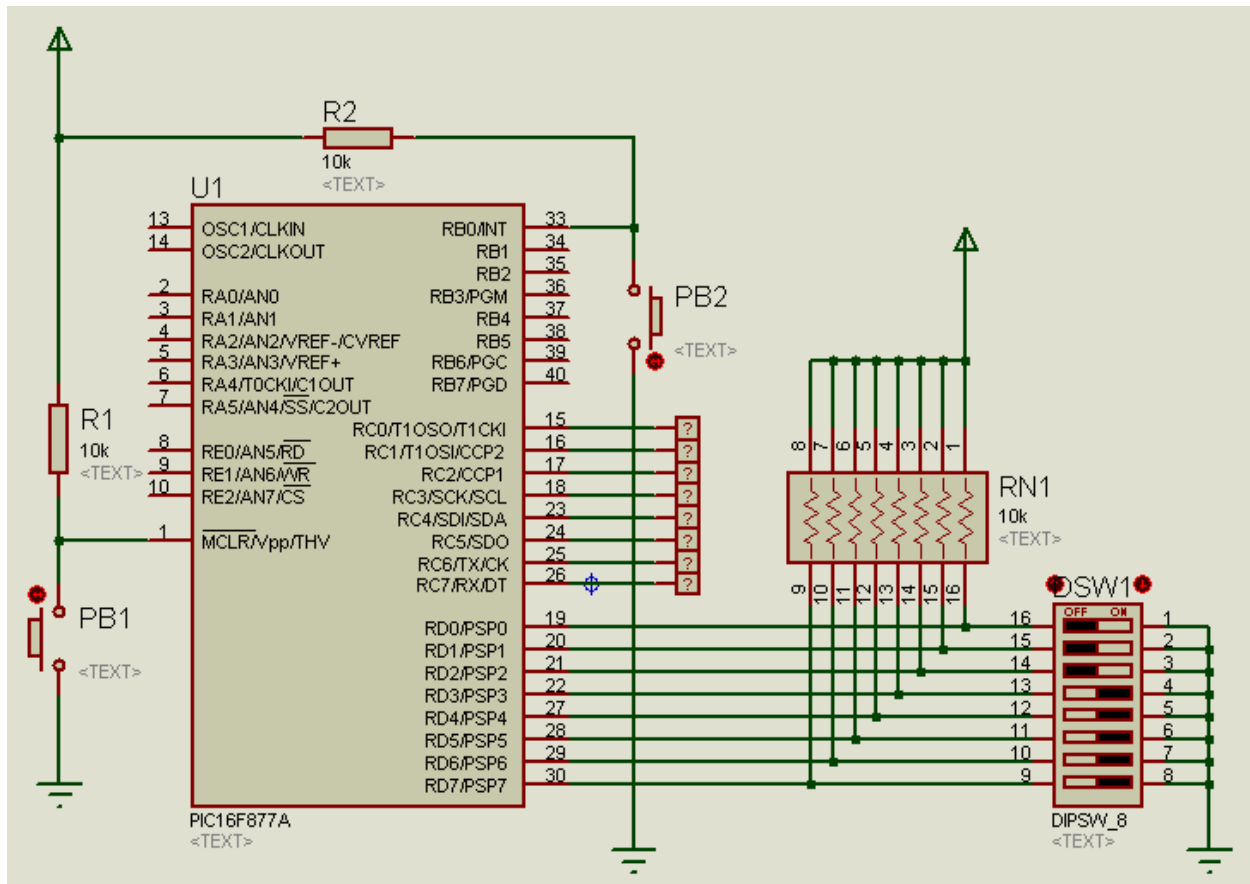


In Campus Activity 3
ECPE402
Nov 18, 2021 | 0100P – 0330P

Construct the circuit below in Proteus.



Note:

- The oscillator to be used is 8 MHz
- PIC16F877A has 256 x 8 bytes of EEPROM data memory whose memory addresses are from 00h until FFh

Create a project named my_adc in MikroC and encode the program below

```
const unsigned short eeprom_address = 0x10;

void interrupt()
{
    // check if RB0/INT requested an interrupt
    if (INTCON.INTF == 1)
    {
        EEPROM_Write(eeprom_address, PORTD);
        INTCON.INTF = 0;          // disable INTF flag
    }
}

void init_interrupt()
{
    INTCON.INTF = 0;              // disable INTF flag
    INTCON.INTE = 1;              // locally enable INTE interrupt through
    RB0/INT pin
    INTCON.GIE = 1;              // globally enable INTE
}

void init_ports()
{
    TRISC = 0x00;                 // PORTC is configured as an output port
    TRISD = 0xFF;                 // PORTD is configured as an input port
    TRISB.B0 = 1;                 // RB0 is configured as an input port
}

void main()
{
    init_ports();
    init_interrupt();

    PORTC = EEPROM_Read(eeprom_address);

    while (1);
}
```

Make sure to check the EEPROM library of MikroC in order to use the functions EEPROM_Read() and EEPROM_Write()

Configure the project by clicking Project -> Edit Project and follow the configuration below

The screenshot shows the MikroC IDE configuration window. On the left, there is a vertical list of settings, each with a dropdown menu:

- Oscillator Selection**: XT oscillator
- Watchdog Timer**: Disabled
- Power-up Timer**: Disabled
- Brown-out Reset**: Enabled
- Low-Voltage (Single-Supply) In-Circuit Serial Programming**: Disabled
- Data EEPROM Memory Code Protection**: Disabled
- Flash Program Memory Write**: Disabled
- In-Circuit Debugger Mode**: Disabled
- Flash Program Memory Code Protection**: Disabled

On the right, there is a section titled "MCU and Oscillator" with the following settings:

- MCU Name**: P16F877A
- MCU Clock Frequency [MHz]**: 8.000000

Below this, there is a "Build Type" section with two radio buttons: "Release" (selected) and "ICD Debug". To the right of this is a "Heap" section with a "Size" input field set to 0.

At the bottom right, there is a "Configuration Registers" section with a text area showing the configuration: "CONFIG : \$2007 : 0x2F49". Below this text area is a button labeled "General Output Settings ...".

On the far right, there are three buttons: "Load Scheme", "Save Scheme", and "Default".

Compile the program.

Simulate the circuit in Proteus.

Change the value of the DIP switch and press PB2 to store the value of the DIP switch to the EEPROM. Reset the microcontroller and observe the output at Port C. Store other values to the EEPROM using the DIP switch and PB2.