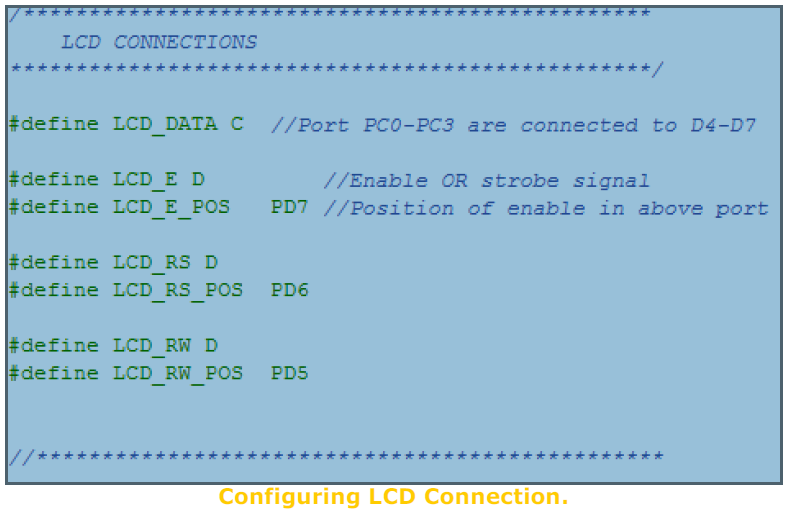
LCDWriteIntXY(9, 1, i, 3);

Print value of an integer (Here i) at Column9, Row1 as three digits (If i = 50, it will print 050).

**LCD ADVANCE CONNECTIONS**

To connect the LCD module to some different I/O ports, just do the following:

1. Open “lcd.h” file.
2. Go to section “LCD Connections”.



1. Set LCD\_DATA to the port where you have connected the LCD data lines. Data lines must be connected to any port say PORTB starting from PIN-0 to PIN-3. i.e. if you set

#define LCD\_DATA B

You should connect

PORTB.0->DATA4

PORTB.1->DATA5

PORTB.2->DATA6

PORTB.3->DATA7

1. Now set the port where you have connected the LCD’s “E” signal. In example it is PORTD. So

#define LCD\_E D

1. Then specify to which PIN of PORTD it is connected, this is done by

#define LCD\_E\_POS PD6

So “E” PIN of LCD is connected to PORTD-6

1. In the same way, set RS and RW signals. And that’s all.