

Autoranging multimeter

November 1, 2013

Microprocessor course project

Aditi Udupa 110260026

Ghaitilak Jyotsna Ravindra 110260001

Shalini Kureel 110260005

Guide: Prof. Pradeep Sarin

Project description

We have tried to build a multimeter capable of measuring voltage, resistance and capacitance and make it autoranging(i.e without need for external setting of range). We have used simple voltage dividers to change ranges and the Arduino is programmed to change ranges accordingly.

Autoranging Voltmeter

We have used branches of resistors to measure any voltage from 0V to 50V. When we supply a voltage (V_{in}) to this voltmeter, if this voltage supplied is lesser 5V then, Pin 2 turns is made an input pin. So no current will flow through resistors and hence the voltage will be read by this voltmeter as it was provided. Now, in second case, if we supply a voltage greater than 5V ,

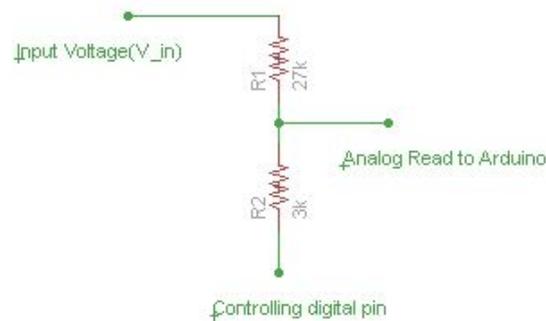


Figure 1: Voltage divider used for measuring voltage

we can not read this voltage as it is because, if we read its original value then we are supplying a voltage greater than 5V to the arduino which will damage it. So, we need to scale down the voltage such that the voltage provided to arduino is less than 5V. So, in this case we put Pin 2 as output pin and keep it low i.e, at ground. Now, the current will flow through this branch of resistors and V_{out} lesser than 5V. and for this case the code written will fetch the actual value of input voltage.

Results and improvements

This worked quite well and was fairly accurate(to two decimal places). We could incorporate more networks to cover higher ranges. We could place

buffers at the input to the arduino and for safety of the arduino, a 5.1V Zener diode could be used.

Autoranging ohmmeter

This Ohmmeter measures different values of resistances ranging from 0 to 10 M Ω . When we want to measure a resistance, we keep one of the pin 1, 2 or 3 as output at HIGH (5V) and other two are inputs So, the current will flow through only one branch at a time and we measure output voltage at pin 5 and using the relation between V_{out} and R_{out} we find the value of R_{out} . In the program, we always start off by assuming that the resistance is in the highest and then proceed to fine-tune the answer by increasing resolution i.e by changing networks.

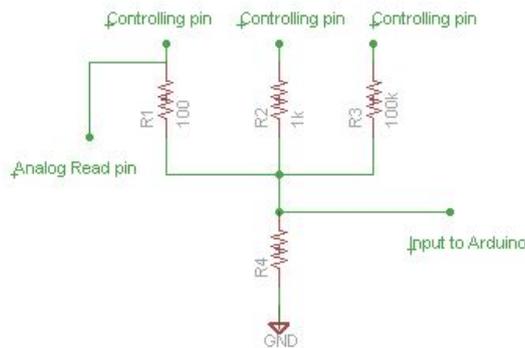


Figure 2: Voltage divider used to measure resistance

Results and improvements

This shows very accurate results right from a few ohms to 1 M Ω . The accuracy beyond this drops slightly. There are a few bugs in the code which cause the program to stop responding correctly after a few iterations. This needs to be sorted out. Buffers at the input pins need to be used. We got to know that there is a high-impedance state for the arduino pins. The usage of this would increase resolution even more.

Autoranging capacitance meter

This capacitance meter is designed to measure values from 1 nF to 100 μ F. The capacitance is calculated by charging the capacitor for a fixed amount of time(1 ms) and measuring the voltage across it. Autoranging is done by changing the resistor by which to charge the capacitor. Charging a μ F capacitor with a M Ω would take too long and hence the voltage after 1 ms will be too small. In the program, we always start off by assuming that the capacitor is in the highest range and then proceed to find the correct range. The capacitance is found by simply using: $V = V_0(1 - \exp(-t/RC))$

Results and improvements

It works very accurately for small values of capacitance but the error increases quite a bit in the 10-100 μ F range. This is an area which needs to be improved upon.

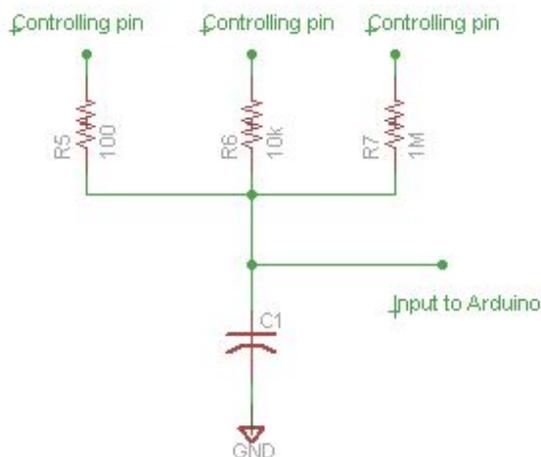


Figure 3: Voltage divider used to measure capacitance

LCD display

Since we were short on pins and the in-built library for the LCD display provided by Arduino requires a minimum of 6 digital pins, we tried to use an alternate way by using only three pins and a different LCD library. We built a circuit using a serial-in parallel-out shift register to perform this task.

But there were some problems in the library which we have not yet been able to resolve.

Work distribution

1. Aditi Udupa - project planning, coding, debugging, circuit testing, report-making
2. Ghaitilak Jyotsna Ravindra - coding, debugging, circuit designing
3. Shalini Kureel - soldering, LCD display circuit and software interfacing, report-making