Arduino workshop Imperial College Winter 2017 – session #1

Exercise #1 (modifying 'Blink'):

Connect Arduino to the computer
Open the IDE
Upload the blink program
Change the blink frequency to 2Hz

Exercise #2 (reading sensor):

Reading an analog sensor (TMP 36)

analogRead; Serial.print; delay; Serial.println; if, else logical conditions; Serial.begin

The TMP36 is a low voltage, precision centigrade temperature sensor. It provides a voltage output that is linearly proportional to the Celsius temperature. It also doesn't require any external calibration to provide typical accuracies of ±1°C at +25°C and ±2°C over the −40°C to +125°C temperature range.

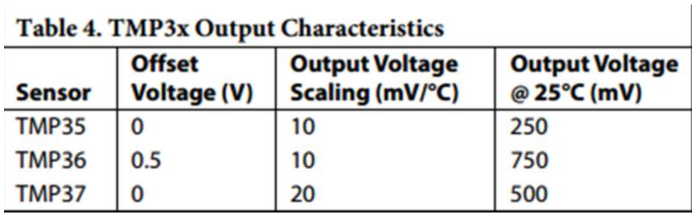


Table 4. TMP3x Output Characteristics

Sensor	Offset Voltage (V)	Output Voltage Scaling (mV/°C)	Output Voltage @ 25°C (mV)
TMP35	0	10	250
TMP36	0.5	10	750
TMP37	0	20	500

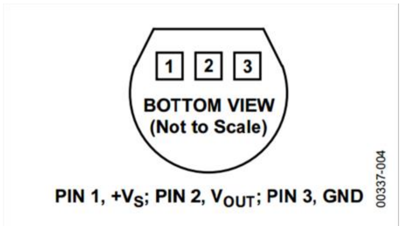
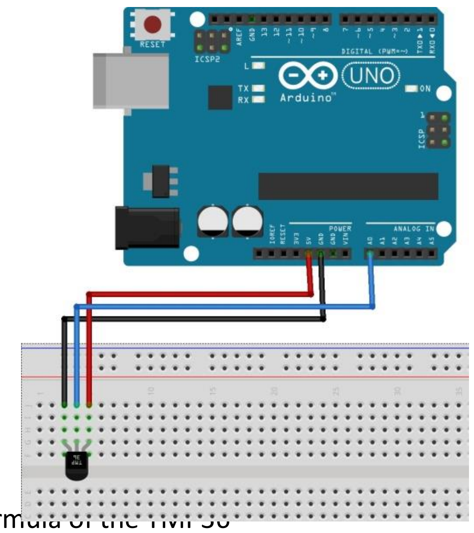


Figure 4. T-3 (TO-92)

For the complete datasheet of the TMP36, please check the 'links' page at the workshop webpages.



<u>The circuit:</u>
* thermometer TMP 36 (pin 2 – center) attached to analog input 0
* thermometer TMP 36 (pin 3) to ground
* thermometer TMP 36 (pin 1) to +5V

!!! Connecting the TMP in an opposite polarity causes an

excessive heating that may kill the device and/or cause burns !!!

Task: Read the temperature of the device and print it every one second

Hints:

1. read the value of the sensor using analogRead())
2. convert the reading into voltage (x refereneceVoltage x 2^{-10})
3. convert the calculated voltage to temperature using the conversion formula of the TMP 36

Modifications:

1. print the time in milliseconds for each measurement (relative)
2. improve the resolution of your measurement (hint: change the variable of 'analogReference()')
3. change the sampling rate to **exactly** 1HZ

<u>Exercise #3 (reading sensor and activating actuator):</u>

Controlling LED illumination s

Light emitting diodes (LEDs) are two-terminals devices that emit light at an intensity that is proportional to the current the flows through the device.

LEDs are polar devices: the long leg should be connected to the positive voltage and the short leg should be connected to the GND pin.

In order not to kill an LED, its current should be limited. Here we use 150 Ohms resistor (which is connected in series to the LED) to limit the LED current to below 20 mA.

<u>The circuit:</u>

* thermometer TMP 36 (pin 2 – center) attached to analog input 0

* thermometer TMP 36 (pin 3) to ground

* thermometer TMP 36 (pin 1) to +5V

* LED positive terminal (long leg) to pin 11 (digital)

* LED negative terminal (short leg) to 150 Ohm resistor

* the other leg of the 150 Ohm resistor to GND pin

Task: Turn on the LED if the light exceeds 27 degrees; otherwise turn it off

Hint:

1. Use the circuit form example1 to read the temperature using the TMP36
2. Turn on the LED using “digitalWrite(XXX,HIGH)”
3. Turn off the LED using “digitalWrite(XXX,LOW)”
4. Use conditional statements: if(){ }

Modifications:

1. Vary the LED intensity according to the reading temperature ('analog write')
2. Use the serial monitor to define the threshold temperature (integer) above which the LEDs is turned on
3. Make the LED blink **without** using the function 'delay' (allows Arduino to continue work in
4. Count the number of times the temperature rises above 28^{o}C