

## Workshop 7: Data Sorting and LCD Output

**Purpose:** Using the embedded controller to input and sort ten binary integers, then display the lowest, highest and average on an LCD.

**Objective :** This workshop is designed to familiarize the student with elements of the embedded microcontroller.

**Real world application:** LCD output

**Requirements:**

Hardware: Scott Edwards Electronics LCD, DIP switch

Software commands: Call clr\_pa4, Call tst\_pa4, Serout ?, N2400, (#b?)

**Procedure:** Create a program that will take read ten integer into eeprom , then sort them, then display the sorted data on an LCD. Use a DIP switch to create a binary input on PB0-6. To set the input to an integer value of 17, PB0 and PB3 would have logic 1 input, all others would have a logic 0. To input a value 113, then PB0, 4,5,6 would be high and all others would be low. Store the ten integers into eeprom starting at address 1. Use PA4 signal when to take an input. When PA4 is high the integer input can be changed, cycling PA4 to logic 0 shall signal the program store the input, cycling the input back to logic 1 will allow the an integer input to be changed again.

After ten inputs are received the program will sort the data in ascending order and then display the data on the LCD as follows:

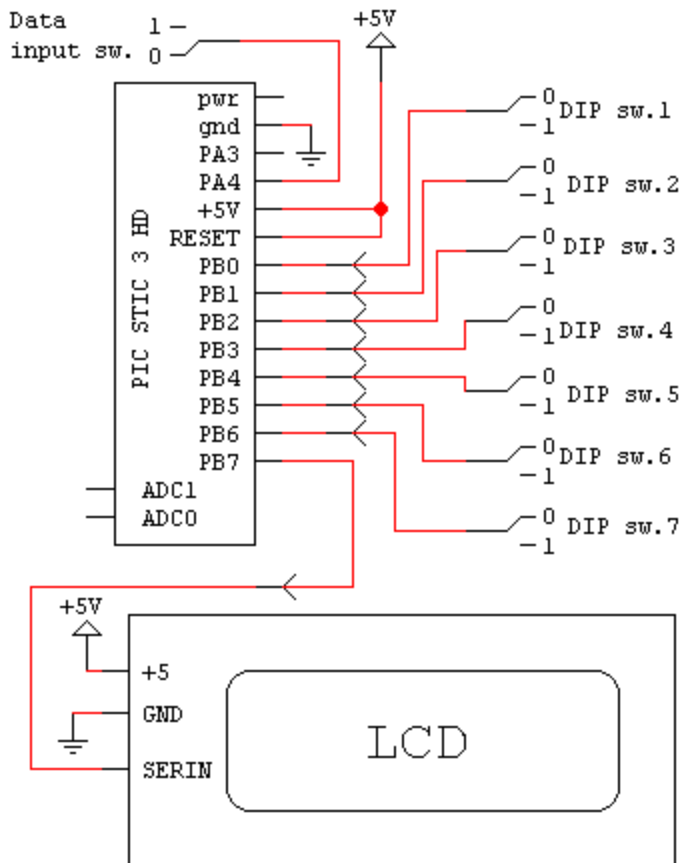
LCD line1:-> LOW#<lowest integer> AVG#<average integer>

LCD line2:-> HIGH#<highest integer>

Hint: Use inverted RS-232 input to the LCD (output from Picstic)

Example: serout 7,N2400,(#b1)

### *Circuit Drawing for Workshop #7*



[illegible]

'Herb Wagner

[illegible]

```
dirs = %10000000 'set PB0-6 as inputs
```

////////////////////////////////////

```
b2 = b2 + 1 'incrm. input counter
```

call `tst_pa4` 'take input from pa4'

goto set

////////////////////////////////////

```
out: serout 7,N2400,(254,1) 'clear LCD
```

```
write b1,b4 'store it into eeprom
```

pause 500 'pause 1/2 sec

call `tst_pa4` 'take new input from pa4'

goto out 'repeat procedure

////////////////////////////////////

'address 1, highest in address 10

```
for b3 = 1 to 9
```

$$b6 = b3 + 1$$

```
if b8 < b4 then swop
```

next b7

swop: write b3,b8

```

goto nex

```

'//////////

'This code will compute the avg value of the 10 nu

for b2 = 1 to 10

```

for b2 = 1
read b2,b7

```

$$w9 = b7 + w9$$

```

w2 = b1
next b2

```

b7 = w9 / 10

```

'////////////////////////////////////
'The following lines will display the lowest value, the highest value
'and the average value on the LCD
display: serout 7,N2400,(254,1)
b3 = 1
read b3,b4
serout 7,N2400,("Low#:")
serout 7,N2400,(#b4)
serout 7,N2400,(" High#:")
b3 = 10
read b3,b4
serout 7,N2400,(254,192) 'carriage return
serout 7,N2400,(#b4)
serout 7,N2400,(" Avg#:")
serout 7,N2400,(#b7)
end
'////////////////////////////////////
'Assembly subroutines to clear pa4 and pole pa4
asm
_clr_pa4 bcf porta,4
bsf status,5
bcf trisa,4
bcf status,5
goto done
_tst_pa4 bsf status,5
bsf trisa,4
bcf status,5
bcf _b0,1
btfsc porta,4
bsf _b0,1
goto done
endasm

```