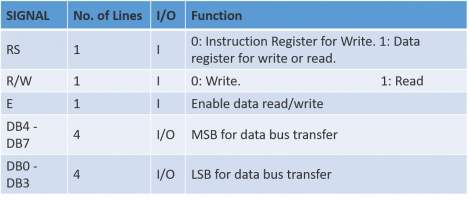
LCD , LPC 1768 interfacing

The 16 x 2 character LCD display is one of the most basic display module used by beginners while learning embedded systems.  
16 characters can be displayed on the screen at a time across two lines. Each of these 16 characters is displayed using pixels that are arranged in a 5 x 7 matrix format.

Features

* The LCD module has 16 pins.
* It can be configured either in 4-bit or 8-bit mode. The 4-bit mode helps in saving an extra 4-pins in the circuitry.
* Controller and driver are automatically reset after power on.
* Instruction functions that include: Display clear, cursor home, display on/off, cursor on/off, display character blink, cursor shift, display shift



Registers

The two 8-bit registers in the HD44780U 16 x 2 character LCD display are

* Instruction Register (IR)  
  Instruction functions are written from the micro-controller to this register for Display Data RAM (DDRAM) and Character Generator RAM (CGRAM).
* Data Register (IR)  
  This register temporarily holds the data to be written into the DDRAM or the CGRAM from the micro-controller. It also temporarily holds the data that is being read from the two registers.

Programming

Interfacing the LCD module with any micro-controller is very easy. It involves only a few functions that instruct the LCD to perform specific tasks and read or write data in the LCD.  
Here, we will be doing an 8-bit interfacing with the LPC1768 micro-controller and only perform a write operation on the LCD.

**DEFINITIONS:**

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Default DATA PORT in OpenLab is PORTB. P0[16:23]

// Default CONTROL PORT in OpenLab is PORTC P1[16:23]

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#define DATA\_BUS LPC\_GPIO0->FIODIR

#define CONTROL\_BUS LPC\_GPIO1->FIODIR

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Control Port definitions

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#define CONTROL\_SET LPC\_GPIO1->FIOSET

#define CONTROL\_CLEAR LPC\_GPIO1->FIOCLR

#define RS 18

#define RW 17

#define EN 16

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Data Port definitions

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#define DATA\_SET LPC\_GPIO0->FIOSET

#define DATA\_CLEAR LPC\_GPIO0->FIOCLR

#define DATA0 16

#define DATA1 17

#define DATA2 18

#define DATA3 19

#define DATA4 20

#define DATA5 21

#define DATA6 22

#define DATA7 23

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//Control and Data Mask

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#define CONTROL\_MASK ((1<<RS)|(1<<RW)|(1<<EN))

#define DATA\_MASK ((1<<DATA7)|(1<<DATA6)|(1<<DATA5)|(1<<DATA4)|(1<<DATA3)|(1<<DATA2)|(1<<DATA1)|(1<<DATA0))

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void INIT\_PORTS()

{

DATA\_BUS = DATA\_MASK;

CONTROL\_BUS = CONTROL\_MASK;

INIT\_LCD();

}

void SEND\_CMD(unsigned char cmd)

{

SEND\_BITWISE(cmd);

CONTROL\_CLEAR = (1<<RS)|(1<<RW);

CONTROL\_SET = (1<<EN);

delay\_ms(1);

CONTROL\_CLEAR = (1<<EN);

}

void INIT\_LCD()

{

SEND\_CMD(0x38); //Initialize LCD in 8-bit mode. 2 Lines. 5\*7 mode

SEND\_CMD(0x0C); //Display ON cursor OFF

SEND\_CMD(0x01); // Clear display

SEND\_CMD(0x06); //Shift cursor to the right

SEND\_CMD(0x80); //Force the cursor to the beginning of the line

}

void SEND\_CHAR\_DATA(unsigned char data)

{

SEND\_BITWISE(data);

CONTROL\_CLEAR = (1<<RS)|(1<<RW);

CONTROL\_SET = (1<<RS)|(1<<EN);

delay\_ms(1);

CONTROL\_CLEAR = (1<<EN);

CONTROL\_CLEAR = (1<<RS);

}

Another write function is created in order to send a string of data to the LCD controller.

void SEND\_STRING\_DATA(char \*str)

{

while(\*str!='\0')

{

SEND\_CHAR\_DATA(\*str);

str++;

}

}