

# SEVEN SEGMENT DISPLAY

LAB EXPERIMENT

---

---

Class

---

Instructor / Professor



2813 Industrial Ln. • Garland, TX 75041 • (972) 926-9303 FAX (972) 926-6063  
support@axman.com Rev 1.02 • web: <http://www.axman.com>

---

# CONTENTS

1 GETTING STARTED .....	2
1.1 INTRODUCTION .....	2
1.2 SOFTWARE .....	2
1.3 SUPPORT SOFTWARE .....	2
1.4 HARDWARE .....	3
2.0 Visual .....	3
3.0 Theory .....	3
4.0 Procedure .....	3
4.1 DESCRIPTION .....	4
4.2 DETAILED STEPS .....	4
4.3 CONCLUSION .....	5
5.0 Seven Segment Program .....	5
5.1 PROGRAM DESCRIPTION .....	5
5.2 RUNNING SEVEN SEGMENT PROGRAM ...	5
5.3 SEVEN SEGMENT SOURCE .....	5
6.0 Quiz .....	6
7.0 TROUBLESHOOTING .....	7
8.0 Tips and Suggestions .....	8

## 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 boards CME11E9 are supported in this experiment. The system comes complete with schematic and 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 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 Development Kit  
Windows based PC  
Seven Segment Lab Kit

## 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 an 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 through the steps of using the development board for its intended purpose, controlling devices. A seven segment display is one device that is controllable by a microcontroller. In this lab, one display is used for the experiment. A seven segment display is a group of seven LEDs arranged in segments. These segments display numbers 0 thru 9. The port on the microcontroller will output a group of 4 bits. These four bits are in the form of BCD. They are applied to the seven segment driver. Depending on the number, the driver will light up the proper segments for displaying the number. All the LEDs in the display are common anode. By connecting the common anode to +5 and using a current limiting resistor on the cathode side, the driver is able to turn each segment on. Segment intensity is dependent on the current flow and should not exceed the limit of the segment. Displays are good indicators for appliances, machinery, cars, clocks & alarms plus many others. They come in several colors such as red and green.

## 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 students will increase their knowledge for other labs or self-study. The student should go through the steps as many times as necessary to master the subject. As an aid in keeping track of location, the check box next to each step should be checked as completed.

### 4.1 Description

You will be using PORT C on the HC11E9 microcontroller. In single chip mode, PORT C is an input/output port. Bits 0,1,2,3 are used as the 4 bit BCD number. This port is located at address \$1003 on the HC11. Writing directly to this port will apply a number to the seven segment driver. Writing \$00 will display a "0" and writing \$09 will display a "9". Before using PORTC, the data direction register must be initialized. Writing \$0F to address \$1007 (DDRC) sets bits 0,1,2,3 as outputs. First steps guide you through the process of setting up port for use. The next steps show how to display different digits.

### 4.2 Detailed Steps

Note in the following steps: PORTC refers to address \$1003  
DDRC refers to address \$1007

- ☐ Verify power is not applied to EVBU.
- ☐ Install seven segment display, driver IC and resistor array on the breadboard area per drawing.

- ☐ Install ModA jumper. Disable ModB jumper. Disable MEM\_EN jumper. This will configure EVBU for single chip operation.
- ☐ Apply power to the EVBU.
- ☐ Write \$0F to DDRC. This forces bits 0,1,2,3 of port C as outputs.
- ☐ Write \$00 to PORTC. verify display is a “0”. This forces all outputs low, thus representing a zero in BCD.
- ☐ Write \$01 to PORTC, verify display is a “1”. This forces PORTC bit 0,1,2,3 to a \$01, thus representing a one in BCD.
- ☐ Write \$02 to PORTC, verify display is a “2”. This forces PORTC bit 0,1,,3,4 to a \$02, thus representing a two in BCD.
- ☐ Write the following hex values in order: \$03,\$04,\$05,\$06,\$07,\$08,\$09 to PORTC, verify each number is displayed “3”, “4”, “5”, “6”, “7”, “8”, “9”. This forces PORTC bit 0,1,2,3 into representing the BCD value of numbers we are displaying.
- ☐ Write \$0F to PORTC, verify display is blank. This forces PORTC bit 0,1,,3,4 to a \$0F, thus representing a invalid decimal number and no number is displayed.

## 4.3 Conclusion

These simple steps showed how easy a display can be added on a microcontroller. Just one digit was used here. But other ways are available for adding multi digits. A four digit bank can be used by multiplexing the digits or using one of the many chips available on the market. These chips have all of the necessary parts inside for driving several digits or even LEDs. A clock would require at least four digits and six digits with seconds. A temp monitor would require three digits on those hot days of 113 F.

## 5.0 Seven Segment Program

### 5.1 Program Description

First the program must setup port C as outputs. The program first clears register A. Register A is written to PORTC. This displays the digit 0. Digit 0 will remain on the

display for the length of the delay subroutine. Next register A is increment by one. The new value of A is checked if it is greater then 9. If so, then the program starts over at the beginning. Otherwise the next digit is written to PORTC. Digits 0 thru 9 are displayed, then the program starts over at the beginning.

## 5.2 Running Seven Segment Program

- ☐ Load program SevenSeg.s19 into Evbu. This program is located at \$0100, which is internal memory. The source is show below.
- ☐ Without pressing any keys on keypad, call 0100 <enter> on Evbu terminal.
- ☐ The seven segment display will cycle thru digits 0 to 9 then repeat forever.
- ☐ Press and hold key "1" on keypad. Enter call 0100 <enter> on Evbu terminal. The register dump shows register A is equal to \$01 and register B contains \$04.

## 5.3 Seven Segment Source

```
*           Example Seven Segment
*           File = SevenSeg
*           Register Equates
PORTC      equ $1003      * port C output register
DDRC       equ $1007      * port C direction register
*
```

This subroutine counts up on the seven segment display

```
*           org    $0100
*
*           ldaa   #$0F      * all outputs
*           staa   DDRC      * on port C
Loop1
*           clra                * clear register A
Loop2
*           staa   PORTC      * write digit to port C
*           bsr    Delay      * delay for visual
*           inca                * increment A
*           cmpa   #$0A      * if greater then 9
*           beq    Loop1      * restart with digit 0
*           bra    Loop2      * display next digit
```

\* Delay

Delay:

ldy #\$FFFF

DelayA:

dey

bne DelayA

rts \* return

## 6.0 Quiz

### Question One

What is the largest digit that can be displayed?

- |      |       |
|------|-------|
| A. 5 | C. 10 |
| B. F | D. 9  |

### Question Two

How many bits are it required for one digit?

- |       |      |
|-------|------|
| A. 8  | C. 4 |
| B. 16 | D. 9 |

### Question Three

How many segments in the display?

- |      |       |
|------|-------|
| A. 5 | C. 10 |
| B. 7 | D. 9  |

### Question Four

Which digit has the most segments lit?

- |      |      |
|------|------|
| A. 0 | C. B |
| B. 5 | D. 8 |

#### Question Five

What controls the segments of the display?

- A. Digit driver
- B. HC11
- C. Segment driver
- D. PORTB

#### Question Six

All the LEDS in the display are?

- A. Common drive
- B. Common Cathode
- C. Forward
- D. Common Anode

#### Bonus Question

What controls the intensity of the segments?

- A. Resistors
- B. Digit
- C. Segment driver
- D. Microcontroller

## 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 CME11E9 development system. You can download the latest software from the Support section of our web page at:

[www.axman.com](http://www.axman.com)





