

Table of Contents

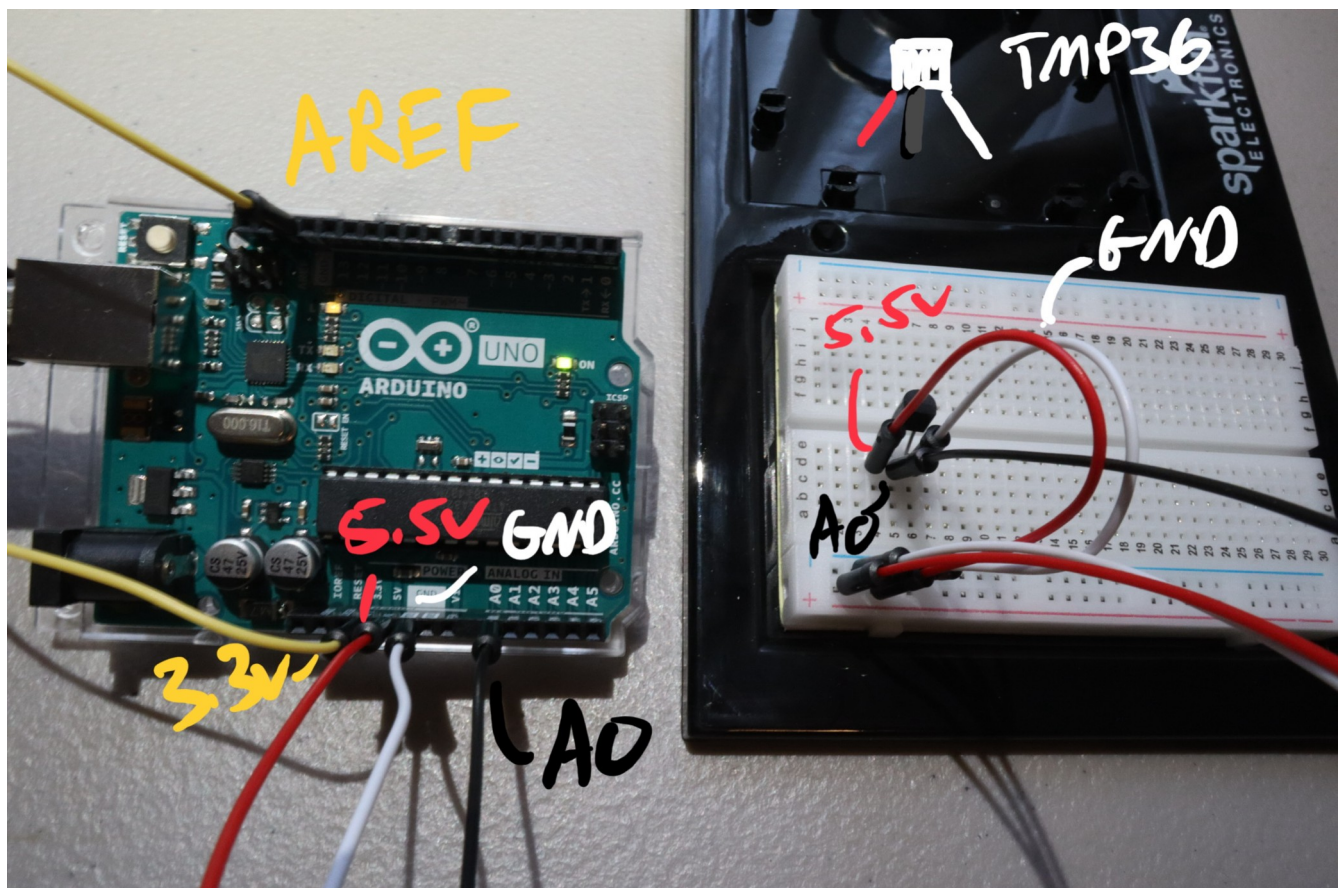
Requirements.....	.1
Design.....	.1
Implementation.....	.2
Demo.....	.3
References.....	.4

Requirements

This project uses an Arduino UNO, a tmp36gz temperature sensor, and wiring to read from a temperature sensor in specific intervals, using interrupts to periodically poll the sensor during a round robin loop. This project was written and tested using the Arduino IDE in C to implement the Project 2 requirements as provided.

Design

The Arduino UNO connects via USB for power and serial output to the host. The TMP36 sensor has 3 pins and is inserted into the breadboard where the outer pins are connected to 5.5V and ground, and the middle pin is connected to the analog A0 pin on the arduino. A wire jumps 3.3V to AREF in order to use the reference voltage to improve the accuracy of the temperature sensor.



Implementation

The Project is written in C like code on Arduino IDE.

```
``` proj2-temp.ino

/*
Project 2 - Temperature Sensor with Interrupts
Alex Shah
2/18/24
*/

//using a tmp36gz on pin a0
int iPin = A0;

volatile float temperatureF = 0;
volatile bool newReadingAvailable = false;

void setup() {
 Serial.begin(9600);
 //use AREF port to enhance precision
 analogReference(EXTERNAL);
 attachInterrupt(digitalPinToInterrupt(2), readTemperature,
FALLING);
 cli(); //suspend interrupts
 //clear timers
 TCCR1A = 0;
 TCCR1B = 0;
 //set cs12, cs10 bits for prescaler 1024
 TCCR1B |= B000000101;
 //use compare match by setting OCIE1A to 1
 TIMSK1 |= B000000010;
 //set compare register A ~ 5 sec
 OCR1A = 15625;
 //Reset Timer 1 value to 0
 TCNT1 = 0;
 sei(); //resume interrupts
}

void loop() {
 if (newReadingAvailable) {
 Serial.println(temperatureF);
 newReadingAvailable = false;
 }
}

void readTemperature() {
 int sensorValue = analogRead(iPin);
 //external reference voltage and resolution
```

```

float fRef = 3.3;
float fRes = 1024.0;
// celsius = (sensor value * (voltage in mw / resolution) -
offset) * 100
float fVlt = sensorValue * (fRef / fRes);
float fTempC = (fVlt - 0.5) * 100;
temperatureF = (fTempC * 9 / 5) + 32;
newReadingAvailable = true;
}

ISR(TIMER1_COMPA_vect) {
 TCNT1 = 0;
 readTemperature();
}

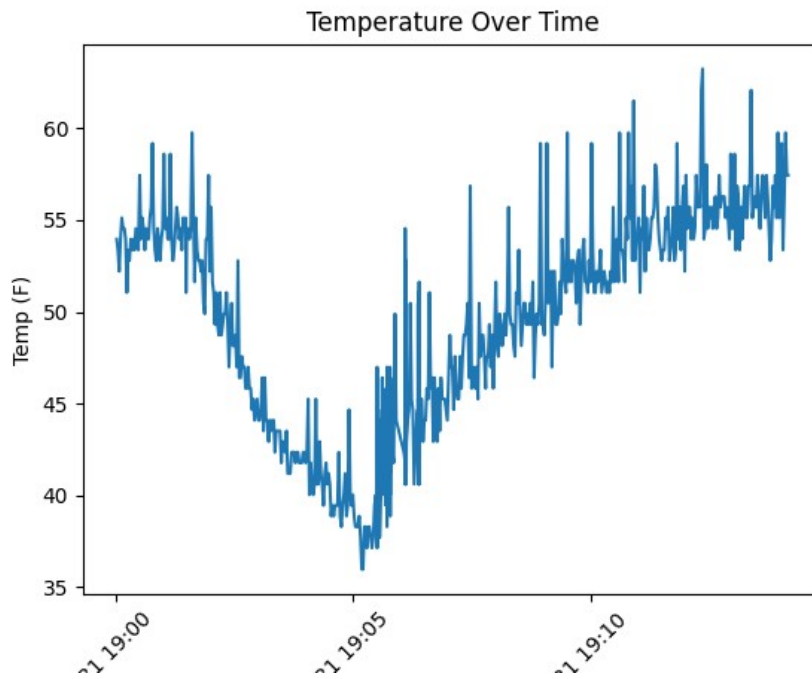
```

## Demo

Video showing working assembly:

<https://drive.google.com/file/d/1oolzOVZK3Xycx-uyM3KyB1P5aNfiinL/view?usp=sharing>

Figure 1 Shows the temperature graph, redone after the demo video was shot, to better show a continuous sampling without the bug I experienced initially. The temperature sensor is put in the fridge again, and it cools down, then recovers when I take it out of the fridge.



(Figure 1) Temperature over Time

## References

[https://electronoobs.com/eng\\_arduino\\_tut140.php](https://electronoobs.com/eng_arduino_tut140.php)