SEVEN SEGMENT DISPLAY

PROJECT BOARD & CML12S LAB EXPERIMENT

Class

Instructor / Professor

xiom
anufacturing
™ © 1999

2813 Industrial Ln. • Garland, TX 75041 • (972) 926-9303 FAX (972) 926-6063
support@axman.com Rev 1.01 • 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 boards CML12S and PROJECT BOARD 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 CML12S Development Kit
Windows based PC
PROJECT BOARD

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 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 student's 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 location, the check box next to each step should be checked as completed.

4.1 Description

You will be using PORT T on the DP256 microcontroller. In single chip mode, PORT T is an Bi_Directional port. Bits 0,1,2,3 are buffered though a 573 on the PROJECT BOARD before being applied to the segment driver four bit BCD input. This port is located at address $0240 on the HC12. 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 PORT T, the data direction register must be initialized. Writing $0F to address $0242 (DDRT) sets bits 0,1,2,3 as outputs. First steps guide you though the process of setting up port T for use. Next steps shows how to display different digits. Also included is the steps for displaying the decimal point on the display.

4.2 Detailed Steps

Note in the following steps: PTT refers to address $0240
DDRT refers to address $0242
PTM refers to address $0250
DDRM refers to address $0252

□ Verify power is not applied to CML12S and PROJECT BOARD.

□ Move jumper JP7 on PROJECT BOARD to position 2 & 3. Install ENABLE jumper.

□ Apply power to the CML12S and PROJECT BOARD. See PROJECT BOARD manual.

□ Write $0F to DDRT. This forces bits 0,1,2,3 of port C as outputs.

□ Write $00 to PTT. verify display is a "0". This forces all outputs low, thus representing a zero in BCD.

□ Write $01 to PTT, verify display is a "1". This forces PORT T bit 0,1,2,3 to a $01, thus representing a one in BCD.

□ Write $02 to PTT, verify display is a “2”. This forces PORT T 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 PTT, verify each number is displayed “3”, “4”, “5”, “6”, “7”, “8”, “9”. This forces PORT T bit 0,1,2,3 into representing the BCD value of numbers we are displaying.

□ Write $0F to PTT, verify display is blank. This forces PORT T bit 0,1,,3,4 to a $0F, thus representing a invalid decimal number and no number is displayed.

□ Write $04 to DDRM. This forces Port M bit 2 as output for the decimal point.

□ Write $00 to PTM. This forces Port M bit 2 low and turns decimal point off.

□ Write $04 to PTM. This forces Port M bit 2 high and turns decimal point on.

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 T bits 0,1,2,3 as outputs. The program first clears register A. Register A is written to PORT T. This displays the digit 0. Digit 0 will remain on the display for the length of the delay subroutine. Next register A is incremented 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 PORT T. Digits 0 thru 9 are displayed first , then program starts over at the beginning.

5.2 Running Seven Segment Program

□ Load program 7-Seg2D.s19 into CML12S. This program is located at $1000, which is internal memory. The source is show below.

□ Program is executed by “CALL 1000” <enter> on CML12S terminal.

□ The seven-segment display will cycle digits 0 thru 9 then repeat forever.

5.3 Seven Segment Source

```
;        Example Seven Segment
;        File = 7-Seg2D
;        HC12 code
;        Register Equates
PTT:         equ     $0240        ; port T output register
DDRT:        equ     $0242        ; port T direction register
PTM:         equ     $0250        ; port M output register
DDRM         equ     $0252        ; port direction register
;
; This subroutine counts up on the seven segment display
             org     $1000        ; program starts here
             movb    #$0F,DDRT    ; all outputs
             movb    #$04,DDRM    ; D.P. as output

Loop1
             clra                 ; clear register A
Loop2
             staa    PTT          ; write digit to port T
             bsr     Toggle       ; toggle Decimal Point
             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:
             ldab    #$10
DelayB
             ldy     #$FFFF
DelayA:
             dey
             bne     DelayA
             decb
             bne     DelayB
             rts
;
;
Toggle:
             ldab    PTM         ; get decimal point value
             andb    #$04        ; mask bit
             comb                ; invert B
             andb    #$04        ; only one bit high
             stab    PTM         ; set Decimal Point
             rts


;
```

;

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 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 C. Segment driver

B. HC12 D. PORT T

Question Six

All the LEDS in the display are?

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

Bonus Question

What controls the intensity of the segments?

A. Resistors C. Segment driver
B. Digit 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 development system. You can download the latest software from the Support section of our web page at:

www.axman.com