

Lab 5 and 6: Working with Parallel Bus Devices – Programming an LCD Display

CompEng 3151: Digital Engineering Lab II

Last revised: July 16, 2019 (BJZ, RJS)

Description/Overview

There are a number of different peripherals that you can interface with a microcontroller. Each of these peripherals can communicate with the AVR can use serial or parallel data busses and various communication protocols (such as I2C or SPI). For this lab, you are going to connect an LCD module up to the AVR Simon Board and communicate with it using a parallel data bus to write to its controller. When you start working with LCD modules, you will start feeling the real power of MCU and your imaginations will be touching the sky, and you will wonder how many exciting, powerful gadgets you can create.

One crucial skill that you need to develop is the ability to read and interpret datasheets. You will be provided with the datasheet for the LCD module, which contains all of the details of the set of commands to be executed to initialize the display and write specified characters to it. Using the information from the datasheet and your knowledge of the AVR Simon Board, you will write a C program to display your name and other information on the LCD panel.

Objectives

Students will:

- Read datasheets and schematics to connect the AVR microcontroller to an 8-bit LCD controller
- Create a C program to interface the AVR microcontroller to the LCD controller and display a simple message (“Hello World”)
- Understand how to change the configuration of the LCD and display multiple line messages

Materials Required

- Atmel Studio 7 Installed on PC (ECE CLC Computers will have this)
- AVR Simon Board with a USB cable
- 2x16 Character LCD Module
- Connection Wires

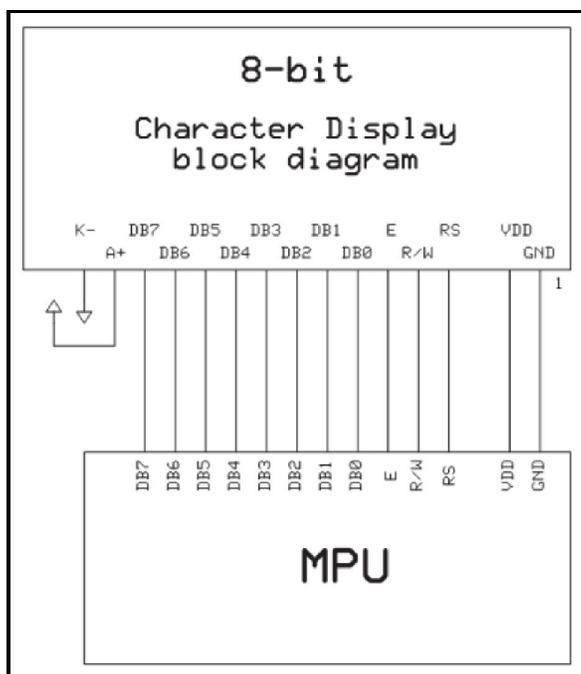
Preparation

In order to be successful with this lab, students should review Chapter 12 in the Mazidi textbook and the datasheet for the LCD module.

Background

LCD, which stands for Liquid Crystal Display, is an electronic device which is used to display data. LCDs are preferable over seven segments and LEDs as they can easily represent data in the form of characters, numbers or animations. LCDs are very easy to program and make your work quite attractive and simple. Numerous types of LCDs are available in the market such as 16x2, 16x4, 20x2, 20x4, graphical LCDs (128x64) etc. The LCD which we are using is 16x2 alphanumeric LCD, which means it displays 16 characters in 2 rows or a total of 32 characters at a time.

The LCD can interface with AVR microcontroller by using two modes – 4-bit mode or 8-bit mode. In this lab, we will use 8-bit mode for interfacing. In 8-bit mode, we send commands data to the LCD by using eight data lines (D0-D7), while in 4-bit mode, we use four data lines (D5-D7) for sending command and data. These data lines can be connected to any port of Atmega324PB.



Connecting the LCD

One of the first things you'll need to do is connect the LCD to the microcontroller. Since we are using 8-bit mode, we will need a full port to send/receive data from the LCD controller. You'll want to choose one of the ports on the AVR and connect your LCD to it. Make sure the individual port bits match up to the corresponding bits on the LCD controller (i.e. PA0 = DB0, PA1 = DB1, etc.).

Communicating with the LCD Module

When you look at the datasheet, you may notice some timing diagrams that show how to read and write to the LCD module and be confused. Basically, the LCD module has two registers – command and data. When we are giving commands to the LCD module, we select the command register, and when we are sending data to LCD for display, we select the data register. Commands are instructions given to LCD in order to perform required functions according to the given command. In order to display textual information, data is sent to LCD. The Register Select (RS) pin in the LCD module is used to select the appropriate register (RS: 0 = Command, 1 = Data)

Another pin you'll want to make a note of is the R/W pin. It is used to tell the LCD module whether you are reading data from it or writing data to it. The most common read operation for the LCD module is to check its Busy Flag to see if it is ready for receiving data or not. Most of the operations we will do, however, will be writing command or data to it. (RW: 0 = Write, 1 = Read).

The final pin used when communicating with the LCD module is the EN pin. This pin is used to enable read/writes to the module. Think of it as the enable pin on a series of latches or a shift register. When EN is

asserted HIGH, it will enable the read or write operation (as specified by the RW pin) of commands or data (as specified by the RS pin) by shifting the data in or out of the parallel data bus.

Configuring the LCD Module

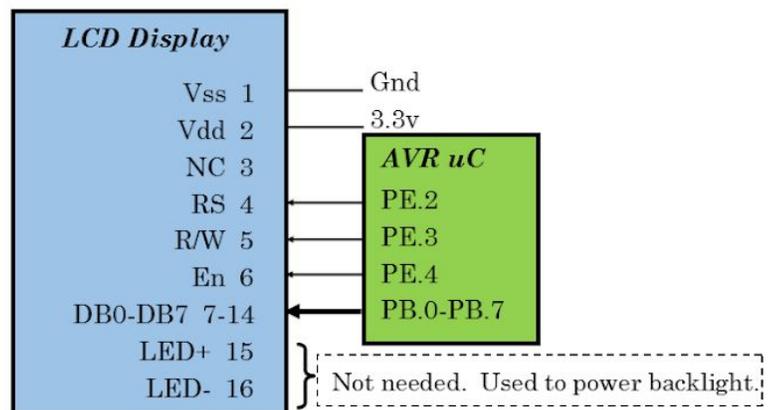
To get started using the LCD module, you'll need to configure the LCD. These are the general steps to initializing the LCD module (hint: you might want to create an initialization function to complete these):

- Initialize the LCD in 8-bit Mode, 2 Line, 5x7 pixel
- Set the Entry Mode – Cursor Increments by Cursor Shift
- Turn the Display On, Cursor Off
- Clear the LCD
- Return the Cursor to Home
- Put the Cursor at the Start of the 1st Line

Remember that in order to write these “commands” to the LCD module, you'll need to set the configuration pins appropriately. After configuring the LCD module, data can be sent and displayed on the LCD freely.

Procedure

- 1) Obtain an LCD module from your TA and some wires and connect the LCD module to the AVR Simon Board. Since we don't need to use the LEDs on the board, move SW12 into the MUX position so that you can use all of the ports along the expansion header on the board. You'll want to connect your LCD data bus to one of the ports that has all 8-bits on the expansion header Port D should work for this. Three other control lines will need to be connected to I/O pins, as well (Suggestion: use port E pins on the expansion header for these).



- 2) After getting the LCD module connected to your AVR Simon Board, write a program to display “Hello World!” on the 1st line of the display. Show this to your TA before proceeding.

NOTE TO GTAS: REQUEST 12 ADDITIONAL SIMON BOARDS FROM MR. YOUNGER. AS STUDENTS CORRECTLY COMPLETE THIS PORTION OF THE LAB, KEEP 12 TOTAL SIMON BOARDS WITH THE LCD PROPERLY WIRED FOR THE FOLLOWING PORTIONS OF THE LAB.

- 3) Modify your program to display your name on the first line of the display and display the name of your favorite undergraduate course on the second line.
- 4) As a final modification, find a way to change your code to get the displayed message to scroll left or right.

Deliverables

The code you created in this lab will serve as a class for future labs where you will utilize the LCD. Make sure the code is documented thoroughly. This code will be submitted as a part of a lab report for the work you completed. In the lab report, the questions/observations below should be included in the results or conclusions section of your report.

Questions/Observations

1. Describe the important control pins and initialization instructions of the LCD module.
2. Is the LCD module edge-triggered or level-triggered? Why?
3. Describe any issues you had getting your name to display or getting the message to display. What was the issue and what did you do to fix them?

References

- Muhammad Ali Mazidi, Sarmad Naimi, and Sepehr Naimi. 2010. *AVR Microcontroller and Embedded Systems: Using Assembly and C (1st ed.)*. Prentice Hall Press, Upper Saddle River, NJ, USA.
- LCD Module datasheet