LAB2-MC

Sensors and DC Motors

CMP(N)211

Spring 2021

By: Salma Abdel Monem

Lab Objectives

✓Control DC motors using PWM and H-bridge

✓Read Analog Signals.

✓Pull up resistance.

✓Temperature sensors types

✓Map thermistor readings to Celsius temperature

✓Use the SimuLIDE.

Recap

Till Now we have learnt the following:

✓ Read Digital Signals . Example ?

✓ Write Digital Signals. Example ?

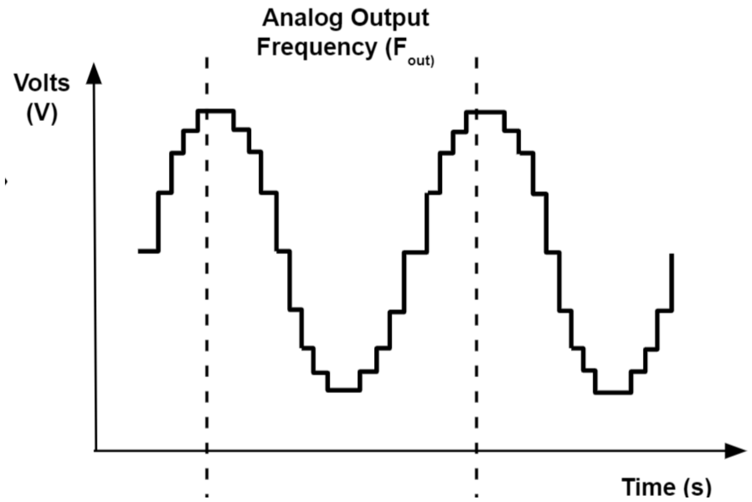
Now it’s time to learn how to Read and Write Analog Signals

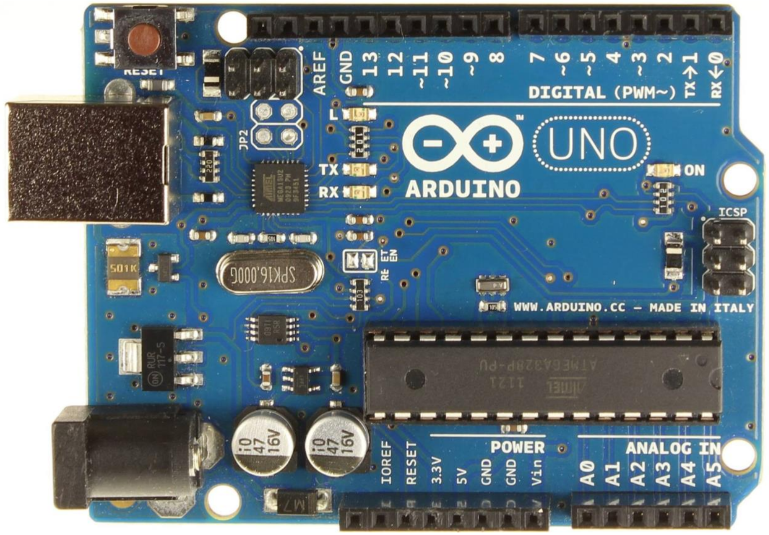
PWM

We will get into it’s details in the next lab.

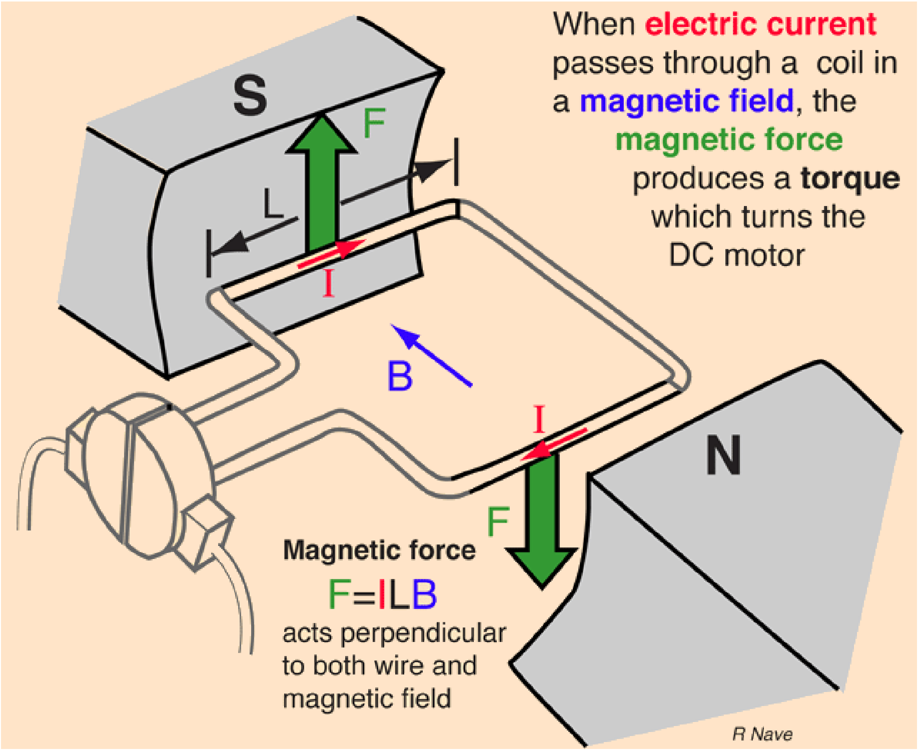
✓Till now, what you need to know is that PWM pins(~) produce Analog like / discretized signal from (0 to 255).

✓Example of writing to PWM pin analogWrite(9, 255);





DC motors



When **electric current** passes through a coil in a **magnetic field**, the **magnetic force** produces a **torque** which turns the DC motor

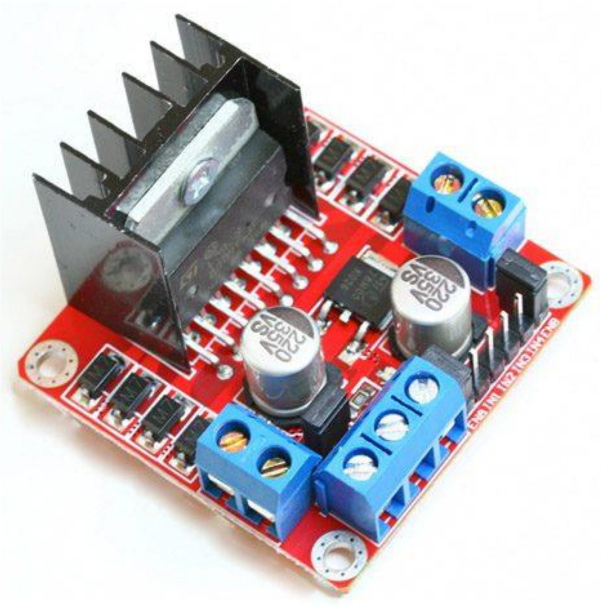
Magnetic force

$F=ILB$

acts perpendicular to both wire and magnetic field

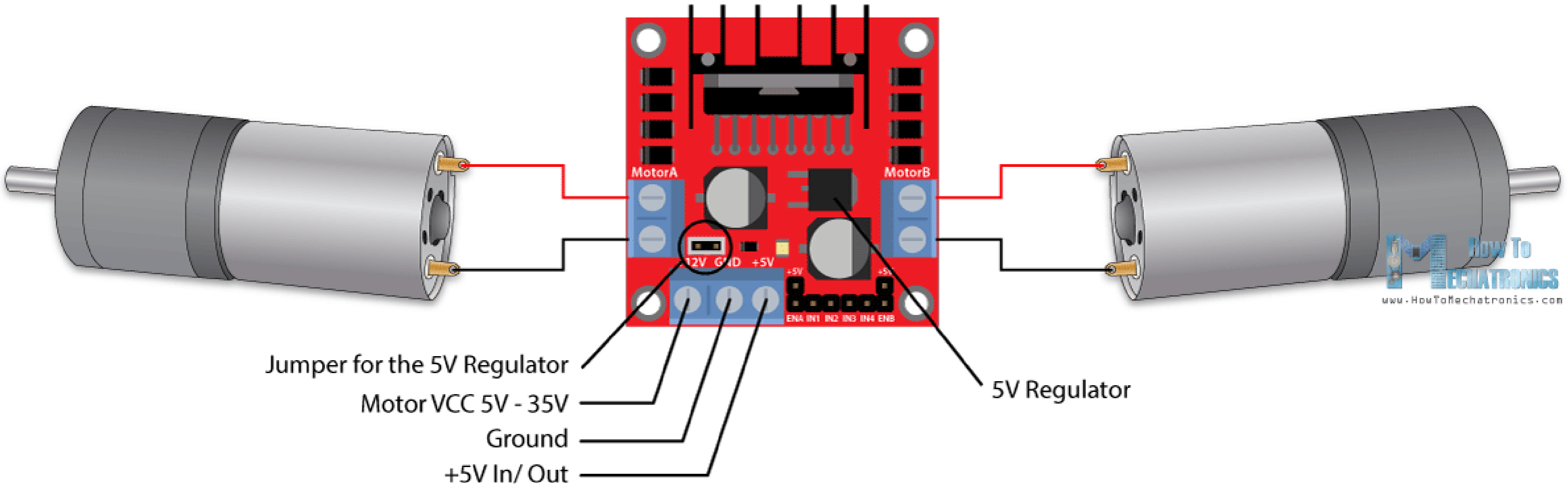
DC motors and H-Bridge

✓The H-Bridge (L298p here) can be used to control the direction of the spinning of the motor and it’s speed as well.



DC motors and H-Bridge

✓L298p can connect two motors



Jumper for the 5V Regulator

Motor VCC 5V - 35V

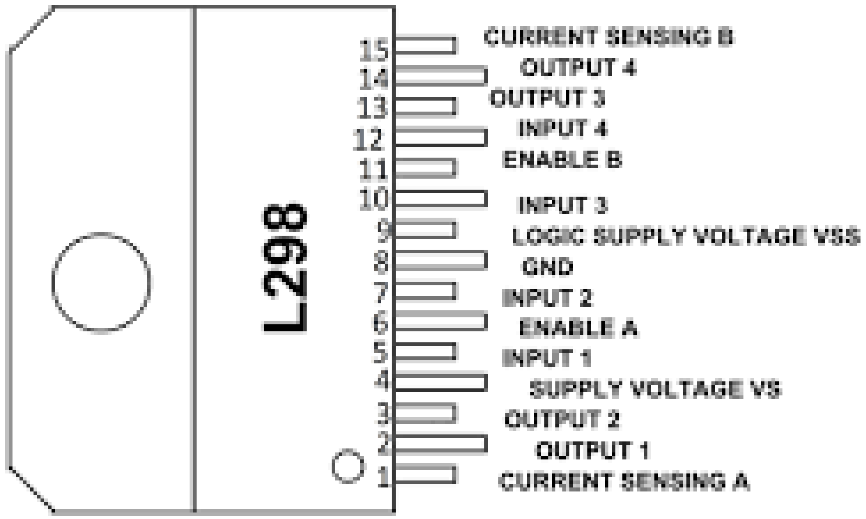
Ground

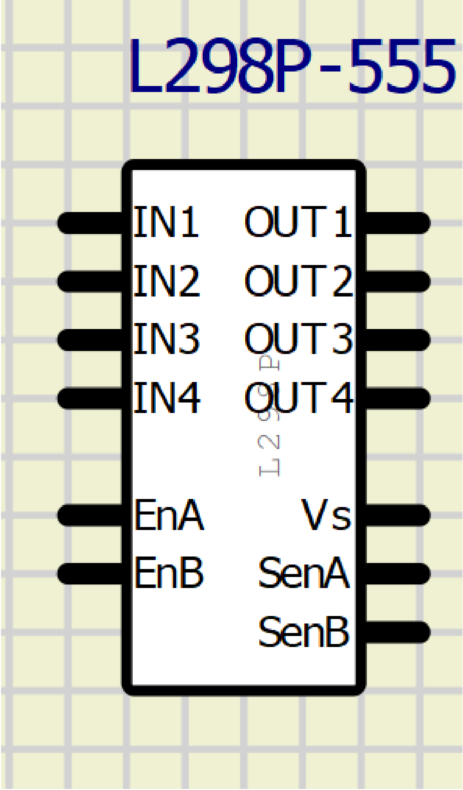
+5V In/ Out

5V Regulator

DC motors and H-Bridge

✓L298p pinout

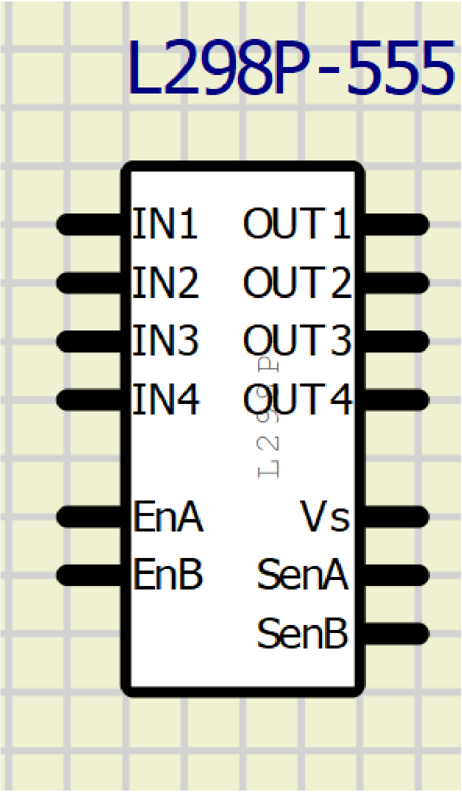




DC motors and H-Bridge

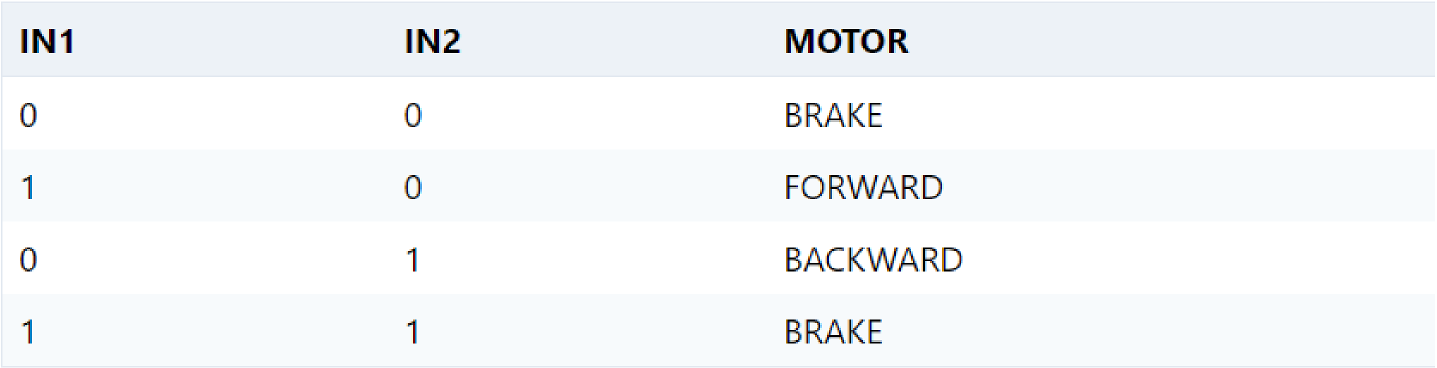
L298p Pinout

- ✓IN1 and IN2 control direction of motor 1 (arduino)
- ✓IN3 and IN3 control direction of motor 2 (arduino)
- ✓EnA controls the speed of the motor 1 (PWM arduino)
- ✓EnB controls the speed of the motor 2 (PWM arduino)
- ✓OUT1 and OUT2 are connected to the both edges of the motor 1
- ✓OUT3 and OUT4 are connected to the both edges of the motor 2
- ✓Vs is connected to 3.3 v from arduino

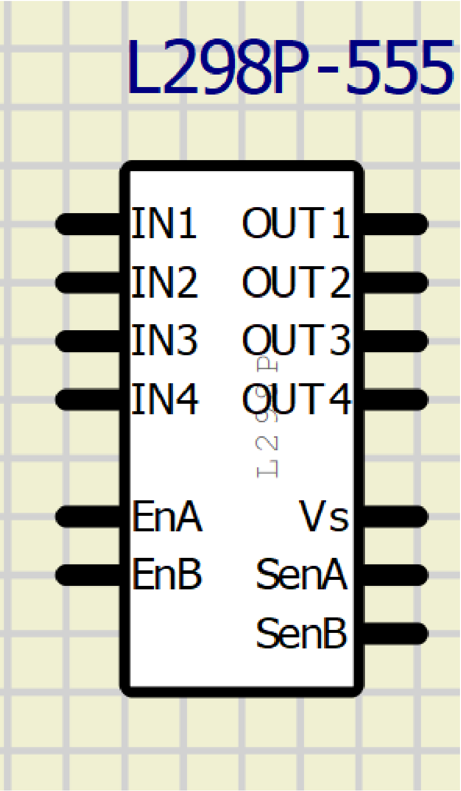


DC motors and H-Bridge

L298p Pinout

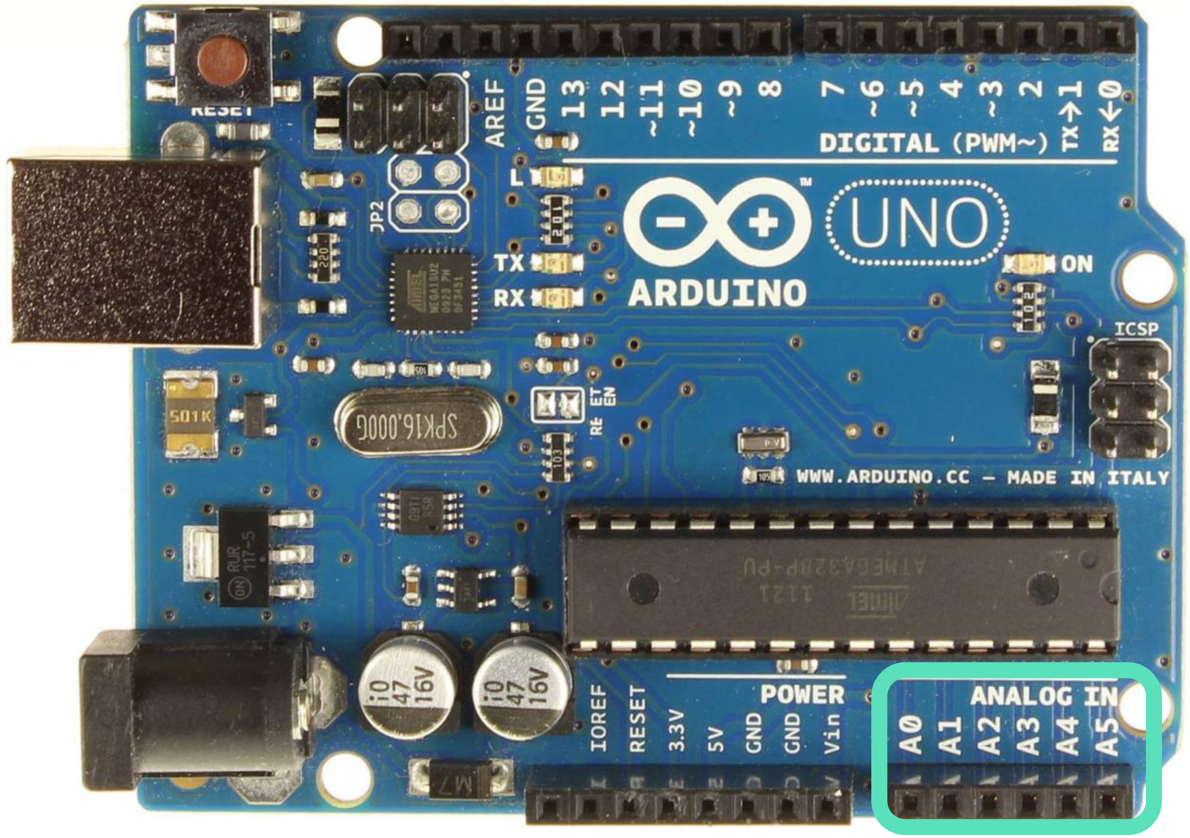


IN1	IN2	MOTOR
0	0	BRAKE
1	0	FORWARD
0	1	BACKWARD
1	1	BRAKE



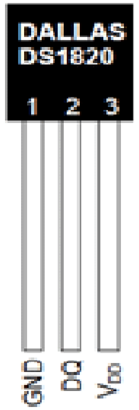
Read Analog Signal

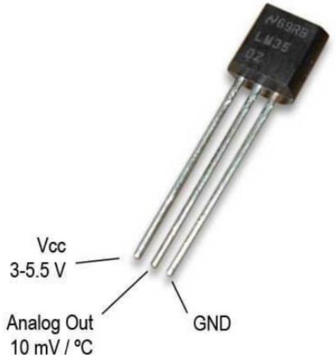
- Use the analog pins {A0 ….A5} to read analog signals.
- The analog input can vary from { 0 to 1023} .
- You might want to map 0 to LOW , and 1023 to HIGH {5volt} in Arduino.
- Pin numbers are (0 : 5) and no need to set them as inputs in the setups.
- Use analogRead(Pin)



Temperature sensor types

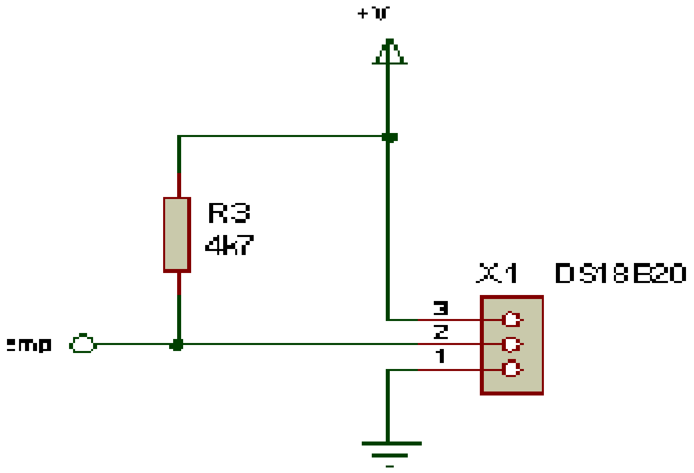
- Thermistor (cheap , wide range , less accurate, non linear)
- Thermocouple
- RTDs (resistive temperature detectors)
- Digital thermometer Ics (more expensive, more accurate, linear)
- Analog thermometer ICs

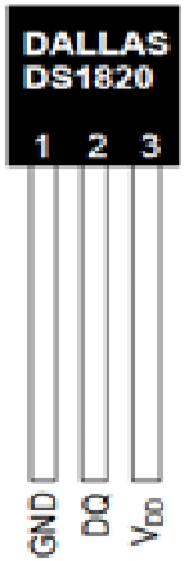




Digital thermometer Ic sensor (DS1820)

- It has three legs like the following:

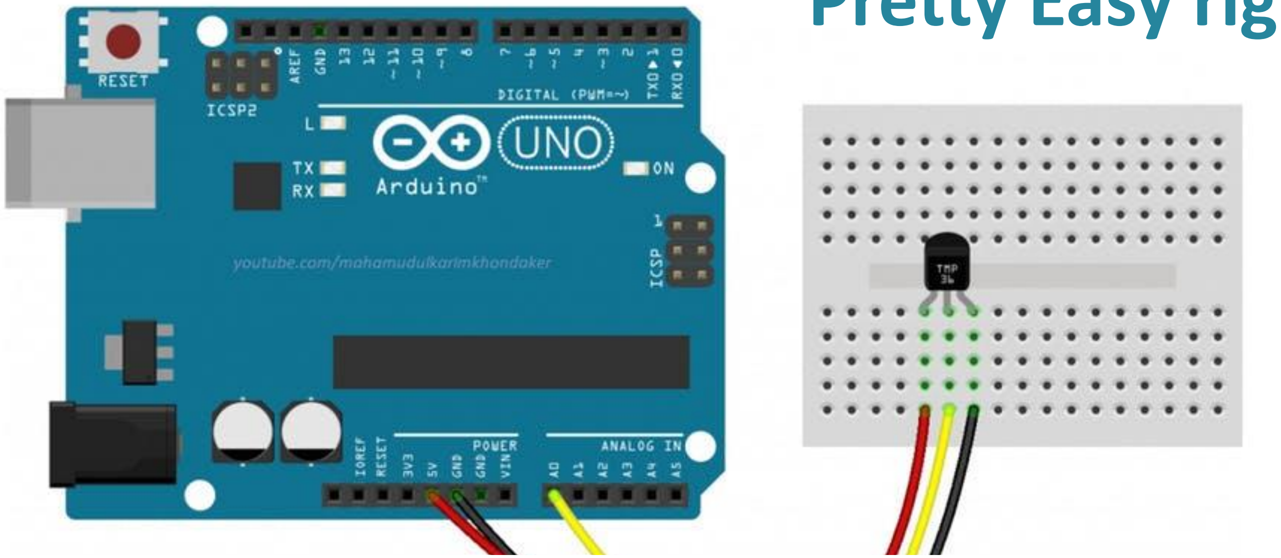




So let’s connect it with an Arduino ??

Yet this is not recommended !

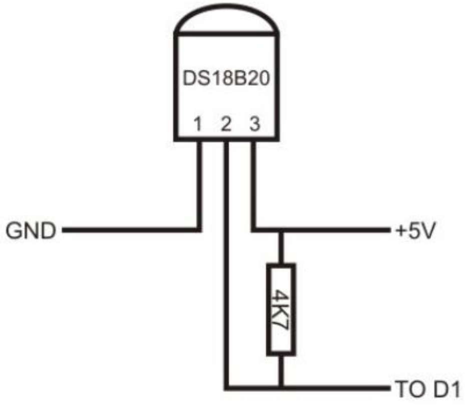
Pretty Easy right?

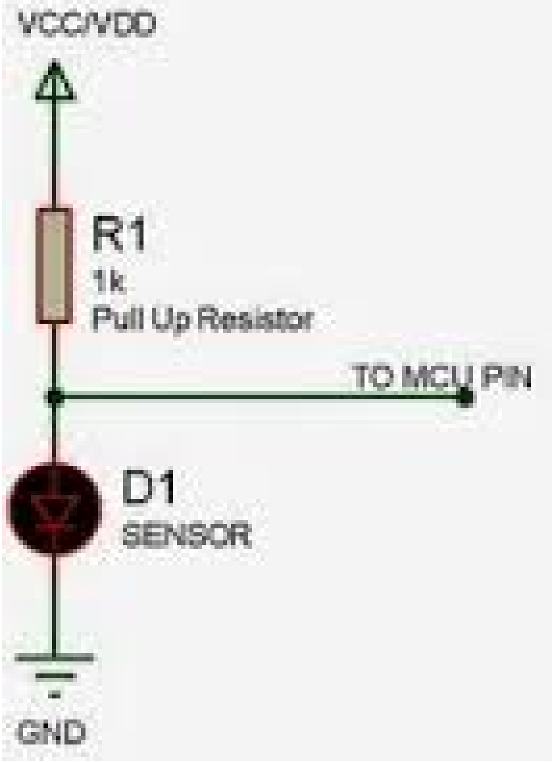


This might cause unstable readings if the sensor is not sending a reading signal

Solution? Pull Up resistors

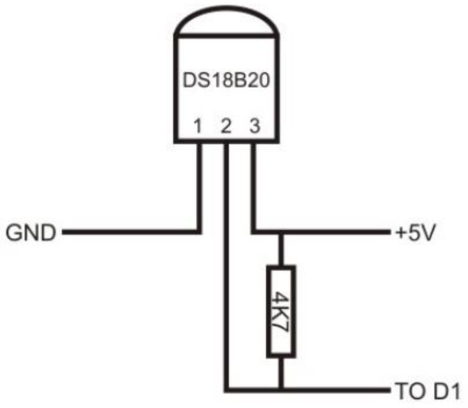
- What happens when the sensor is open circuit?
- Answer is : floating input !!

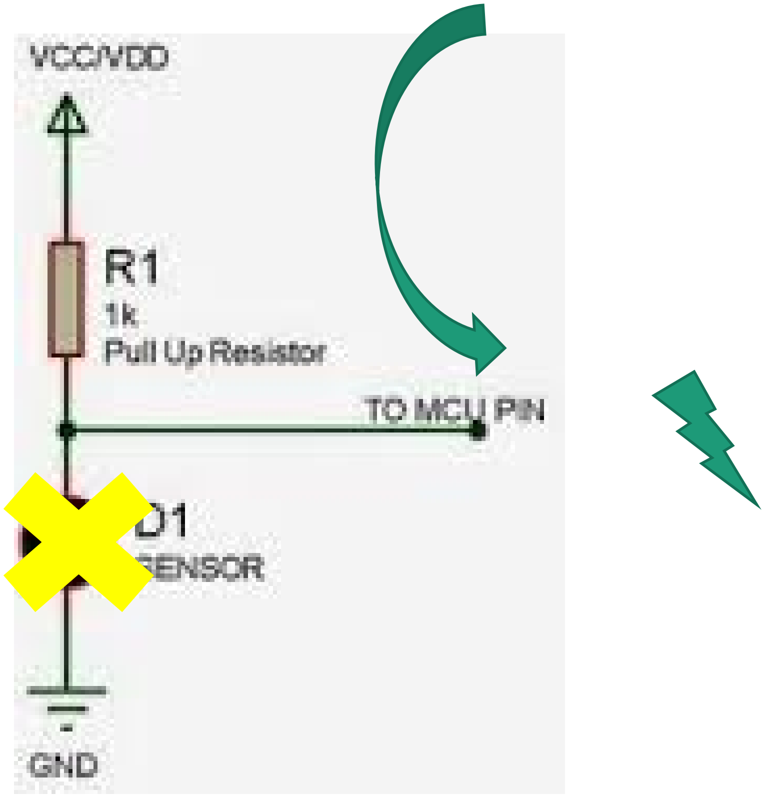




Solution? Pull Up resistors

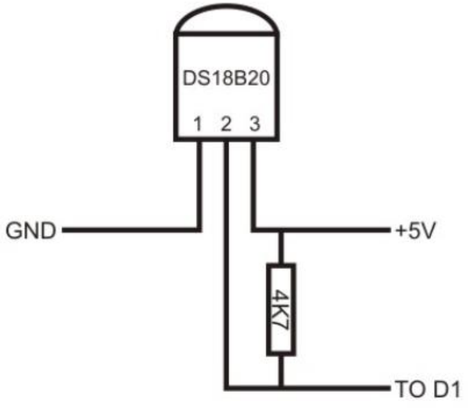
- The sensor is open circuit

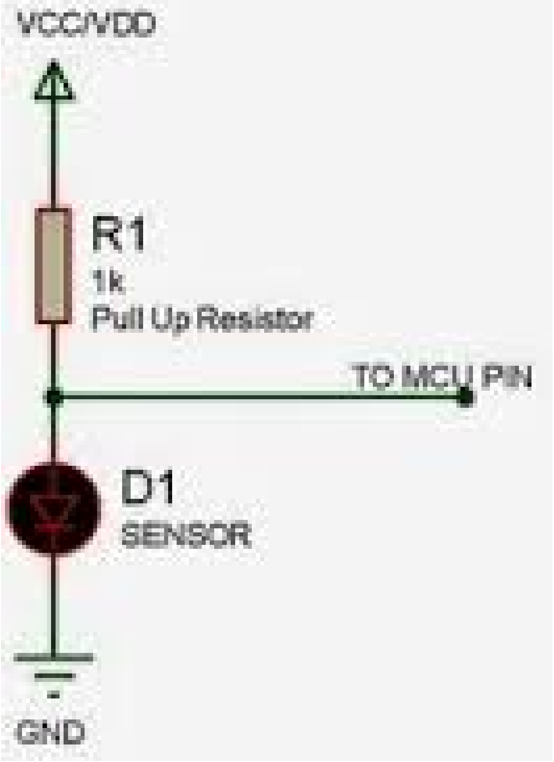




Solution? Pull Up resistors

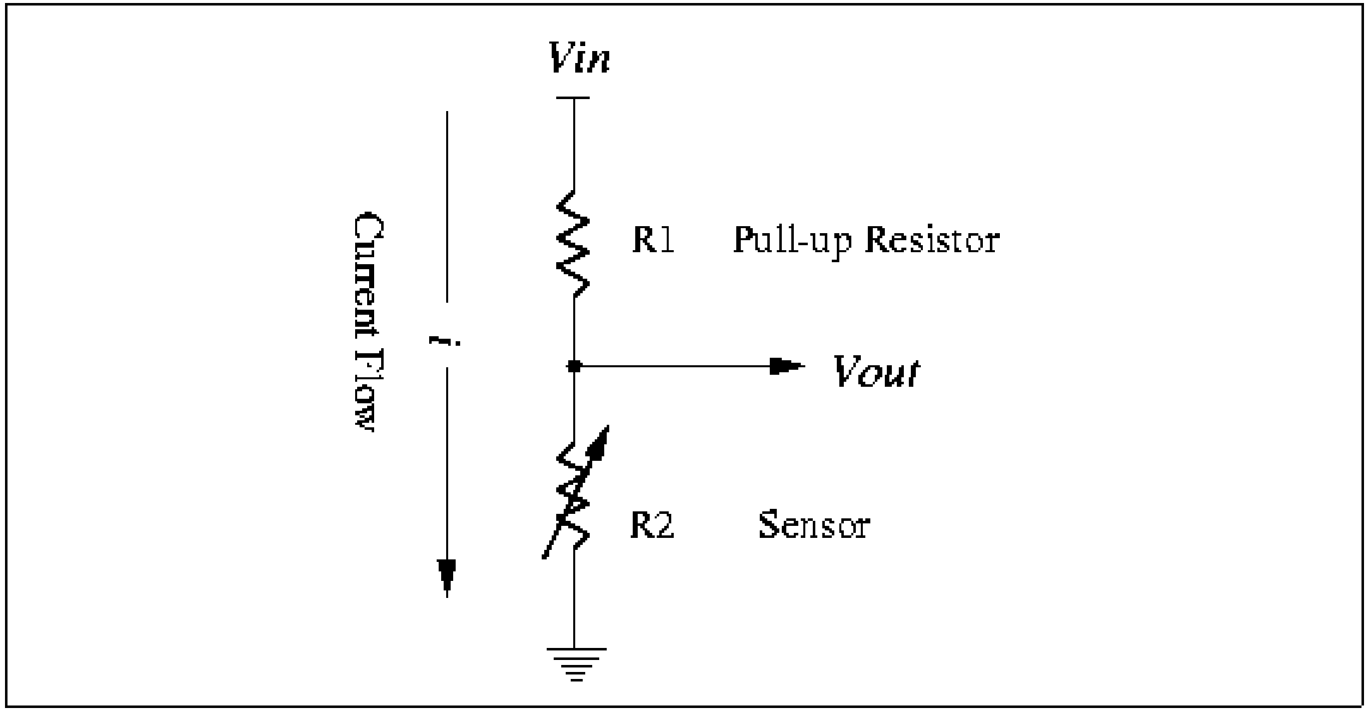
- The sensor is closed circuit





What will be the reading then?

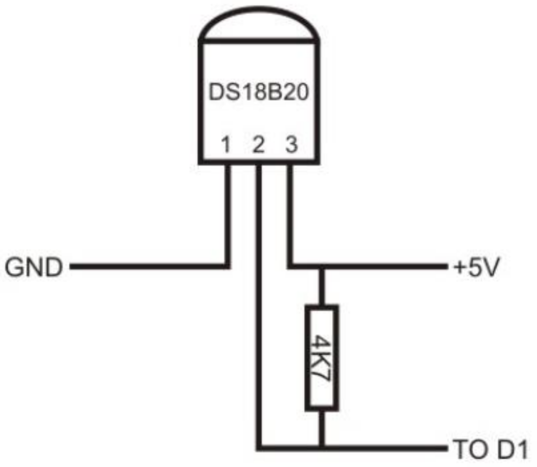
- Remember the voltage divider rule.
- When the sensor is closed--circuit; it’s resistance changes with the temperature.
- The reading voltage *Vout* is then what the Arduino senses.

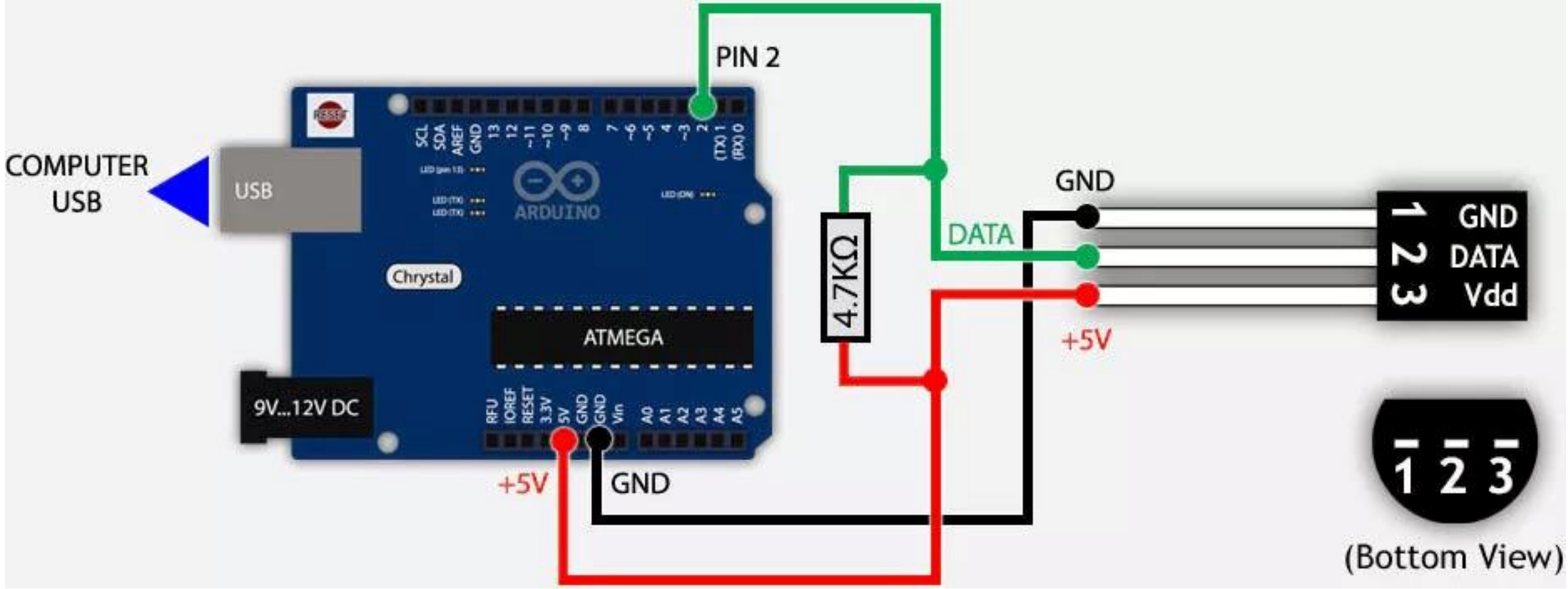


$$\mathbf{Vout} = \frac{\boldsymbol{R1}}{\boldsymbol{R1} + \boldsymbol{R2}} * \boldsymbol{Vin}$$

Then how should we connect it ?

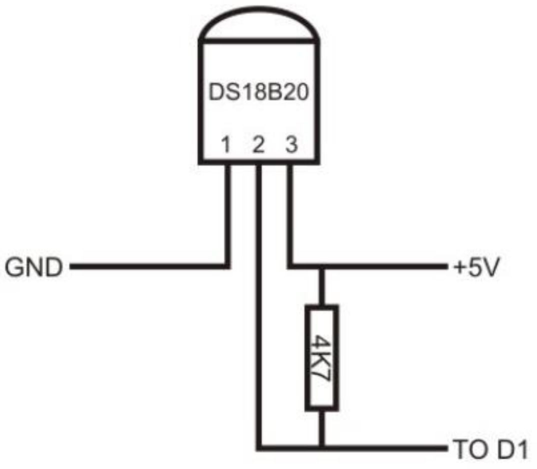
- Using a pull-up resistance

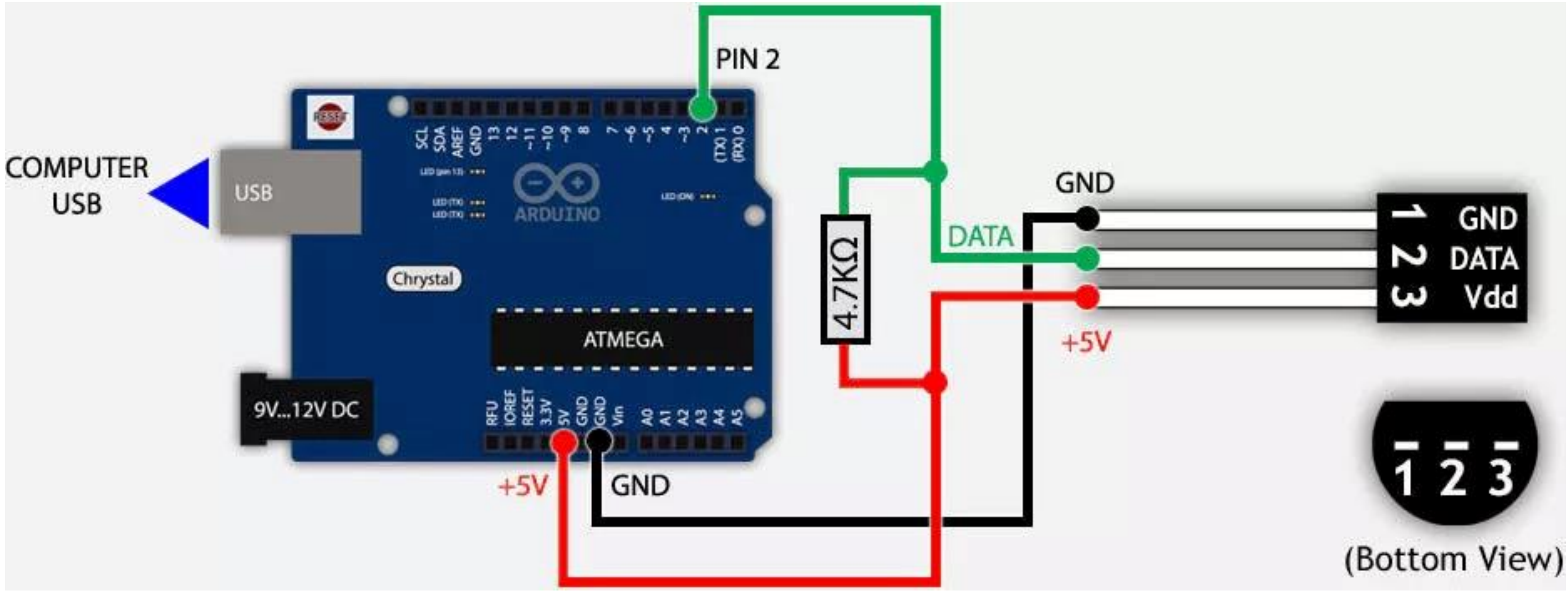




Then how should we connect it ?

- Using a pull-up resistance



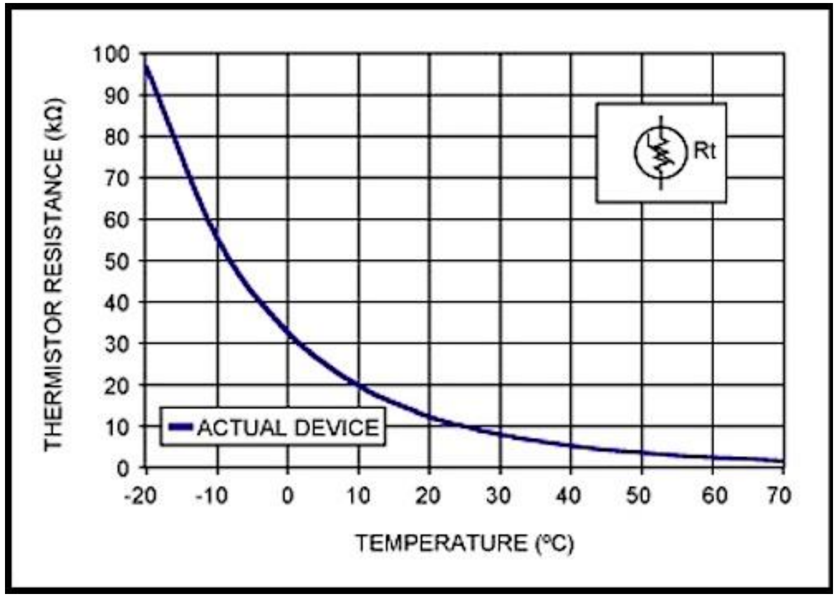


Additional notes

- You have first to use the 1-wire library with arduino.
- Following the steps in this site.
- You can also see the details of the ADC in the datasheet.
- It has linear mapping between the reading and the temperature.

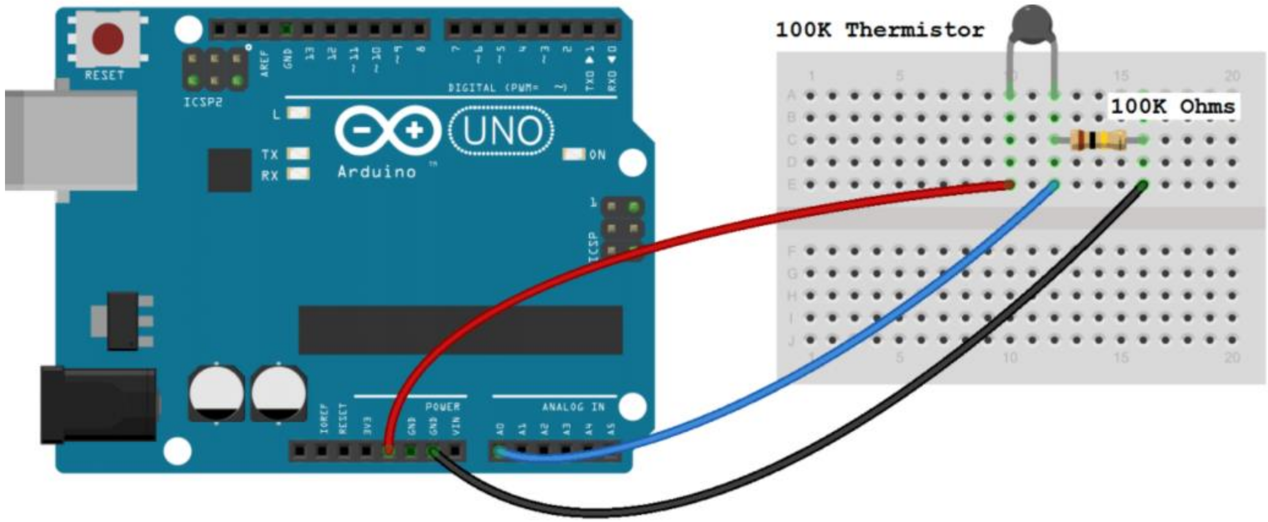
Thermistor(NTC)

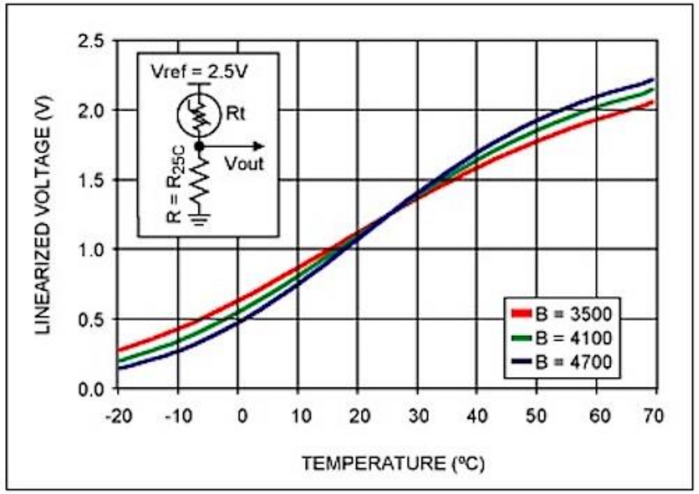
- It has two legs like the following
- Simple and cheap but it doesn’t have linear mapping for temperature vs. it’s analog readings.



Thermistor(NTC)

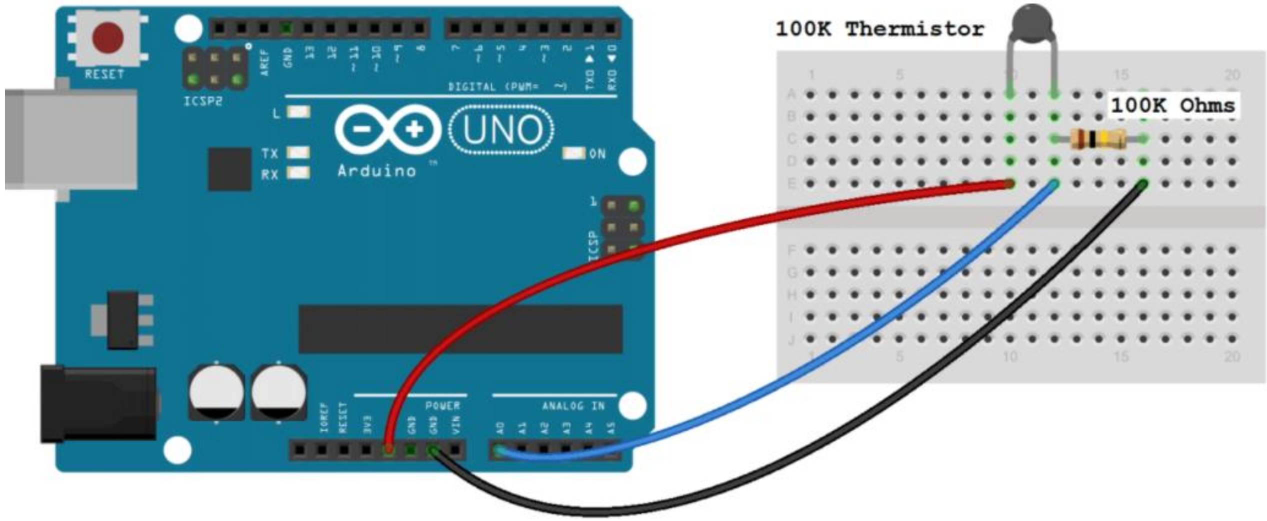
- Linear mapping technique (Voltage Mode Linearization)
- Places the thermistor in series with a normal resistor forming a voltage divider circuit.

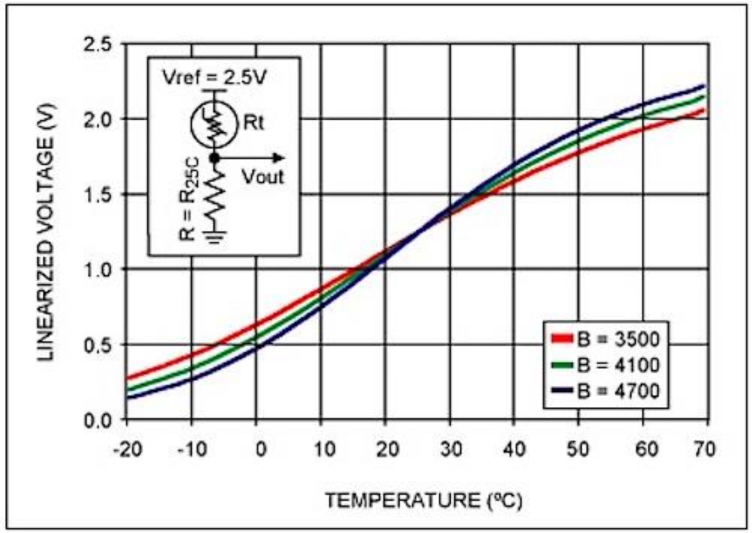




Thermistor(NTC)

- Linear mapping technique (Voltage Mode Linearization)
- If the resistor's value is equal to the thermistor's resistance at room temperature, then the region of linearization will be symmetrical around room temperature





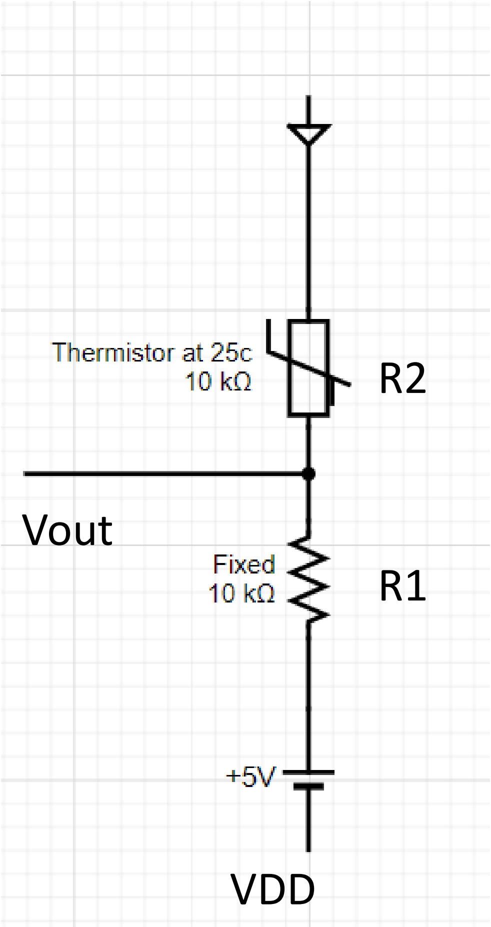
So what is the actual temperature ?

- Remember that the Arduino reads the voltage *Vout*.
- Using the voltage Vin=1023 and *Vout* (reading) from {0:1023} we can get the sensor's resistance right ?

$$\mathbf{Vout} = \frac{\boldsymbol{R1}}{\boldsymbol{R1} + \boldsymbol{R2}} * \boldsymbol{Vin}$$

$$\Big\downarrow$$

$$\mathbf{R2} = \boldsymbol{R1} / \left(\frac{\boldsymbol{Vin}}{\boldsymbol{Vout}} - \mathbf{1} \right)$$



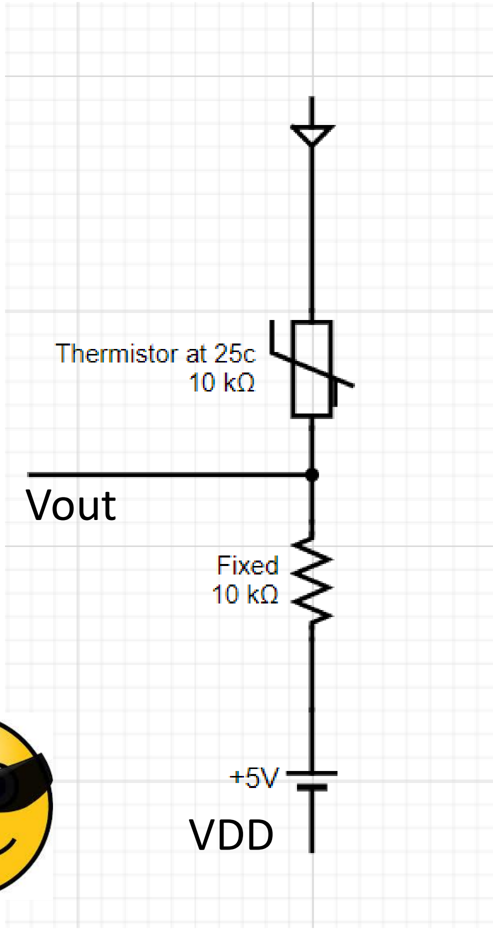
So what is the actual temperature ?

$$\mathbf{R2} = \boldsymbol{R1}/\left(\frac{\boldsymbol{Vin}}{\boldsymbol{Vout}} - \mathbf{1}\right)$$

- Using the “Steinhart–Hart equation”
- A,B and C are constants related with the sensor.
- T is temperature and R is the sensor’s resistance.

$$\frac{1}{T} = A + B \ln R + C(\ln R)^3,$$

- T is the current temperature + 273.15 (To at 25 C)
- So subtract 273.15 from T and you got the temperature.



Exercise

- Use the SimuLIDE to Perform the following requirement.
- Read the temperature from a thermistor and control a dc motor using it.
- If the reading is between (0 and 50 c) the motor stops.
- If the reading is between (51 and 100c) the motor moves forward with half the speed.
- If the reading is higher than 100c the motor moves backwards with maximum speed.
- Submit a link to google drive folder file including your (.ino file , .hex file , .simu file and no more than 3 minutes video showing all the testcases)
- Name all these files with your name _ id_lab2.
- No cooperation or plagiarism (individual work).
- Violation of the submission rules will cause zero grade.

Appendix

- Thermistor constants for NTC 10k ohm are :
- float A = 1.009249522e-03, B = 2.378405444e-04, C = 2.019202697e-07;
- You can use log function in arduino instead of ln.