The purpose of this lab is to become familiar with the Motorola 68HC11 Lab Trainer kit, related software (Buffalo monitor and TExaS simulator) and learn to program the system. In addition to the basics of the EVBU board, the simulation technique TExaS is introduced in this lab. This will be used to create a functional code that will display messages on the LCD display of the microcontroller.

**Objectives**
- Understanding the CME11E9-EVBU Development Board from Axiom Manufacturing,
- Assembling a few pre-written programs so that you can learn hot to run it on the TExaS simulator
- Becoming familiar with the Keypad and LCD display mounted on the Axiom EVBU, including:

**Prelab:**
Go through the TExas Tutorial file on which gives a brief introduction to the TExaS simulator.
Read the help section in the THR SIm11 software to get a brief idea about the simulator before entering the lab.
Answer the following questions:
   1. List the different assembly directives available for 68HC11 and explain their functions.
   2. List the different types of instructions for 6811 with a few examples for each type.
   3. What are subroutines in an assembly language program?
   4. Draw a memory allocation map for this axiom board.
   5. What is the reason for starting the program sections from $2000$?

**Tasks for the Lab:**
There are a few short programs in this lab that you are required to run on the EVBU board. Go through the instructions given in the Tips for the lab.pdf document, which will help you run the programs on the EVBU board. The TA will also assist you in running the programs.

**Task 1:**
Run the program lab1.1.rtf on the EVBU board using TExaS. Follow the instructions given in Tips for the lab.pdf to run the program.
This program is designed to help you learn two tasks
• How to clear the LCD display.
• How to initialize the pins of a port as output pins. Here PORTD is used as an example.

Notice that in the beginning of the program labels are assigned to a few LCD command codes. These commands are used to control the cursor position on the display and also the character position. These labels make it very convenient for the user to use the different commands in a program frequently. Refer to this link to check out the various commands LCD command and character codes available.

Similarly there are other labels given to various addressed which will be used in the program very frequently.

Note that the program should start from $2000 (RAM) in order for it to run on the board. Using this address you cannot simulate the program on TExaS.

The program starts with clearing the stack and disabling all the interrupts. It is a good practice to do this in the beginning of every program. Follow the comments given in the program to understand what each instruction does.

There are separate subroutines which do the job of clearing the LCD and also initialize the PORTD. These subroutines are called in the main program and once their job is done the program should be ended. Usually this is done using a reset vector while simulating a program on TExaS. The reset vector brings the control back to the beginning of the program once it ends. Since, we cannot use this while running the program on the board; the control is set to follow an infinite loop at the end of the program. The line ‘here: bra here’ is used in the current program to for this purpose.

There are also two other subroutines used in the program viz. ‘short_delay’ and ‘long-delay’. These are used to set delays whenever needed. A down counter is created to cause the delay. The value of the counter determines the amount of delay.

Show the output to the TA and get their signature.
TA Initial _______________

**Task 2:**
Run the program `lab1.2.rtf` on the EVBU board using TExaS.
The program shows you how to display a single character on the LCD panel of the board.

• The first step is to initialize the LCD so that anything that is left from the previous program is cleared out.
• Then, to display a character it is sent to the accumulator and then to the LCD data register which displays the character.
• Note that when displaying any character it is sent in the ASCII format and that is the reason ‘ ‘ are used with a # sign.

Get the TA initial after running the program__________________
Task 3:

Run the program `lab1.2.1.RTF` on the EVBU board using TExaS. The program shows you how to display a string of characters on the LCD panel.

- The first step is to assign a label to the particular message to be displayed and store it in a memory location.
- The next step is to start the main program and clear the LCD.
- Next, there is a subroutine called 'cputs' in which the string is retrieved from the memory locations and sent to the LCD data register one character at a time and displayed.
- One character of the string is accessed at a time and sent to LCD data register through ‘putc’ subroutine.
- When the end of string has been reached, the control jumps back to the main program.
- Note that the address of each line on the LCD has been labeled in the beginning of the program.

Show the output to the TA and get their initial ________________

Task 4:

Run the program `lab1.3.rtf` on the EVBU board using TExaS. This program shows you how to display a character on a particular line on the LCD panel.

- The first step is to select which line on the LCD panel you need to display a character.
- The next step is to get the address of that particular line and send it to the LCD command register.
- This places the cursor at the beginning of that line.
- The subroutine ‘csr_to_ln1’ and other similar subroutines do the job of getting the address of the corresponding lines and sending it to the command register. Thus, when you need to display something on a particular line, call the corresponding subroutine.
- Once the cursor is placed on the line, the next step is to send the character to be displayed to the LCD data register.
- Please follow the comments beside the line to understand the sequence of events.

TA initial: ________________

Task 5:

Run the program `lab1.3.1.rtf` on the EVBU board using TExaS.
This program also displays a string of characters on the LCD panel. Isn’t this similar to task 3? Yes, but there is a slight difference in that the subroutine ‘put_lcd’ is used instead of ‘putc’ to display each character of a string.

**Question:** Explain the difference between the results that you obtained in task 3 and task 5. Explain what difference ‘put_lcd’ makes and how so?

Get TA initials after you run the program and also answer the question
TA initial: ______________

**Task 6:**

Run the program [lab1.4.rtf](#) on the EVBU board using TExaS.

This program shows you how to get any character from the keyboard and display it on the LCD panel.

- A table is created in the beginning of the program which includes all the keys from the keypad, in the ASCII format (Eg: ‘A’). This table is used for future reference in the program.
- PORT D and PORT E are used for communication between the controller and the keypad.
- The PORT D (bits 2, 3, 4, 5) acts as the output port used to enable the rows of the keyboard.
- PORT E (bits 0, 1, 2, 3) is the input port used to get a signal from the columns of the keypad.
- The ‘anykey’ subroutine detects if a key is depressed on the keypad. This is done by enabling all the rows and getting a signal from the particular column which consists of the depressed key.
- The ‘get_row_column’ subroutine detects the row and column corresponding to the key depressed. This is done by enabling one row at a time and checking if any column returns a signal (‘get_column’ subroutine) from that row.
- The identification of the key is done using the TABLE as reference. The ‘key_lookup’ subroutine compares the key obtained from the keypad with every character in the TABLE and hence identifies it.
- The ‘putc’ subroutine is used next to display the key on the keypad.
- The entire process is repeated until the user releases his hand from the keypad.

**Question:** Write a detailed flowchart for this particular program and hand it over to the TA at the beginning of the next lab session.