Practical PCB Design and Manufacture

Lab-16 Report: Differential or Single Ended signaling and ground noise

Objective / Purpose of the Lab:

- The lab's goal is to compare analog signal quality from a sensor using single-ended and differential pair measurements, assessing how each method captures the sensor's output amidst potential noise and interference. Single-ended measurement involves referencing the signal to a common ground, which can be prone to noise, while differential pair measurement uses two wires to capture the signal, reducing noise impact by comparing the difference between them. Additionally, the lab explores I2C bus signals, a two-wire communication protocol, to understand digital communication between devices, focusing on data timing and signal integrity.

Component Listing:

- Solderless Bread Board
- Arduino UNO
- TMP36 Temperature Sensor
- ADS1115 module
- Connecting Wires
- Function Generator
- Oscilloscope
- Digital Multimeter

Arduino Code:

```
#include <Adafruit_ADS1X15.h>
#include <Wire.h>
Adafruit_ADS1115 ads; // instantiates the object
float mV_per_ADU;
long n_operations = 100;
long iTimer0_usec;
long iDummy;
float Time_X_usec;
void setup()
{
        Serial.begin(1000000);
        delay(1000); //lets the serial port get initialized
        ads.begin(0x48); //default address with addr pin connected to gnd
        ads.setGain(GAIN_TWOTHIRDS); // 2/3x gain +/- 6.144V 1 bit = 3mV 0.1875mV
        mV_per_ADU = 0.1875; //for ADS1115
}
void loop()
{
        //Serial.println(ads.readADC_SingleEnded(0)*mV_per_ADU);
        iTimer0_usec = micros();
        for (int i = 1; i <= n_operations; i++)
        {
                iDummy = ads.readADC_SingleEnded(0);
        }
        Time_X_usec = (micros() - iTimer0_usec) / (n_operations * 1.0);
        Serial.print(" Xtime_usec: ");
        Serial.print(Time_X_usec, 3);
        Serial.print(" Frequency (Hz): ");
        Serial.print(1e6 / (Time_X_usec), 3);
        Serial.println();
}
```

The code given above is used to measure the execution time of the ADS1115 which is used to sample the measurements. After running the code, I was getting a sample rate of about 105 Samples/sec.

For averaging the ADS1115 readings, we have implemented the code given below.

```
#include <Wire.h>
#include <Adafruit_ADS1X15.h>

Adafruit_ADS1115 ads; /* Use this for the 16-bit version */
int iTime_ave_msec = 100; //averaging time per point
long iTime_stop_msec;
int iCounter1 = 0; //used in counting number of points
float V_diff_mV; //used for diff voltage
float V_SE_mV; // used for SE measurement

void setup(void)
{
        Serial.begin(2000000);
        // ADS1015 ADS1115
        // ------- -------
        // ads.setGain(GAIN_TWOTHIRDS); // 2/3x gain +/- 6.144V 1 bit = 3mV 0.1875mV (default)
        // ads.setGain(GAIN_ONE); // 1x gain +/- 4.096V 1 bit = 2mV 0.125mV
        // ads.setGain(GAIN_TWO); // 2x gain +/- 2.048V 1 bit = 1mV 0.0625mV
        ads.setGain(GAIN_FOUR); // 4x gain +/- 1.024V 1 bit = 0.5mV 0.03125mV
        // ads.setGain(GAIN_EIGHT); // 8x gain +/- 0.512V 1 bit = 0.25mV 0.015625mV
        // ads.setGain(GAIN_SIXTEEN); // 16x gain +/- 0.256V 1 bit = 0.125mV 0.0078125mV
        ads.begin();            //with the ADR pin set to gnd.
        ads.setDataRate(RATE_ADS1115_860SPS);
}

void loop(void)
{
        V_diff_mV = 0.0;
        V_SE_mV = 0.0;
        iCounter1 = 0;
        iTime_stop_msec = micros() / 1000 + iTime_ave_msec;
        while (micros() / 1000 <= iTime_stop_msec)
        {
                V_diff_mV = V_diff_mV + ads.readADC_Differential_0_1() * 0.0312;
                V_SE_mV = V_SE_mV + ads.readADC_SingleEnded(0) * 0.0312;
                iCounter1++;
        }

        V_diff_mV = V_diff_mV / iCounter1;
        V_SE_mV = V_SE_mV / iCounter1;
        Serial.print(720);
        Serial.print(", ");
        Serial.print(760);
        Serial.print(", ");
        Serial.print(V_diff_mV);
        Serial.print(", ");
        Serial.println(V_SE_mV);

}
```

This code given above uses the Adafruit ADS1115 library to measure and compare differential and single-ended (SE) voltages from an ADS1115 analog-to-digital converter (ADC). In the setup() function, it initializes serial communication at 2 Mbps, sets the ADC's gain, begins ADC operation, and sets its data rate. The loop() function repeatedly measures differential and SE voltages over a specified averaging time (iTime_ave_msec). It accumulates these voltages and counts the number of measurements taken within the time frame. After the time elapses, it calculates the average voltage for both differential and SE measurements by dividing the total accumulated voltage by the count of measurements. Finally, it prints these averages to the serial monitor, along with two static values (720 and 760), for further analysis or display.

Serial Plotter Shots with Analysis:

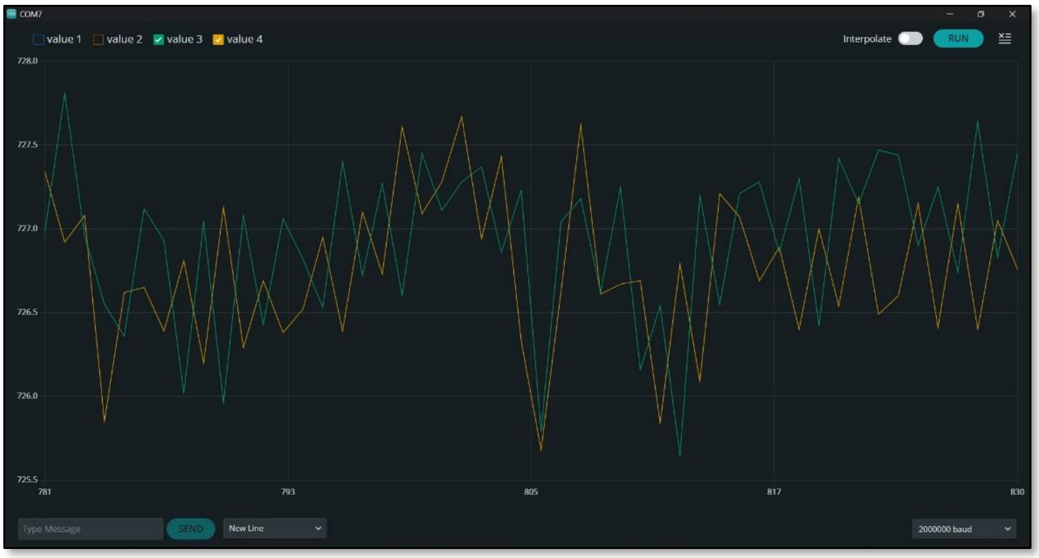

Fig – 1: SE vs diff without function generator

Both the single-ended and differential measurements are the same in the absence of noise in the ground channel, as the following illustration illustrates. The single-ended measurement is represented by the yellow channel, and the differential measurement is represented by the green channel.

About 20 degrees Celsius will be displayed on the temperature sensor. TMP36 has a sensitivity of 10mV/degC. Using a DMM, the voltage across the temperature sensor is measured and found to be 0.72V. Temperature and voltage are related by the following equation:

T[degC] = V [volts] * 100 [degC / V] - 50 [degC]

The voltage difference between the TMP36's single-ended measurement and differential measurement is insignificant when there is no current flowing through the ground return line. On the line, there is noise of roughly +/-5 mV. So, there is no benefit to differential measurement over single-ended when there is no noise in the return current channel.

When no function generator is connected, the voltage difference between single-ended and differential measurements should be negligible, indicating both methods are picking up similar levels of ambient electrical noise or inherent system voltage. This situation serves as a baseline, showing that without an external signal, any measured voltage primarily reflects the system's noise or residual voltage. It demonstrates the measurement system's sensitivity and accuracy in a no-signal state, highlighting its ability to minimize background interference and providing a reference point for evaluating performance when measuring actual signals.

A function generator is utilized to introduce noise into the return current path. The single-ended signal is coupled to a ground path that receives a 500mHz, 10V peak to peak signal. Between the TMP36 and ADS1115, this connection creates a periodic voltage drop between the two ends of the ground return channel.

The scatter plot shown below is for the output waveform when an input is provided using the function generator.

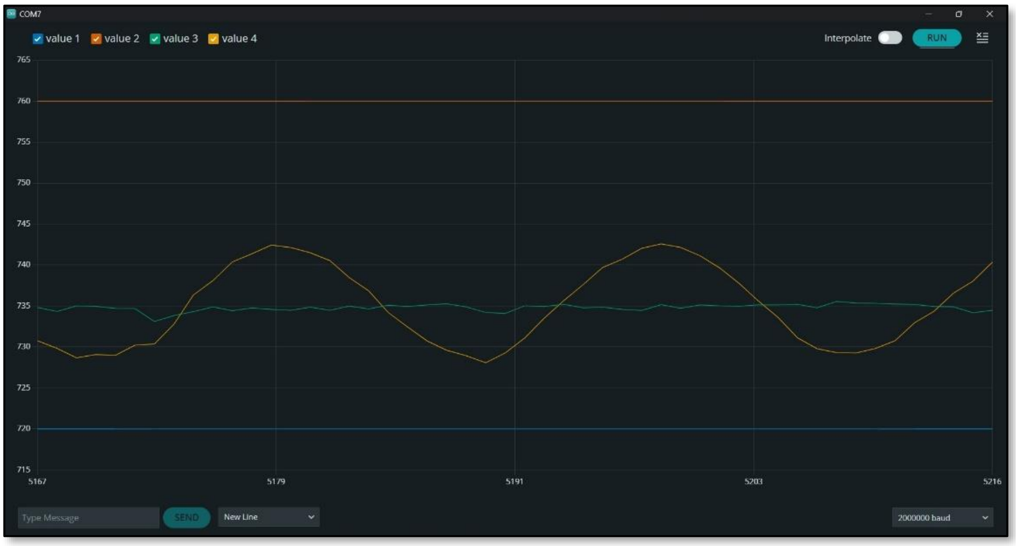

Fig – 2: SE vs diff with function generator

As predicted by the computations of the return path's resistance and current, the modulated voltage noise is between 10 and 15 mV peak to peak.

We are getting a square wave with 10V pk-pk at 500mHz (sine wave) from the signal generator. Since the comparator measures the differential signal, noise will be cancelled from both terminals, allowing us to observe no noise in the differential signal. However, because the single ended signal does not pass this feature, we can clearly see that the noise in the single ended signal (orange) is much higher than that of the differential signal.

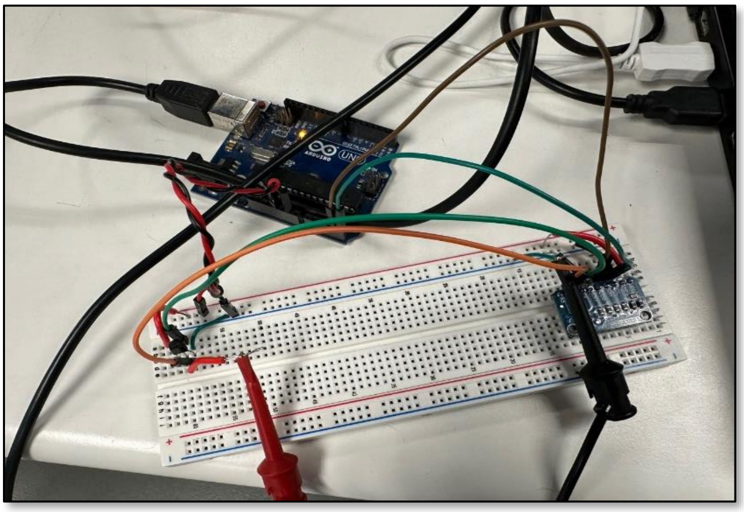

Fig – 3: Board Connections

Conclusion / Inference:

- Differential measurement neutralizes noise, ensuring clean signal output by comparing two closely related signals.
- Reduce crosstalk by routing differential lines far from other signal lines as possible in the layout.
- Ensure differential lines are closely paired in the layout to experience identical environmental noise, allowing effective noise cancellation.