Capacitance meter

# Abstract:

Most of the cheap multi meters do not come with capacitance measuring capabilities, so the primary focus of this project is to make a capacitance meter using AVR based microcontroller. Once a voltage is applied across the ends of the capacitor it starts to charge itself and the time it takes is dependent on its value, so by monitoring the time taken by the capacitor to charge (using microcontroller) and putting it in mathematical equations, the capacitance value can be calculated.

# Introduction:

Time period of a capacitor is the amount of time it takes to reach to 63.2% of the applied voltage.

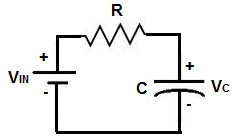


Figure 1

As shown in figure 1, the capacitor shows transient behaviour and takes time to charge.

Where,

*Time period = R \* C*

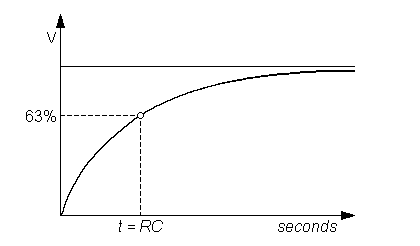


Figure 2

Figure 2 shows the transient behaviour a capacitor while charging.

So we need a timer that automatically shuts itself down as soon as the voltage across the capacitor reaches 63.2% of the applied voltage.

# Components:

### ATmega32

The datasheet suggests, ATmega32 can be operated at a frequency of 16MHz and has 8 channel 10bit ADC additionally it supports UART that can be used for communication.

### HC-05

Bluetooth module that will be used to show the calculated results.

### Resistor

### 1000 ohm (1k) resistor for charging the capacitor

# Circuit diagram:

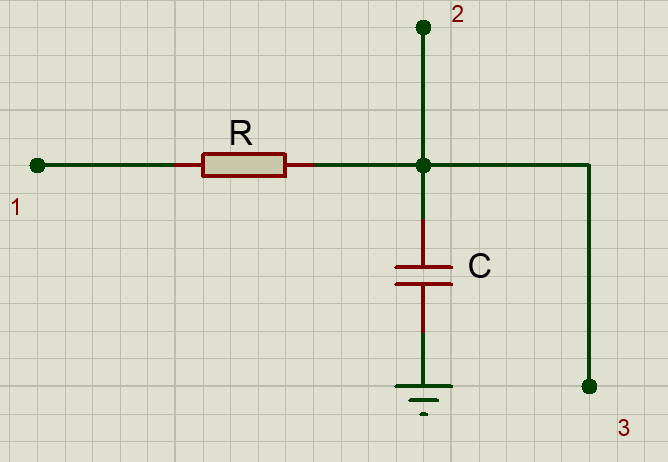


Figure 3

* 1 🡪 PB0 (charging pin)
* 2 🡪 ADC0
* 3 🡪 PB1 (discharging pin)
* R 🡪 1000 ohm (1k)
* C 🡪 Unknown Capacitance

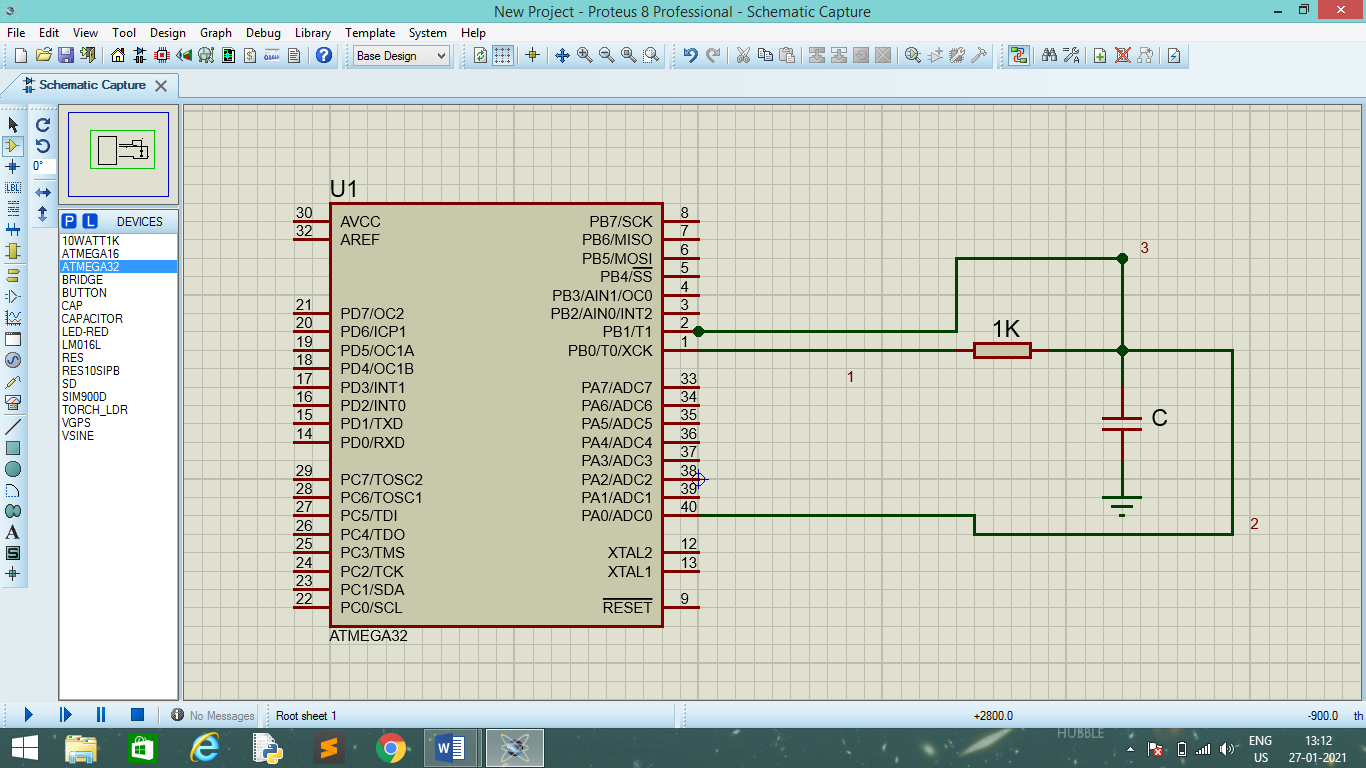


Figure 4

Visit the git hub link for more complete schematic.

# Equations:

### Baud Rate

Where,

* BAUD = Baud Rate
* fosc = Clock frequency
* UBBR = USART Baud Rate Register

### 10 bit ADC

Where,

* Value = ADC value
* Vin = input voltage
* Vref = reference voltage

### Capacitance value

Where*,*

* *=* timeperiod
* *R =* resistance in series with capacitor

# Program flow:

The whole program flow model can be sub categorised into four main blocks for better understanding

### Initialising the UART

HC-05 operates on a baud rate of 9600, so we initialise the UBBR with a value that corresponds to 9600 baud rate. To calculate the value required inside the baud rate consult the equations section, alternatively datasheet can also be referred.

### Initialising the TIMER1

ATmega32 has 3 timers and in this project we will be using TIMER1, which is a 16-bit timer. To initialise the timer we need to set the timer pre scaler. In order to have better precision ATmega32 is made to operate at 16MHz clock, and TIMER1 overflow interrupt is used.

### Initialising the ADC

For better accuracy ADC is made to operate in 10-bit mode, internal Vref = 5V is set for its operation. Just like in TIMER1, ADC conversion complete interrupt is used for better accuracy

# Working:

Initially the PINB0 is set HIGH to charge the capacitor.

The ADC continuously monitors voltage across the capacitor, once the voltage reaches 63.2% of 5V, i.e. ADC register points to 647

## Teacher initials \_\_\_\_\_\_\_\_\_\_\_\_

### You are ready to create your project! Please revisit the project tasks/requirements as you work.

# Summarize what you learned:

Possible student prompts: What did you learn? What worked well? What was the most challenging aspect of this project? What will you do differently next time?

# Add the link to your project here:

Link to access project