

Madison Area Technical College  
Microcontroller  
Laboratory Activity

Name: \_\_\_\_\_

**Laboratory Activity - USART - Serial Communications Module**

**Description:**

The purpose of this laboratory activity is to transmit the ratio in % of the input to rail voltage level from one Microcontroller to another via a USART serial communications channel. The ratio to be transmitted can be calculated as

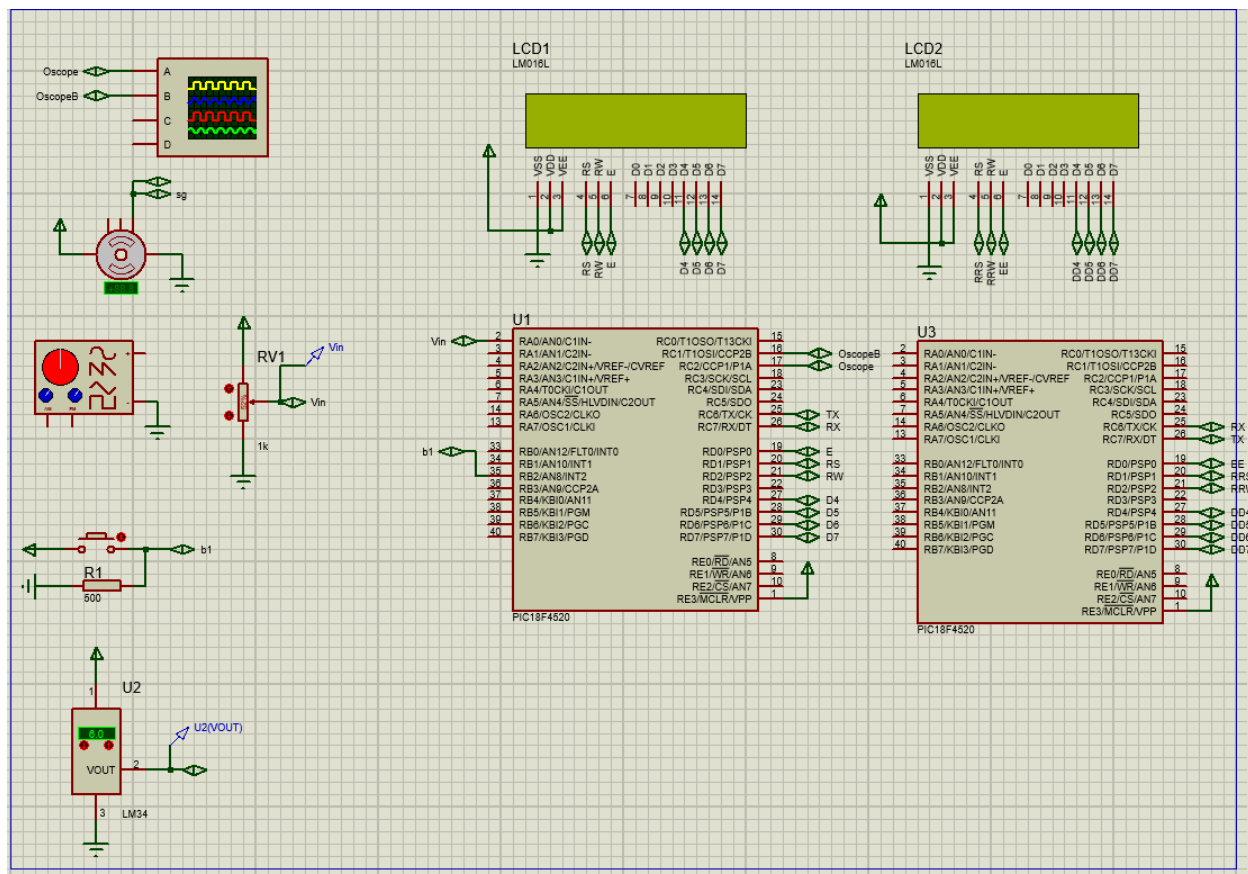
$$Ratio = V_{in}/V_{cc}$$

For example, if the input voltage is 2.5V and the rail voltage is 5V the ratio will be

$$Ratio = V_{in}/V_{cc} = 2.5/5.0 = 0.5$$

or 50%. The main Microcontroller must detect the input voltage through an interrupt driven A/D converter unit connected to PIN\_A0 and it must display the input voltage in Volts on its LCD panel.

The secondary Microcontroller should receive the data from the main Microcontroller and display the % Ratio on the first line of its LCD panel and the input voltage in Volts on the second line of the LCD panel.



Tx:

```
1  #include <18f4520.h>
2  #use delay (clock = 2000000)
3  #fuses HS, NOWDT, NOLVP
4  #include "../Library/myLibrary.h"
5  #include "../Library/modifiedlcd.h"
6
7  float Vin = 0;
8  int16 vin_send = 0;
9
10 #INT_RDA
11 void int_rda_isr(){
12     printf(lcd_putc, "\f Vin: %f", Vin);
13 }
14
15 #INT_AD
16 void int_ad_isr() {
17     Vin = *Q * (5.0 / 1023.0 );
18     vin_send = (Vin * 10.0);
19     output_toggle( PIN_C0 );
20 }
21
22 main(){
23     // Analog setup
24     *TRISA = 0x01;
25     ADCON1->PCFGx = 0xE;
26     ADCON0->ADON = 1;
27     ADCON0->CHSx = 0;    // Channel 0
28     ADCON1->VCFG0 = 0;
29     ADCON1->VCFG1 = 0;
30     ADCON2->ADFM = 1;    // Right Justified
31     ADCON2->ACQTx = 5;
32     ADCON2->ADCSx = 5;
33     PIE1 -> ADIE = 1;
34
35     *TRISC = 0x80;        // C7 input
36
37     TXSTA->TXEN=1;        // Transmit Enable
38     TXSTA->SYNC=0;        // Asynch Mode
39
40     RCSTA->SPEN=1;        // Serial Enable
41     RCSTA->CREN=1;        // Receiver Enable
42
43     TXSTA->BRGH = 0;
44     BAUDCON->BRG16 = 0;
45     *SPBRG = 31;         // Baud Rate 9.6KHz
46
47     PIE1->RCIE=1;        // Receiver Interrupt Enable INT_RDA
48     INTCON->PEIE=1;
49     INTCON->GIE=1;
50
51     lcd_init();          // LCD Panel Initialize
52 }
```

```

53 //printf(lcd_putc, "\f Hello");
54 while(1){
55     ADCON0->GODONE=1;    // Trigger
56     delay_ms( 500 );
57     printf(lcd_putc, "\f Vin=%f V", *Q*(5.0/1023.0));
58     *TXREG = vin_send;    // Transmitting 'Vin'
59 }
60 }
61
62

```

Rx:

```

1  #include <18f4520.h>
2  #use delay (clock = 2000000)
3  #fuses HS, NOWDT, NOLVP
4  #include "../Library/myLibrary.h"
5  #include "../Library/modifiedlcd.h"
6
7  float ratioed = 0;
8
9  #INT_RDA
10 void int_rda_isr() {
11     ratioed = (*RCREG / 10.0);
12     printf(lcd_putc, "\f %f %%", (ratioed / 5.0) * 100.0 );
13     printf(lcd_putc, "\n Vin= %f", ratioed);
14 }
15
16 main(){
17
18     *TRISC = 0x80;    // C7 input
19
20     TXSTA->TXEN=1;    // Transmit Enable
21     TXSTA->SYNC=0;    // Asynch Mode
22
23     RCSTA->SPEN=1;    // Serial Enable
24     RCSTA->CREN=1;    // Receiver Enable
25
26     TXSTA->BRGH = 0;
27     BAUDCON->BRG16 = 0;
28     *SPBRG = 31;
29
30     PIE1->RCIE=1;    // Receiver Interrupt Enable
31     INTCON->PEIE=1;
32     INTCON->GIE=1;
33
34     lcd_init();    // LCD Panel Initialize
35     while(1){
36     }
37 }
38
39

```