Digital Capacitance Meter
PART NO. 2261010

This project lets you measure capacitors in an alone range of measure from 0.000pF to 1000uF. That is, a 16x2 LCD Display will be displaying a sole scale from 0.000pF to 1000uF whose main components will be an Arduino Uno and a 16X2 LCD Display. Go to: http://www.instructables.com/id/Digital-Capacitance-Meter/

Time Required: 10 hours depending on experience
Experience Level: Intermediate

Required tools and parts:
You will need:
Soldering iron and solder
Wire strippers and cutter
Needle nose pliers
Multimeter
Electrical Drill
Drill bit of 1/16” #22, Wire any color, 1 meter
A 9V Battery

Bill of Materials:

<table>
<thead>
<tr>
<th>Qty</th>
<th>Jameco SKU</th>
<th>Component Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2160374</td>
<td>16x2 Character LCD Display - White on Blue 5V</td>
</tr>
<tr>
<td>1</td>
<td>2151486</td>
<td>Arduino Uno R3 DIP Edition (Revision 3)</td>
</tr>
<tr>
<td>1</td>
<td>2152438</td>
<td>Arduino Uno Proto Shield (PCB only)</td>
</tr>
<tr>
<td>1</td>
<td>2207056</td>
<td>9V Battery Snap with 2.1mm Barrel Plug</td>
</tr>
<tr>
<td>1</td>
<td>222608</td>
<td>Cable USB2.0 A/B 3 Feet Black USB-A Male To USB-B Male</td>
</tr>
<tr>
<td>1</td>
<td>160882</td>
<td>Connector Unshrouded Header 40 Position 2.54mm Straight Thru-Hole</td>
</tr>
<tr>
<td>1</td>
<td>182837</td>
<td>Potentiometer 1/4” Square Cermet 1/2W 10Kohm</td>
</tr>
<tr>
<td>1</td>
<td>2134926</td>
<td>ABS Plastic Enclosure for Arduino Board - Fits UNO or MEGA</td>
</tr>
<tr>
<td>1</td>
<td>2144614</td>
<td>6 Position Female Header - Pass through Style for Arduino</td>
</tr>
</tbody>
</table>

Step 1 - List of Materials
1. 16X2 PARALLEL LCD DISPLAY
1. Arduino Uno R3 DIP Edition (Revision 3)
1. Arduino Uno Proto Shield (PCB only)
1. 9V Battery Snap with 2.1mm Barrel Plug
1. Cable USB2.0 A/B 3 Feet Black USB-A Male to USB-B Male
1. Connector Unshrouded Header 40 Position 2.54mm Straight Thru-Hole
1. Potentiometer 1/4” Square Cermet 1/2W 10Kohm
1. ABS Plastic Enclosure for Arduino Board - Fits UNO or MEGA
1. 6 Position Female Header - Pass through Style for Arduino
Step 2 - Schematic Diagram

In this step, you are going to concentrate very well in what you will be constructing. That is, this step of your project is crucial since you will need to understand how to connect each component so that the whole project functions correctly. Therefore, this step becomes the main step or an imperative measure so that your project be completed successfully. A larger version will be at the end of the instructions.

Step 3 - 16X2 LCD Display

In this step, you can cut 2x6 pins and put them into the holes of the 16x2 LCD Display in the corresponding pins: 1 to 6 & 11 to 16 so that you can have more space for working between the PCB and the Display when this last one be installed.
Step 4 - 16X2 LCD Display 2

In this step, you are going to do the main connections to the display those that you will connect to your Arduino Uno later. Then, you should identify the connections from 16x2 LCD Display in the pins: 4, 6, 11, 12, 13, and 14 that will be connected respectively to the Arduino Uno in the pins: 11, 9, 5, 4, 3, and 2 without forgetting the connections to +5V, GND, and Pot of 10K.

Step 5 - 16X2 LCD Display 3

In this step, you are going to match the connections done previously to your LCD Display with the future connections that you will do in the PCB: observe the photo where you can see the details closer.

Step 6 - PCB

Once you know how to do the connections between your 16X2 LCD Display and the PCB, you should separate them so that you can install on the PCB: the Connector Unshrouded Headers by utilizing 2X8 pins in the side of the digital pins while using 2 pins in the other side for connecting to GND and +5V.
Step 7 - PCB 2

In this step, you are going to connect the GND’s so that you can have all GND’s connected and so you can install the pot of 10K directly on the GND track and then you can connect it to +5V track as well while completing the connections to this component soldering the central pin to other close track.

Step 8 - PCB 3

Now, you can do all of connections: that is, preparing the connections and leaving enough space so that can later install the 16x2 LCD Display

Step 9 - Joining the PCB & the Display

It’s time of matching each connection between the PCB and your LCD display so that can later solder correctly each element installed.
Step 10 - Joining the PCB & the Display 2

Check carefully the connections in the back side of your PCB so that you can observe if everything is OK with those connections that you did between the PCB and the LCD Display. Obviously, you are going to check the connections done to GND and +5V on the right track as well.

Step 11 - Completing the Project

Once completed the project, leave clear which the outputs are. That is, it's imperative to define the outputs for this project: in this case, they are A0 for the negative (-) and A4 for the positive (+).

Step 12 - Getting Ready For Doing Precision Measures

Before uploading the code, you should eliminate the metallic parts of the PCB's holes of A0 and A4 using a drill bit of 1/16" in order to keeping '0' (zero) capacitance when the 1x6 Position Female Header - Pass through Style for Arduino Uno be installed.
Step 13 - Arduino Enclosure

Before installing the Arduino Uno in its enclosure, cut the plastic posts like it is showed in the photo.

Step 14 - Mounting the Shield on the Arduino Uno

Once installed the Arduino Uno inside the enclosure, you can mount the shield on the Arduino Uno.

Step 15 - Inserting the 1x6 Position Female Header - Pass through Style for Arduino

In this step, you can insert the 1x6 Position Female Header - Pass through Style for Arduino Uno and so when you upload the code then the cursor will display 0.000pF.
Step 16 - Uploading the Code

Plugging the USB-A to USB-B cable between your project and the computer, upload the code at: http://pastebin.com/njKZrfv
Next, observe the cursor where you will see 0.000pF.

Step 17 - Using Your Project

Once uploaded the code from http://pastebin.com/njKZrfv, unplug the USB-A to USB-B cable that is plugged between the computer and your project so that you can plug your 9V Battery Snap with 2.1mm Barrel Plug and so can also use your 9V battery to get the measure of each capacitor what you want to measure. In this case, I am measuring a 1 pF capacitor. Note that before measuring the capacitor, you can observe in the cursor: 0.000 pF.

Step 18 - Using Your Project 2

Now measuring a capacitor of 3.3pF.
Step 19 - Using Your Project 3
Measuring a capacitor of 10pF.

Step 20 - Using Your Project 4
Measuring a capacitor of 10nF.
//Digital Capacitance Meter
//Measuring from 0.000pF to 1000uF

#include <LiquidCrystal.h>

LiquidCrystal lcd(11, 9, 5, 4, 3, 2);
const int OUT_PIN = A4;
const int IN_PIN = A0;
const float IN_STRAY_CAP_TO_GND = 24.48;
const float IN_CAP_TO_GND = IN_STRAY_CAP_TO_GND;
const float R_PULLUP = 34.8;
const int MAX_ADC_VALUE = 1023;

void setup()
{
  pinMode(OUT_PIN, OUTPUT);
  pinMode(IN_PIN, OUTPUT);
  lcd.begin(16, 2);
}
void loop()
{
  pinMode(IN_PIN, INPUT);
digitalWrite(OUT_PIN, HIGH);
  int val = analogRead(IN_PIN);
digitalWrite(OUT_PIN, LOW);

  if (val < 1000)
  {
    pinMode(IN_PIN, OUTPUT);
    float capacitance = (float)val * IN_CAP_TO_GND / (float)(MAX_ADC_VALUE - val);
    lcd.setCursor(0,0);
lcd.print("Capacitance = ");
lcd.setCursor(0,1);
lcd.print(capacitance, 3); // for the best precision
lcd.print("pF ");
  }
  else
  {
    pinMode(IN_PIN, OUTPUT);
delay(1);
    pinMode(OUT_PIN, INPUT_PULLUP);
unsigned long u1 = micros();
unsigned long t;
int digVal;

    do
    {
      digVal = digitalRead(OUT_PIN);
      unsigned long u2 = micros();
t = u2 > u1 ? u2 - u1 : u1 - u2;
    }
    while ((digVal < 1) && (t < 400000L));

    pinMode(OUT_PIN, INPUT);
    val = analogRead(OUT_PIN);
digitalWrite(IN_PIN, HIGH);
    int dischargeTime = (int)(t / 1000L) * 5;
delay(dischargeTime);
    pinMode(OUT_PIN, OUTPUT);
    digitalWrite(OUT_PIN, LOW);
    digitalWrite(IN_PIN, LOW);
    float capacitance = -(float)t / R_PULLUP
      / log(1.0 - (float)val / (float)MAX_ADC_VALUE);
    lcd.setCursor(0,0);
lcd.print("Capacitance = ");
    if (capacitance > 1000.0)
    {
      lcd.setCursor(0,1);
lcd.print("                   ");
lcd.setCursor(0,1);
lcd.print(capacitance / 1000.0, 2);
lcd.print("uF ");
    }
    else
    {
      lcd.setCursor(0,1);
lcd.print("                   ");
lcd.setCursor(0,1);
lcd.print(capacitance, 2);
lcd.print("nF ");
    }

    while (millis() % 1000 != 0);
  }
}