LCD

EVBU LAB EXPERIMENT

_____

Class

_____

Instructor / Professor

2813 Industrial Ln. • Garland, TX 75041 • (972) 926-9303 FAX (972) 926-6063
support@axman.com Rev 1.00 • web: http://www.axman.com

CONTENTS

1 GETTING STARTED

The following section has been designed to help users to quickly learn proper setup and operation of the lab experiment.

1.1 Introduction

The experiment requires a single board development system that is fully assembled and fully functional from AXIOM MANUFACTURING. Development board CME11E9 EVBU is supported in this experiment. The system comes complete with instructions. All software, drawings, and manuals are contained on the CD.

1.2 Software

The CD comes with AxIDE, which is an integrated development environment designed exclusively for use with AXIOM development boards, providing an interface to programs running on these boards. AxIDE also makes uploading programs and easy via the COM port. Read your board manual for setting up the AxIDE.

1.3 Support Software

There are many useful programs on the included CD that can make developing projects easier. The CD also contains example software programs for this experiment on each board. You can also download the latest software free from our web site at:
http://www.axman.com.

1.4 Hardware

The following hardware is required:

AXIOM CME11E9 EVBU
Windows based PC
AXIOM LCD

2.0 Visual

Devices used in this lab are static sensitive and are easy damaged by mishandling. Use caution when installing wires and devices on the breadboard to prevent the bending of leads. Experiments should be laid out in a orderly fashion. Start your lab time with the bench clean and free of metal objects. Leave the lab area in a clean condition by picking up loose parts, wires and small objects.

3.0 Theory

AXIOM's development board is designed for quickly and effectively learning the basics of microcontrollers. This lab will walk the student though the steps of using the development board for its intended purpose, controlling devices. A LCD is a display device that is commonly used by a microcontroller for displaying instructions or results. In this lab, an AXIOM LCD is used for the experiment. The microcontroller will interface to the LCD though

the external bus at addresses $B5F0 and $B5F1. Data bits D0 – D7 are read or written to the LCD and are controlled by the R/W signal. When the R/W is high, it reads the display. A 8 bit value is written to display when R/W is low. The display is powered from the +5v on board the EVBU. A bias voltage controls the LCD contrast. On the EVBU, this bias is preset by a 100 ohm resister. Address A0 is applied to the LCD. When A0 is low, it selects the command register ($B5F0) of the LCD. When A0 is high, it selects the data register ($B5F1) of the LCD. A LCD is a good output device for appliances, machinery, cars, & alarms plus many others.

4.0 Procedure

The procedure is arranged in a series of steps. Each step is to be completed before moving on to the next step. As each step is built on prior steps, the student will increase their knowledge for other labs or self-study. The student should go though the steps as many times as necessary to master the subject. As an aid in keeping track of your location, the check box next to each step should be checked as completed.

4.1 Description

You will be using the LCD_PORT on the HC11E9 EVBU board. The LCD_PORT is located on the bus of the HC11 microcontroller at predefined addresses. The command register is first written with the proper commands to setup of the LCD. The LCD commands normally send during setup are home cursor, clear display and cursor On with right shift. Next, a series of ASCII characters is send and displayed on the LCD. Then the cursor is moved on the LCD and more characters are displayed. Reference section 4.4 for LCD commands. Reference section 4.5 for all list of ASCII characters.

LCD Commands example

$01 Clear all display & return cursor to home
$02 Cursor to Home
$06 Cursor shift right
$0F Display on, Cursor visible, blinking cursor
$14 Shift cursor right
$3C 8 bit data, 2 lines, 5x11 dots

4.2 Detailed Steps

- [ ] Note in the following steps: LCD_CMD refers to address $B5F0
  LCD_DATA refers to address $B5F1
- [ ] Verify power is not applied to EVBU.
- [ ] Install an AXIOM LCD on the LCD_PORT of the EVBU.

- [ ] Disable ModA and ModB jumpers. Enable MEM_EN jumper. This will configure EVBU for expanded memory.
- [ ] Apply power to the EVBU.
- [ ] Write $3C to LCD_CMD. This configures LCD for 8 bit interface.
- [ ] Write $0F to LCD_CMD. Turns display on with a blinking cursor visible.
- [ ] Write $01 to LCD_CMD. The display will clear any characters and return cursor home. The display is now ready for characters.
- [ ] Write the following values to LCD_DATA: $57,$65,$6C,$6C,$20,$44,$6F,$6E,$65. Note: Buffalo will add a space between characters.
- [ ] Verify the message “Well Done” is displayed on the LCD.
- [ ] Write $10 to LCD_CMD. Verify cursor moves left on LCD.
- [ ] Write $18 to LCD_CMD. Verify entire message is shifted left on LCD.
- [ ] Read LCD_CMD. Verify bit 7 is low by the hex value read as being less then $80. This is the Busy bit. In a normal program, a wait routine must be installed between memory writes to the LCD. This is necessary because the LCD has a controller on board that is much slower then the mircrocontroller write cycle. This is normally a set software delay but you can monitor the busy bit as a way to speed up write cycles.
- [ ] Write $01 to LCD_CMD. You will notice display is cleared and cursor returns to home.

4.3 Conclusion

In conclusion, by selecting the correct command, the display can be cleared or the cursor can be moved. An entire message for the operator can be displayed or information about the system status can be displayed. The vast amount of information a LCD can present makes the device popular for instruments and consumer products. The example program LCD1E.S19 displays a message on the LCD.

4.4 LCD COMMANDS

LCD PORT CONTROL

Clear all display data

A0	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	1

Move cursor home

A0	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	1	0

Entry Mode Set

I/D 0 = decrement cursor 1 = increment cursor
SH 0 = disable display shift 1 = enable display shift

A0	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	1	I/D	SH

Display ON/OFF control

D 0 = display OFF 1 = ON
C 0 = cursor OFF 1 = cursor ON
B 0 = cursor blink OFF 1 = cursor blink ON

A0	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

Cursor or Display Shift

S/C = 0 R/L = 0 shift cursor left, AC is decreased
S/C = 0 R/L = 1 shift cursor right, AC is increased
S/C = 1 R/L = 0 shift display left, AC not changed
S/C = 1 R/L = 1 shift display right, AC not changed
Note: AC = address counter

A0	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	1	S/C	R/L	*	*

Function Set

DL = 0 4 bit mode DL = 1 8 bit mode
N = 0 one line N = 1 two lines
F = 0 5x8 dots F = 1 5x11 dots

A0	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	*	*

Read Busy Flag and Address

BF = 0 not busy 1 = busy
AC = Address Counter

A0	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0

4.5 STANDARD LCD CHARACTER CODES

D = DATA Char = Character

D	Char	D	Char	D	Char	D	Char	D	Char	D	Char
$20	Space	$30	0	$40	Time	$50	P	$60	`	$70	p
$21	!	$31	1	$41	A	$51	Q	$61	a	$71	q
$22	“	$32	2	$42	B	$52	R	$62	b	$72	r
$23	#	$33	3	$43	C	$53	S	$63	c	$73	s
$24	$	$34	4	$44	D	$54	T	$64	d	$74	t
$25	%	$35	5	$45	E	$55	U	$65	e	$75	u
$26	&	$36	6	$46	F	$56	V	$66	f	$76	v
$27	‘	$37	7	$47	G	$57	W	$67	g	$77	w
$28	(	$38	8	$48	H	$58	X	$68	h	$78	x
$29	)	$39	9	$49	I	$59	Y	$69	i	$79	y
$2A	*	$3A	:	$4A	J	$5A	Z	$6A	j	$7A	z
$2B	+	$3B	;	$4B	K	$5B	[	$6B	k	$7B	{
$2C	,	$3C	{	$4C	L	$5C	Yen	$6C	l	$7C	|
$2D	-	$3D	=	$4D	M	$5D	]	$6D	m	$7D	}
$2E	.	$3E	}	$4E	N	$5E	^	$6E	n	$7E	>
$2F	/	$3F	?	$4F	O	$5F	_	$6F	o	$7F	<

5.0 LCD PROGRAM

5.1 Program Description

The LCD1E.S19 program is used for displaying a simple message. First the LCD_CMD is used to setup the LCD for operation. Next, each character of the message is send to the LCD using the LCD_DATA command.

5.2 Running LCD Program

- ☐ Load program LCD1E.S19 into EVBU. This program is located at $0100, which is internal memory. The source is show below.
- ☐ Call 0100 on EVBU.
- ☐ The program will first setup the LCD display with cursor On and display cleared.
- ☐ A string subroutine is called next. This subroutine will output a string of characters to the LCD. This basically is the code an embedded project would contain.

5.3 LCD Source

```
*                Example LCD Display
*                File = LCD1E.S19
*                Equates
End             equ  $00
LCD_CMD    equ  $B5F0          * LCD command address
LCD_DATA   equ  $B5F1           *  LCD DATA address
*
* This subroutine will scan keypad
                ORG  $0100
                ldaa    #$3C            * Setup LCD for 8 bit interface
                staa    LCD_CMD
                bsr     Delay
                ldaa    #$0f            * Display On , blinking cursor
                staa    LCD_CMD     *
                bsr     Delay
                ldaa    #$01            * Clear display, return to home
                staa    LCD_CMD
                bsr     Delay
                ldx     #Running        * message to display
                bsr     String          * send message
                rts                     * return


* This subroutine will send a string to LCD
String
                ldaa     0,x            * load char pointed to by X
                beq      StringE        * Quit if End
                staa     LCD_DATA       * send to LCD
                inx
                bsr      Delay          * delay
```

```
                bra     String          * repeat until end of string
        StringE
                rts
        * This subroutine will Delay for the LCD
        Delay
                ldy     #$FFFF          * load a delay value
        DelayLp
                dey                     * decrement count
                bne     DelayLp         * continue delay
                rts                     * return
*
*  Message to display
Running         fcc     'Running'       * message running
                fcb     End
```

6.0 QUIZ

Question One
What address is LCD_CMD? A. $B5F0 C. $B5F1 B. $1008 D. $E000

Question Two
What address is LCD_DATA? A. $1008 C. $E000 B. $B5F1 D. $B5F0

Question Three
Writing the command $01 will? A. Display "Well Done" C. Display OFF B. Adjust Contrast D. Clear Display

Question Four
What does the position of the cursor represent? A. Count Value C. LCD_CMD B. Next Character D. Contrast

Question Five

The LCD accepts what type of characters?

A. Binary C. ASCII
B. Boolean D. Decimal

Question Six

LCD contrast is controlled by?

A. LCD_CMD C. $14
B. Resistor D. +5v

Question Seven

A R/W signal low represents what to the LCD?

A. Read C. Bias
B. Write D. Address

Question Eight

The LCD display can be?

A. Shifted left C. Cleared
B. Shifted right D. Answers A B C

Question Nine

What is the purpose of a delay?

A. Warm Up C. Wait for LCD
B. Wait for operator D. Instruction Time

Question Ten

Normally a LCD cursor is setup for?

A. Shift Left C. Shift Right
B. Shift Down D. Shift Up

Bonus Question

A0 on the LCD_PORT is used for?

A. Select Read/Write
B. Select Command/Data
C. Select Shift
D. Select Home

7.0 TROUBLESHOOTING

The development system is fully tested and operational before shipping. If it fails to function properly, consult the troubleshooting section of your user manual.

8.0 Tips and Suggestions

Following are a number of tips, suggestions and answers to common questions that will solve most problems users have with the development system. You can download the latest software from the Support section of our web page at:

www.axman.com

A. Remove all components and verify operation using the development board manual.