TeachEngineering

Ignite STEM learning in K-12

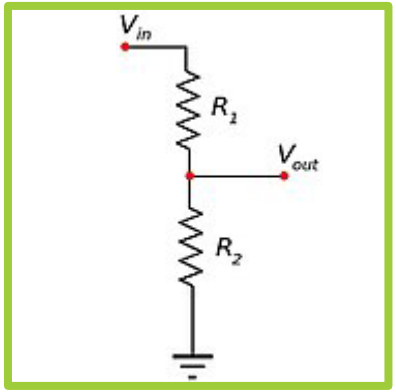
Oh Baby! Contractions and Calculations / Activity Part 2

Voltage Dividers

In this activity, you will connect the force-sensitive resistor to an Arduino microcontroller to read the sensor values. The analog inputs of an Arduino microcontroller measure voltage and not resistance. A voltage divider is used to interface between the two. Based on the input voltage, the resistance of the FSR can be calculated.

The input of the analog pin is the voltage across the pull-down resistor. The pull-down resistor ensures a known state so when no input (no pressure) is applied, it is essentially an open circuit.

The FSR and the pull-up resistor are in series and their common node is connected to the analog pin. The source voltage Vin is distributed across both resistors which forms a voltage divider. The voltage divides in direct proportion to the resistance.



Voltage divider circuit

Voltage Dividers, cont.

For our circuit:
*** Vin = 5 V**
*** R1 = FSR variable resistance**
*** R2 = 10 kΩ pull-up resistor**

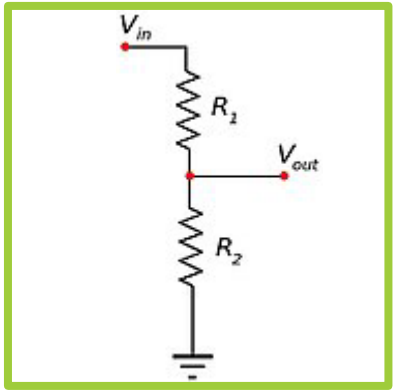
We are interested in V_{out}, which is the voltage across the pull-up resistor R2, but you can also calculate the voltage across the FSR R1.

The voltage across R_1:

$$V_1 = V_{in} \cdot \frac{R_1}{R_1 + R_2}$$

The voltage across R_2, or V_{out}:

$$V_1 = V_{in} \cdot \frac{R_2}{R_1 + R_2}$$

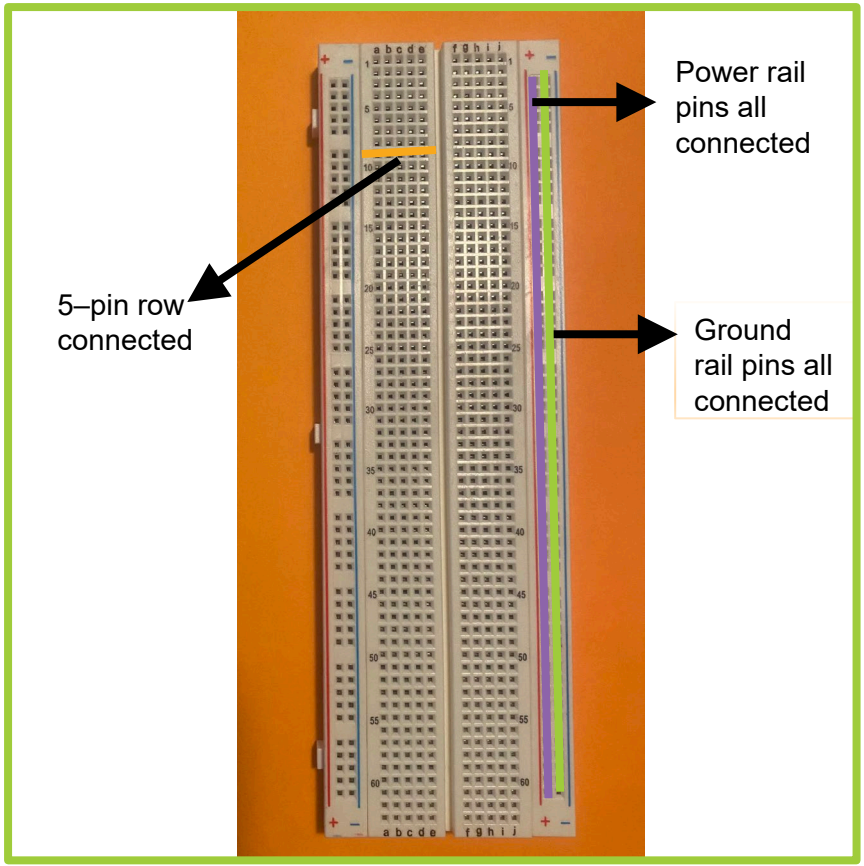


Voltage divider circuit

Breadboarding Basics

A breadboard is used to construct prototype (temporary) circuits for testing or simply to try out an idea. No soldering is needed so it is easy to change connections and replace components. Parts will not be damaged by soldering so they will be available for re-use afterwards.

Don’t be fooled! You can still damage components due to overload or short circuit resulting in blue smoke.

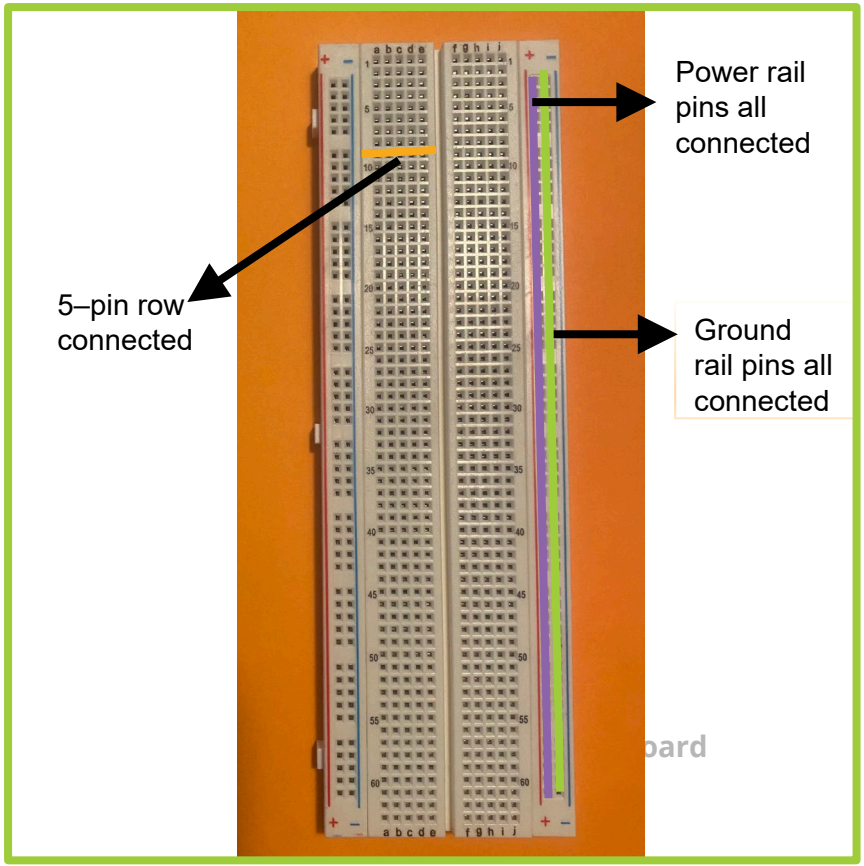


Solderless breadboard

Conductive Layout

Power Rails: **When you build a circuit you may need many components requiring power. The entire row of 30 holes is electrically connected together. There is another row on the opposite side providing another 30 holes. The two rows are not connected so if you want them to be, you can add a jumper wire between the rows.**

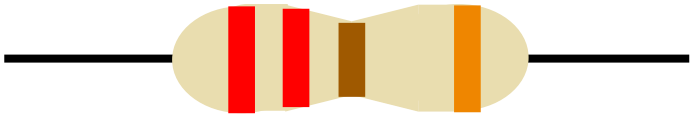
Terminal Strips: **Each group of 5 pins is also connected together. Note that the columns on either side of the middle divider are *not* connected together. This means a maximum of five components per column.**



Resistor Bands Refresher

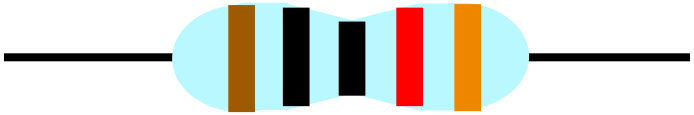
Color	Digit	Mulitplier	0.5%
Black	0	$x10^0$	X
Brown	1	$x10^1$	±1%
Red	2	$x10^2$	±2%
Orange	3	$x10^3$	X
Yellow	4	$x10^4$	X
Green	5	$x10^5$	±0.5%
Blue	6	$x10^6$	±0.25%
Purple	7	$x10^7$	±0.1%
Grey	8	$x10^8$	±0.005%
White	9	$x10^9$	X
Silver	X	X	±10%
Gold	X	X	±5%

3-Band Resistor	Digit	Digit	Multiplier	X	X
4-Band Resistor	Digit	Digit	Multiplier	Tolerance	X
5-Band Resistor	Digit	Digit	Digit	Multiplier	Tolerance



220 Ω Resistor

Red	Red	Brown	Gold
2	2	x10	5%
220 Ω			



10,000 Ω Resistor

Brown	Black	Black	Red	Gold
1	0	0	x100	5%
10,000 Ω				

Digital Input/Output

You are probably familiar with circuits that use the digital input/output pins on an Arduino microcontroller to turn a switch or an LED on and off. The digital pins have only two values which you can look at in a variety of ways.

Digital Input/Output Pins	
On	Off
Closed Circuit	Open Circuit
High	Low
+5 V	0 V
Binary 1	Binary 0

Analog Input/Output

The Arduino's built-in analog-to-digital converter (ADC) accepts analog input (continuous signal) as voltages in the range 0–~5 V and automatically converts them to a digital value (discrete values) 0–1023. The ADC is a 10-bit converter which means it can handle 1024 distinct values.

ADC Converter	
Analog Input	Digital Output
0 mV	0
Anything in-between	Rounded digital value
5000 mV	1023

The analogRead() line is the command that reads the digital value from the specified pin. The map() function proportionally converts one range of values to another: fsrVoltage = map(FSRval, 0, 1023, 0, 5000). The map value only works with integers and will round an output to the nearest integer. This is why we will work in mV rather than V.