Embedded System

Assignment - 2

Amar kumar

CED17I029

Problem Statement - Design a 16x2 LCD driver program for Arduino.

Code Explanation

```
1  #include <LiquidCrystal.h>
2  // initialize the library with the numbers of the interface pins
3  LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

Line 1 - This is a library where various functions related to LCD is available

Line 3 - This is creation of an object named as **"lcd"**. Where 1st argument is for Register select signal, 2nd argument is for Read/write select signal, 3rd to 6th argument is Data Bus lines which indicates that i am using 4 bits interface.

```
4  // output control pins
5  #define LCD_CONTRAST_PIN 6
```

Line 5 - PIN 6 of arduino is used for changing brightness of text. So i am defining PIN 6 of arduino using LCD_CONTRAST_PIN

```
6  void setup() {
7    // set up the LCD's number of columns and rows:
8    lcd.begin(16, 2);
9    analogWrite(LCD_CONTRAST_PIN, 0);
10 }
```

Line 8 - This tells us that we are using LCD where number of rows are 2 and number of columns are 16. We also have other dimensions of LCD like 20x4 etc. but 16x2 is used more as compared to 20x4.

Line 9 - This tells that write LCD_CONTRAST_PIN i.e. PIN 6 of arduino with 0 so that text displayed is clearly visible i.e. of white color. As the value inside this pin is increased, it becomes darker and after some time it starts vanishing.

```
14    lcd.print ("Amar kumar");
15    lcd.setCursor (0,1);
16    lcd.print ("CED17I029");
17    delay(3000);
18    lcd.clear ();
```

Line 14 - This basically prints "Amar kumar" without quotes on the LCD and by default it start printing from 0th row and 0th column index.

Line 15 - In this line cursor is being moved to 1st row and 0th column index so that "CED17I029" prints in the next line after "Amar kumar".

Line 16 - This is same as line 14. It prints "CED17I029" on LCD.

Line 17 - This is a function which is used to generate a delay of given argument in milliseconds. Here 3000 milliseconds delay is being given.

Line 18 - This is used to clear the LCD screen.

```
20    for(int i=0 ; i<5 ; ++i){
21       lcd.setCursor(0, 0);
22       lcd.print ("Blinking Text");
23       delay(500);
24       lcd.clear ();
25       delay(500);
26    }
```

This is basically a loop which is being used to Blink the text 5 times.
Main concept behind this is firstly, i am setting the cursor position and then printing the text "Blinking Text" then i am taking a delay of 500 milliseconds so that it could be visible. After that i am clearing the screen and again taking a delay of 500 milliseconds. In this way, the text appears for 500 ms and then disappear for 500 ms and we are able to see the blinking text.

```
28    for(int i=0 ; i<=11 ; ++i){
29       if(i<=5){
30          lcd.setCursor(i, 0);
31          lcd.print ("Moving Text");
32          delay(500);
33          lcd.clear ();
34       }
35       else{
36          lcd.setCursor(i-6, 1);
37          lcd.print ("Moving Text");
38          delay(500);
39          lcd.clear ();
40       }
41    }
```

In this loop, i am scrolling the text named "Moving Text".

Concept is that i am first setting the cursor position at 0th row and 0th column then i am printing the text and keep it for 500ms then clearing the text and increasing the column index by 1. I am not giving a delay after clearing the text here because if we give delay after clearing text then it will also look like blinking and moving forward and here i am only scrolling the text. After completing 1st row its row index is being increased by 1 so that the text moves to the next line and start scrolling. Value of cursor position is taken according to the text i am moving so that the text does not go out of display of LCD. For this example, my text consists of 11 characters. So i am moving it 5 columns further and then increasing row value by 1, so that the text moves in the next line and start scrolling. Here, it also moves 5 columns ahead so that the text does not go out of screen. To move text faster, we can decrease the delay value and to slow it down, we can increase the delay value.

```
43    lcd.setCursor(0, 0);
44    lcd.print("Static Text");
45    delay(5000);
46    lcd.clear();
```

This is basically printing a text at 0th row and 0th column index and keeping it for 5000 milliseconds so that it could look like a static text. For text to visible for more time, we can increase the delay.

```
48    lcd.print("Dimming Text");
49    // loop through values for contrast
50    for (int i=0; i <= 150; i++){
51      analogWrite(LCD_CONTRAST_PIN, i);
52      lcd.setCursor(0, 1);
53      //lcd.print(i);
54      delay(80);
55    }
56
57    lcd.clear();
```

This is firstly printing the text "Dimming text" and then we are decreasing the brightness by increasing the value inside the PIN 6 of arduino at an interval of 80ms with the help of analogWrite() function.

About Some useful functions
lcd.begin() - Initializes the interface to the LCD screen, and specifies the dimensions (width and height) of the display

analogWrite() - This function is used to set a value to the PIN of arduino. It takes two arguments. First is the PIN number to which we want to set the value. Second is the value which we want to set to the given PIN number in the first argument.
lcd.print() - Prints text to the LCD.

lcd.clear() - Clears the LCD screen and positions the cursor in the upper-left corner.

lcd.setCursor() - Position the LCD cursor; that is, set the location at which subsequent text written to the LCD will be displayed. First argument is column and second argument is row starting from 0th index.

About PINS of LCD

1	VSS	GND Arduino	Signal ground
2	VDD	5 V Arduino	Logic power for LCD
3	V0	10 kΩ potentiometer	Contrast adjustment
4	RS	Pin 2 Arduino	Register select signal
5	R/W	GND Arduino	Read/write select signal
6	E	Pin 3 Arduino	Operation enable signal
7 – 14	D0 – D7	–	Data bus lines used for 8-bit mode
11 – 14	D4 – D7	Pin 4 – 7 Arduino	Data bus lines used for 4-bit mode
15	A (LED+)	5 V Arduino	Anode for LCD backlight
16	K (LED-)	GND Arduino	Cathode for LCD backlight