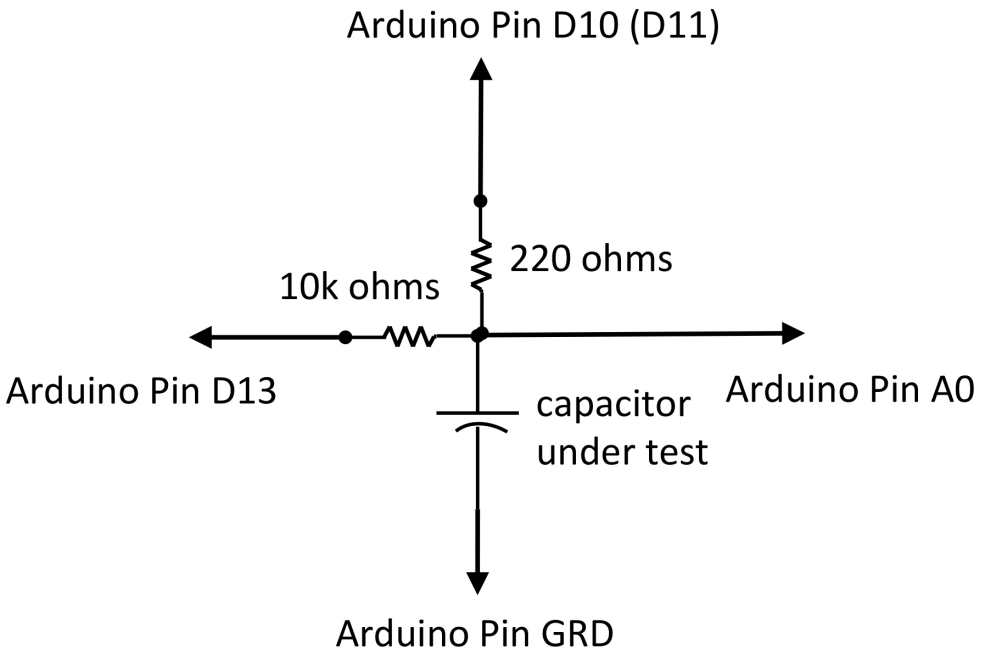
Lab 4 Determining the value of a Capacitor

1. Connect two known resistors (10k and 220) with a capacitor in the following fashion. If you use an electrolytic capacitor that the polarization of voltage applied matches the polarization of the capacitor terminals. ***<u>Make sure you connect the 220 ohm resistor to digital pin D11.</u>***



2. Run the sketch from at the end of this document.
3. The way this sketch works
    a. The Arduino reads the voltage across the capacitor from the Analog pin A0
    b. The Arduino sets Digital Pin D13 to high to charge the capacitor
    c. The Arduino reads the voltage of the capacitor to see when 1 time constant has elapsed.
    d. When 1 time constant has elapsed, knowing that the charging resistor is 10k ohms it calculates the value of the capacitance by dividing the time for 1 time constant.
    e. It then determines if the capacitor it's units are microfarads or nanofarads.
    f. It then discharges the capacitor through digital pin 11.

4. Does the Capacitance reading from the sketch the same as the manufacturers designation.
5. Repeat the experiment with 2 different capacitors

Some background

The basic equation of a charging (discharging) capacitor is:

$$v_c(t) = V - Ve^{-(\frac{t}{RC})}$$

where V is the voltage it charges to, R is the resistance through which it charges, C is its Capacitance and RC is called it's time constant.

It takes an infinite amount of time to truly charge up

$$v_c(t \to \infty)|_{\max} = V - Ve^{-(\frac{t \to \infty}{RC})} = V$$

however after a time equivalent to 1 time constant. it reaches 63% of it's final charge.

$$\frac{v_c(t = RC)}{V} = 1 - e^{-1} = 0.63$$

On the Arduino analog inputs range from

0-1023

Therefore, $0.63 \times 1023 =$

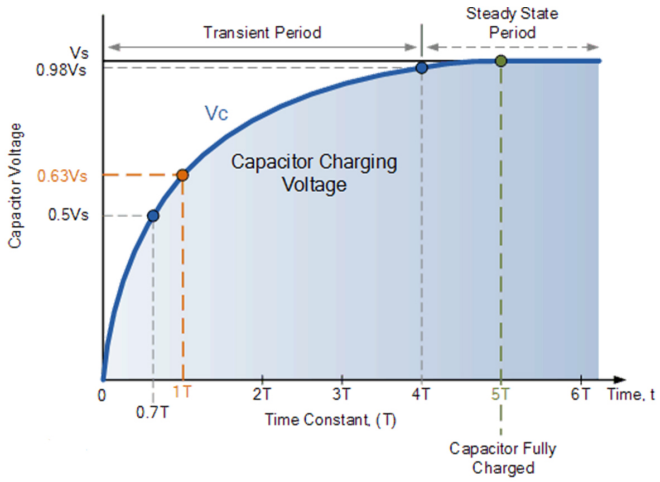
$646.7 = 647$

Elapsed time, $t_{elapsed}$, to reach 648 or 63.2% of 5volts equals the

time constant, RC seconds.

$$C = \frac{t_{elapsed}}{R}$$

That is, when the analog input reaches to 648 one time constant has elapsed.



https://www.electronics-tutorials.ws/rc/rc_1.html

Arduino Sketch

```
/*  RCTiming_capacitance_meter
 *   Paul Badger 2008
 *  Demonstrates use of RC time constants to measure the value of a capacitor
 *
 * Theory   A capcitor will charge, through a resistor, in one time constant, defined as T seconds where
 *    TC = R * C
 *
 *    TC = time constant period in seconds
 *    R = resistance in ohms
 *    C = capacitance in farads (1 microfarad (ufd) = .0000001 farad = 10^-6 farads )
 *
 *    The capacitor's voltage at one time constant is defined as 63.2% of the charging voltage.
 *
 *  Hardware setup:
 *  Test Capacitor between common point and ground (positive side of an electrolytic capacitor  to common)
 *  Test Resistor between chargePin and common point
 *  220 ohm resistor between dischargePin and common point
 *  Wire between common point and analogPin (A/D input)
 */

#define analogPin      0         // analog pin for measuring capacitor voltage
#define chargePin      13        // pin to charge the capacitor - connected to one end of the charging resistor
#define dischargePin   11        // pin to discharge the capacitor
#define resistorValue  10000.0F   // change this to whatever resistor value you are using
                        // F formatter tells compliler it's a floating point value

unsigned long startTime;
unsigned long elapsedTime;
float microFarads;              // floating point variable to preserve precision, make calculations
float nanoFarads;

void setup(){
  pinMode(chargePin, OUTPUT);     // set chargePin to output
  digitalWrite(chargePin, LOW);

  Serial.begin(9600);           // initialize serial transmission for debugging

void loop(){
```

```
 digitalWrite(chargePin, HIGH);  // set chargePin HIGH and capacitor charging
 startTime = millis();

 while(analogRead(analogPin) < 648){      // 647 is 63.2% of 1023, which
corresponds to full-scale voltage
 }

 elapsedTime= millis() - startTime;
// convert milliseconds to seconds ( 10^-3 ) and Farads to microFarads ( 10^6 ),  net
10^3 (1000)
 microFarads = ((float)elapsedTime / resistorValue) * 1000;
 Serial.print(elapsedTime);      // print the value to serial port
 Serial.print(" mS    ");       // print units and carriage return


 if (microFarads > 1){
  Serial.print((long)microFarads);      // print the value to serial port
  Serial.println(" microFarads");       // print units and carriage return
 }
 else
 {
  // if value is smaller than one microFarad, convert to nanoFarads (10^-9 Farad).
  // This is  a workaround because Serial.print will not print floats

  nanoFarads = microFarads * 1000.0;     // multiply by 1000 to convert to
nanoFarads (10^-9 Farads)
  Serial.print((long)nanoFarads);       // print the value to serial port
  Serial.println(" nanoFarads");        // print units and carriage return
 }

 /* dicharge the capacitor  */
 digitalWrite(chargePin, LOW);           // set charge pin to  LOW
 pinMode(dischargePin, OUTPUT);          // set discharge pin to output
 digitalWrite(dischargePin, LOW);        // set discharge pin LOW
 while(analogRead(analogPin) > 0){       // wait until capacitor is completely
discharged
 }

 pinMode(dischargePin, INPUT);          // set discharge pin back to input
}
```