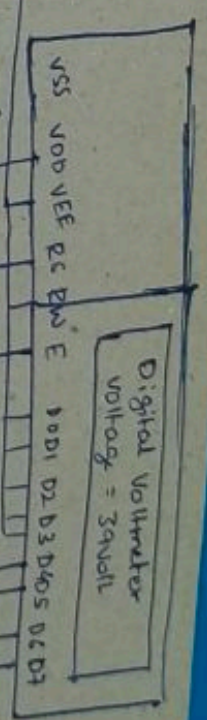
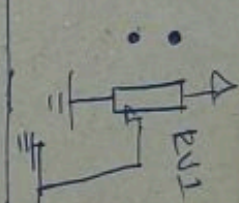
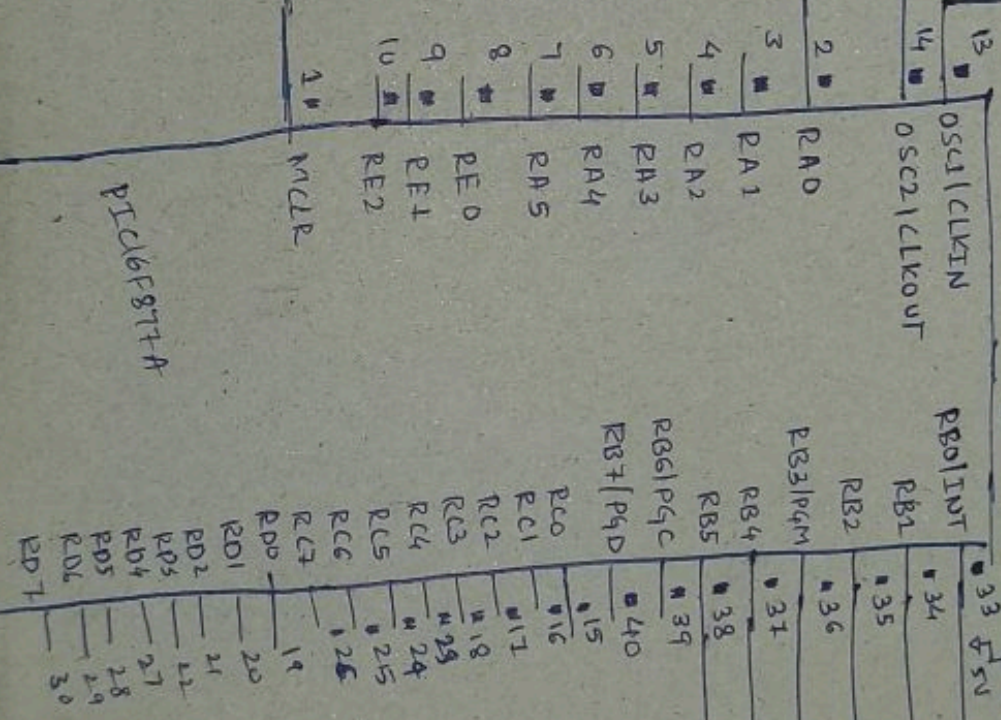


PIC16F877A



Voltage divider formula.

To measure 0 to 100V using a microcontroller using PIC, we can use a voltage divided circuit to step down the voltage to a range that the microcontroller's ADC can handle.

Voltage divider uses two resistors to scale down the input the formula is
$$V_{out} = V_{in} \times \frac{R_2}{R_1 + R_2}$$

to scale down 100V to 3.3V

$$3.3V = 100V \times \frac{R_2}{R_1 + R_2}$$

Let's choose R_2 as 3.3V.

$$\frac{3.3 \Omega}{R_1 + 3.3 \Omega} = 0.33$$

$$R_1 = 96.7 K\Omega$$

$$R_1 = 100 K\Omega \text{ and } R_2 = 3.3 K\Omega$$

Use a Zener diode Across the ADC input and Ground to protect Against over voltage.

Add a small capacitor Across the ADC input to filter out Noise.

Here's the code for measuring the voltage using
XC8 compiler.

```
#include <xc.h>
```

```
#define XTAL_FREQ 20000000 // Define oscillator  
// freq for delay
```

```
void ADC_init()
```

```
{
```

```
    ADCON0 = 0x41;
```

```
    ADCON1 = 0x80;
```

```
}
```

```
unsigned int ADC_read (unsigned char channel)
```

```
{
```

```
    if (channel > 7) return 0;
```

```
    ADCON0 &= 0xC5;
```

```
    ADCON0 |= channel << 3;
```

```
    _delay_ms(2);
```

```
    GO_DONE = 1;
```

```
    while (GO_DONE);
```

```
    return ((ADRES1 << 8) + ADRESL);
```

```
}
```

```
void main()
```

```
{
```

```
    unsigned int adc_value;
```

```
    float voltage, input_voltage;
```



```
TRISA = 0xFF;
```

```
ADC_Init();
```

```
while(1)
```

```
{
```

```
    adc_value = ADC_read(0);
```

```
    voltage = (adc_value * 5.0) / 1023.0;
```

```
    input_voltage = voltage * ((100.0 / 3.3) / (3.3));
```

```
    delay_ms(500);
```

```
}
```

```
}
```

Explain^{alt} of the code

Initialization - 'ADC_Init()': Initialize the ADC module
'ADCON0' configures ADCON0 register to turn ON the ADC and set the conversion clock.
ADCON1 → Configures ADC voltage reference and Result format.

Reading ADC value - ADC_read(): Reads the ADC value from the specified channel. 'ADCON0' configuration ensures correct channel selection.

Voltage calculation - Convert the ADC value to voltage
calculate the Actual input voltage using the voltage divider ratio (input voltage)

Schematic Representation

High voltage (0-100V)

|

R_1 (100k Ω)

|

-----> ADC pin (PIC)

|

R_2 (3.3k Ω)

|

GND.