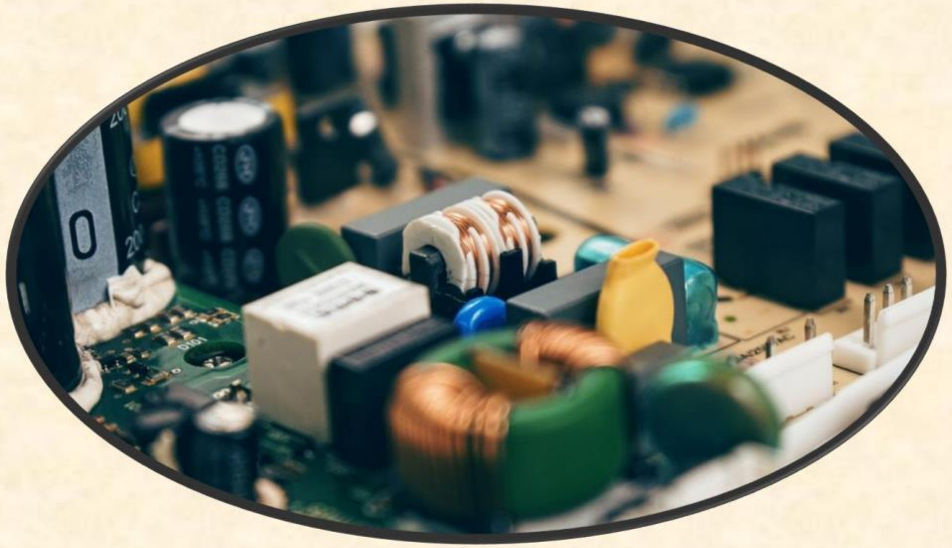
CIRCUIT DESIGN OF LOAD CELL TRANSDUCER



By

Amr Ashraf

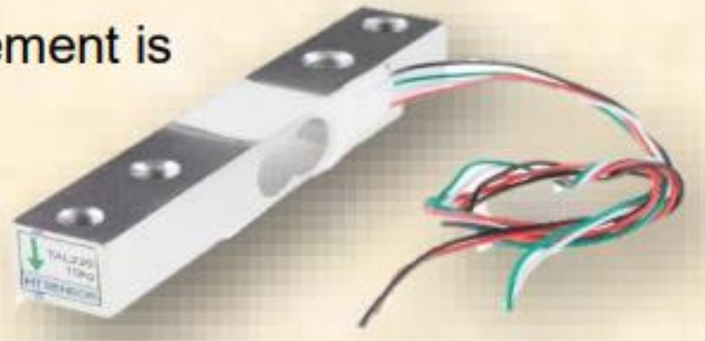
Marwan Ahmed

Khaled Ashraf

Ahmed Abdel Sattar

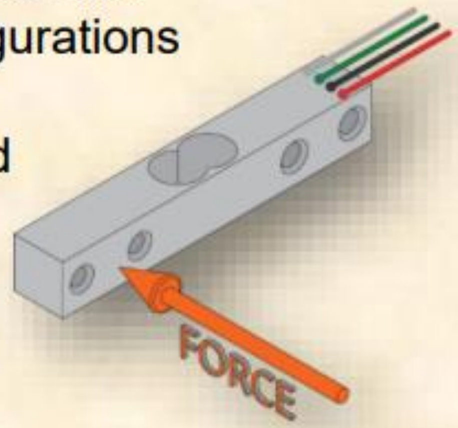
Introduction:

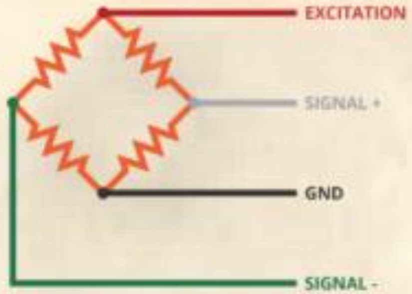
A load cell transducer is a sensor or transducer widely used to convert mechanical forces or loads into electrical signals, playing a crucial role in industrial, manufacturing, and scientific fields where precise force measurement is essential. Operating based on the principle of strain gauge technology, load cells incorporate one or more strain gauges, typically arranged in a Wheatstone bridge configuration, to measure the strain induced by a load. The electrical signal produced by the strain gauges is then converted into a proportional force or weight measurement.



Output voltage:

The output voltage of a load cell transducer is typically determined by the strain-induced changes in resistance of the strain gauges within the load cell. Load cells often use Wheatstone bridge configurations of strain gauges, and the output voltage can be calculated based on the bridge's imbalance caused by the applied force.





The formula to calculate the output voltage (V_{out}) of a load cell in a Wheatstone bridge configuration is:

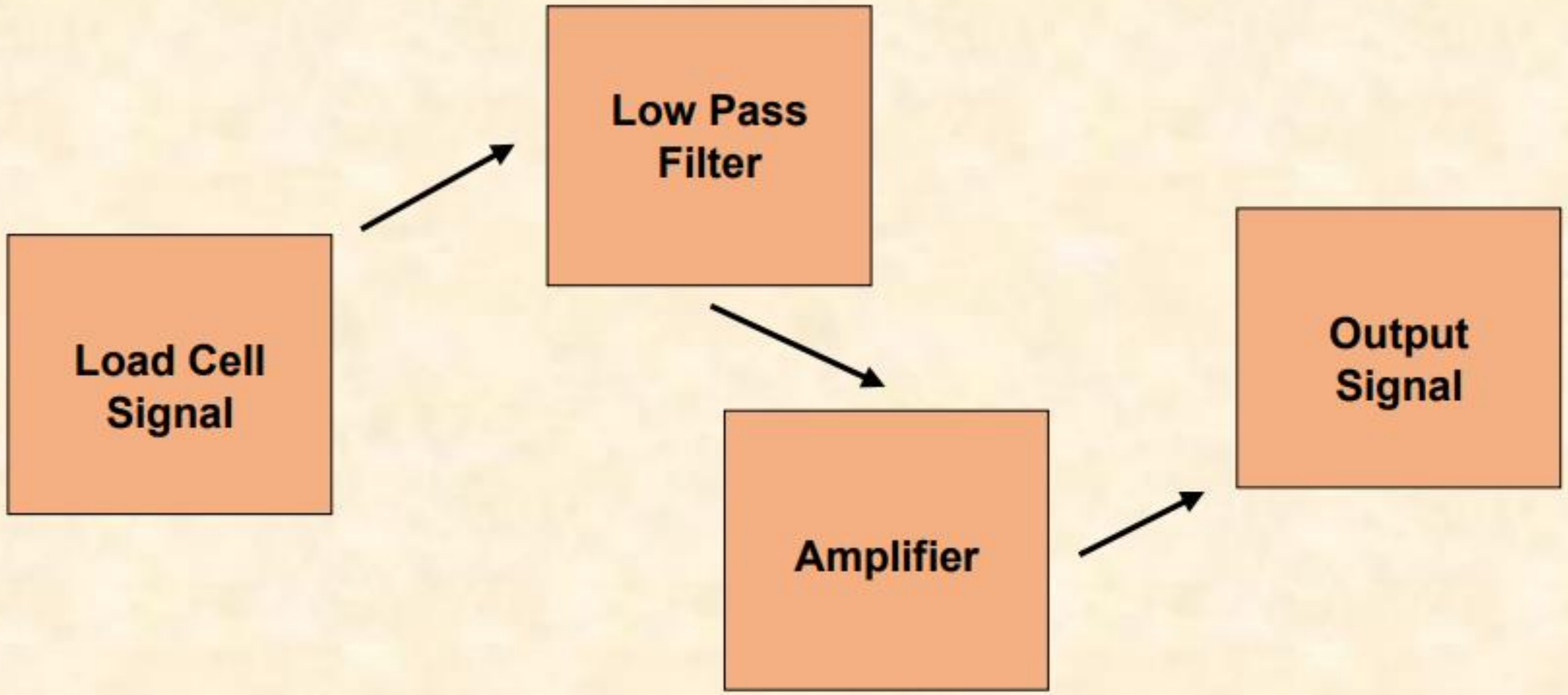
$$V_{out} = V_{exc} * G * \left(\frac{R}{\Delta R}\right)$$

Types of Load Cells:

Understanding the unique features and applications of each type of load cell is essential for selecting the most suitable one for a given industrial or scientific task. These devices contribute significantly to processes that require accurate and reliable force and weight measurements, ensuring efficiency and safety across various industries.

of

Block Diagram:

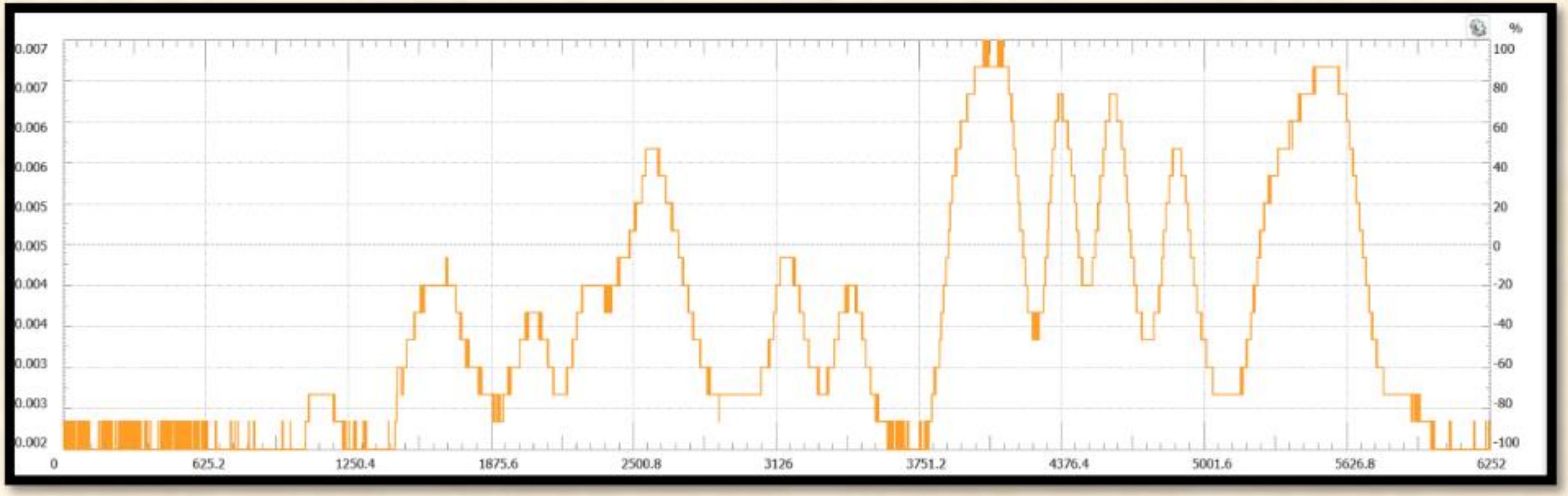


low-pass filter: to allow low-frequency signals pass through while attenuating higher frequencies.

Amplifier: Use to amplify the signal from the Load Cell sensor. This amplifier provides high input impedance, which prevents loading of the sensor and ensures accurate signal transfer.

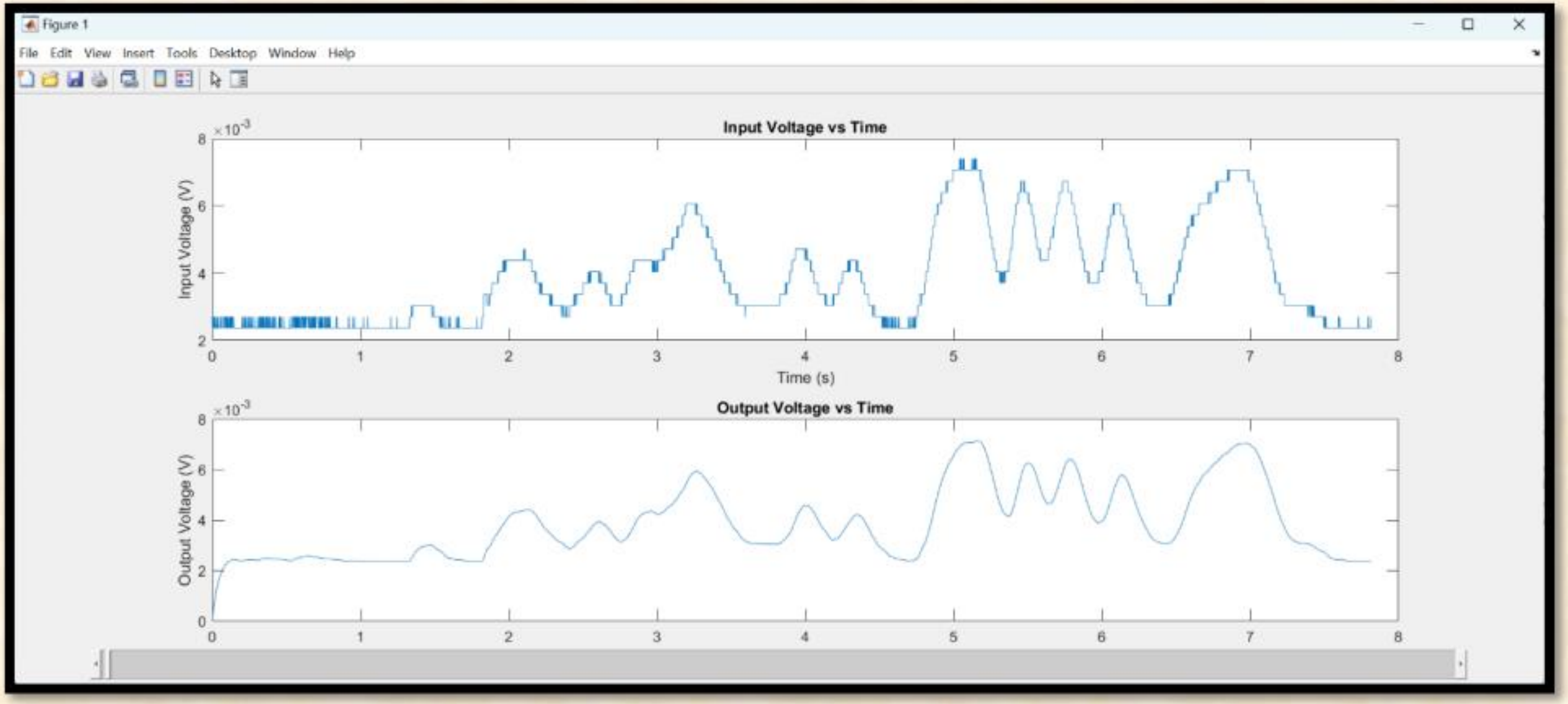
Export Data:

Export the acquired raw signal data (with headers and without headers) for further analysis or processing in external tools or software like LTspice.



Choose the filter:

Our goal is to allow low-frequency components to pass through while attenuating or blocking high-frequency components, and we need to amplify the signal. Therefore, our choice will be an active low-pass filter, and we will design it. we used a simple mat-lap script to see the output signal in terms of different omegas to choose the smoothest one which is the same result 25 rad/sec. we can change the omega from the bottom scroll bar.



The circuit: we then calculated the circuit components in terms of our available components in the lab as following

1- we chose R_{i1} as 33k

2- we choose R_{f1} as 33k

3- we choose the omega as 25 rad/sec

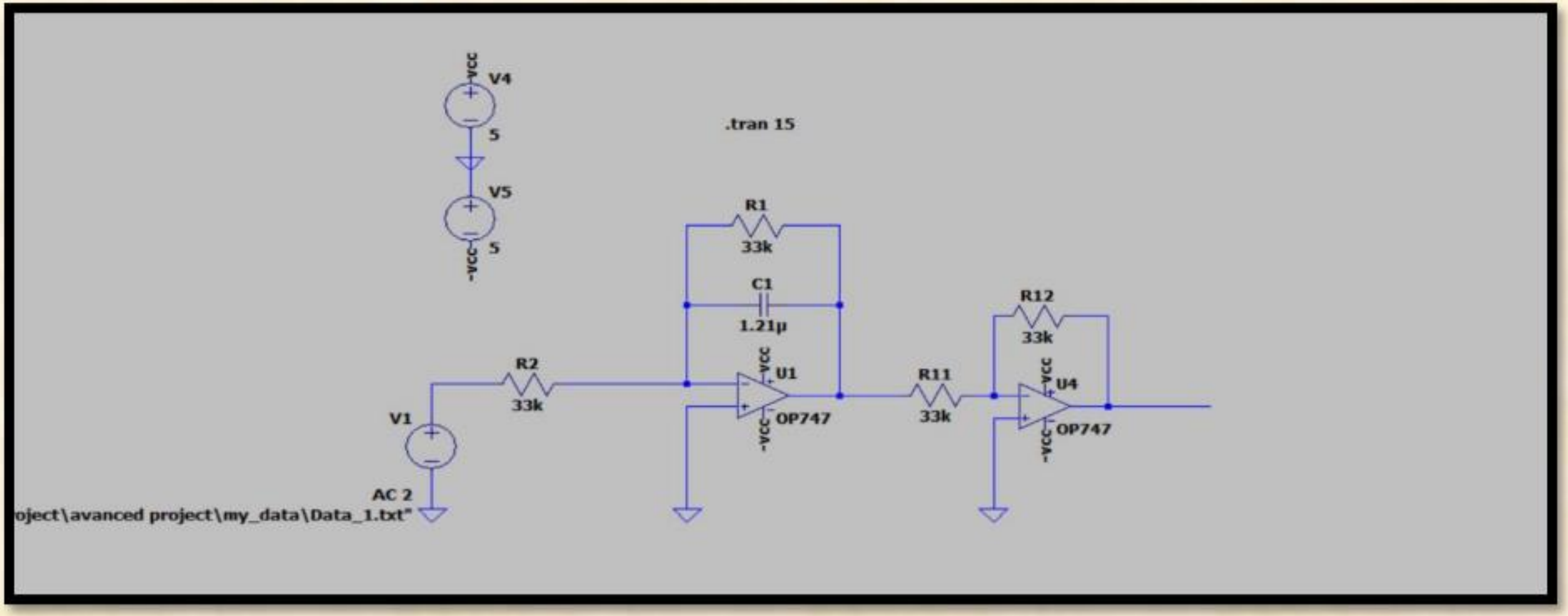
4- we calculated the $C_f = 1 / (W^*R_{f1})$

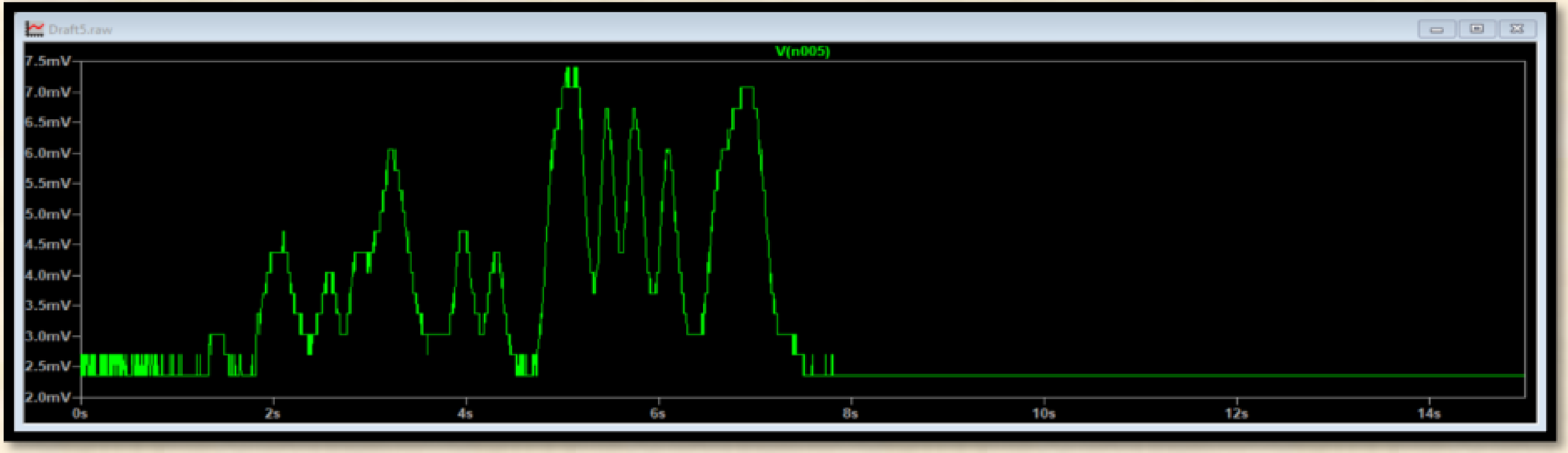
5- we choose the gain as 150

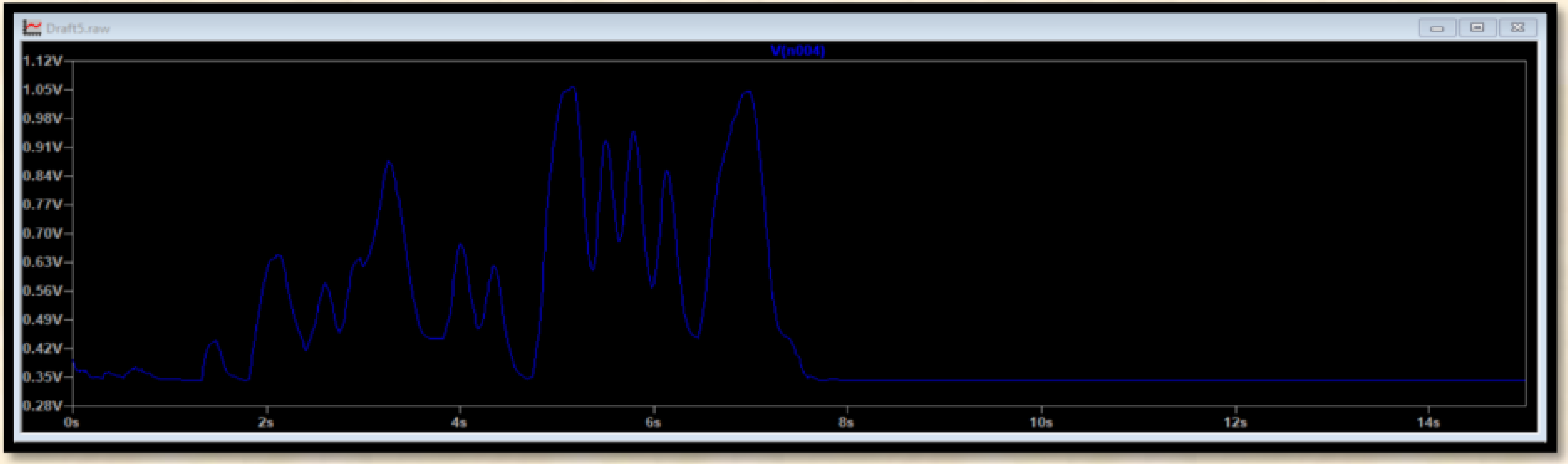
6- we choose the R_{i2} = 33k

7- we choose R_{f2} = 4.95M to make a 150 gain

LTspice: Now, we begin designing the circuit in LTspice, importing the raw signal, and measuring the output. The green signal represents the raw input, while the blue signal signifies the filtered output. Notably, the filtered signal exhibits increased amplitude, and a significant reduction in noise is observed.

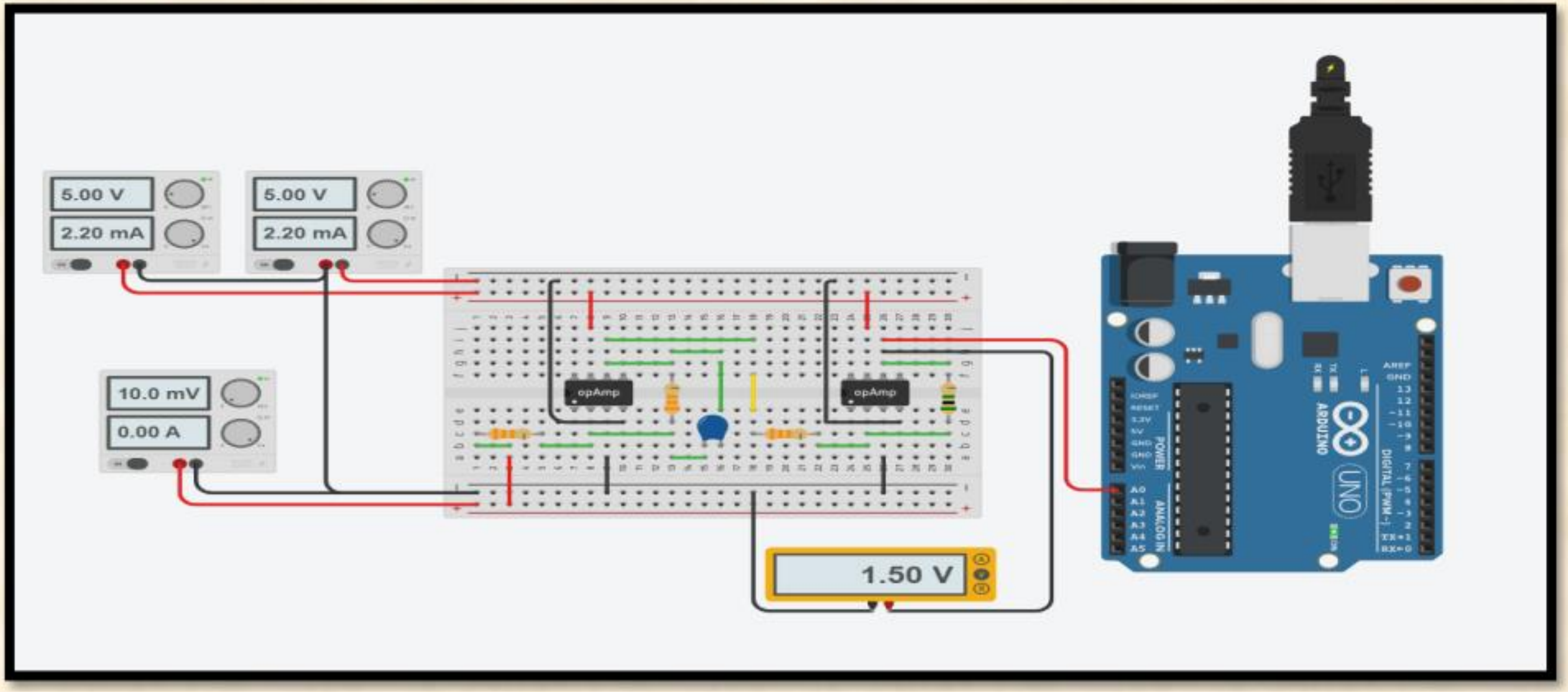


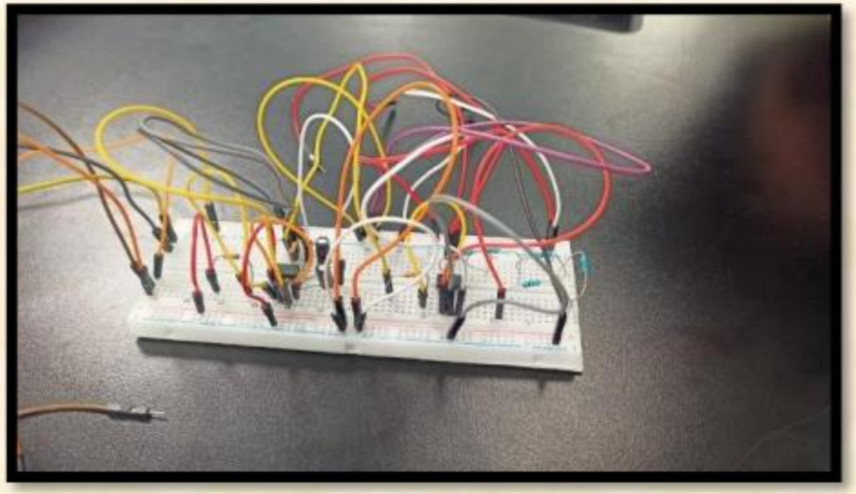
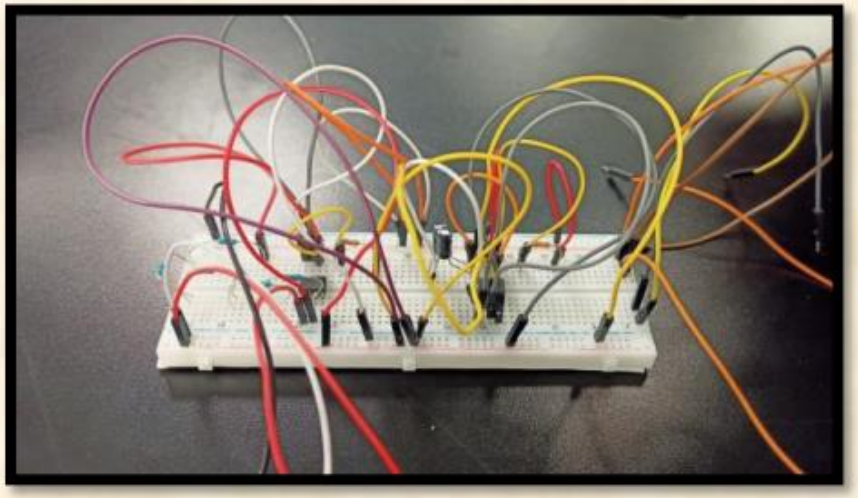
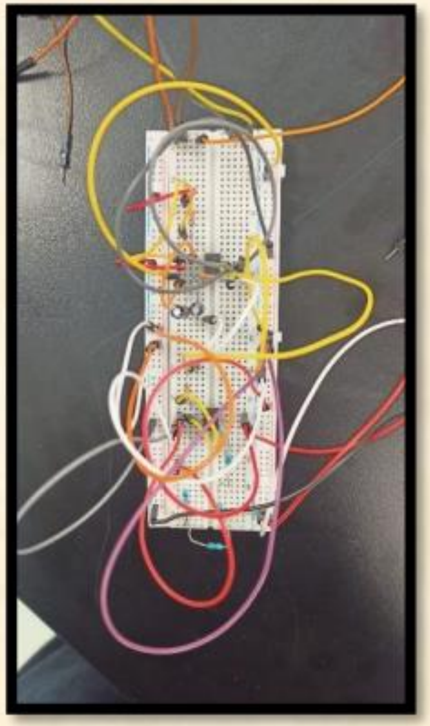
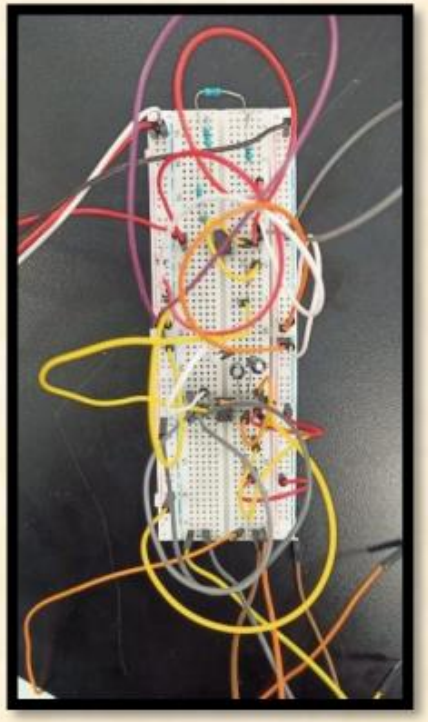

Draft5.raw
V(n005)
7.5mV
7.0mV
6.5mV
6.0mV
5.5mV
5.0mV
4.5mV
4.0mV
3.5mV
3.0mV
2.5mV
2.0mV
0s
2s
4s
6s
8s
10s
12s
14s


Draft5.raw
V(n004)
1.12V
1.05V
0.98V
0.91V
0.84V
0.77V
0.70V
0.63V
0.56V
0.49V
0.42V
0.35V
0.28V
0s
2s
4s
6s
8s
10s
12s
14s

Build the circuit:

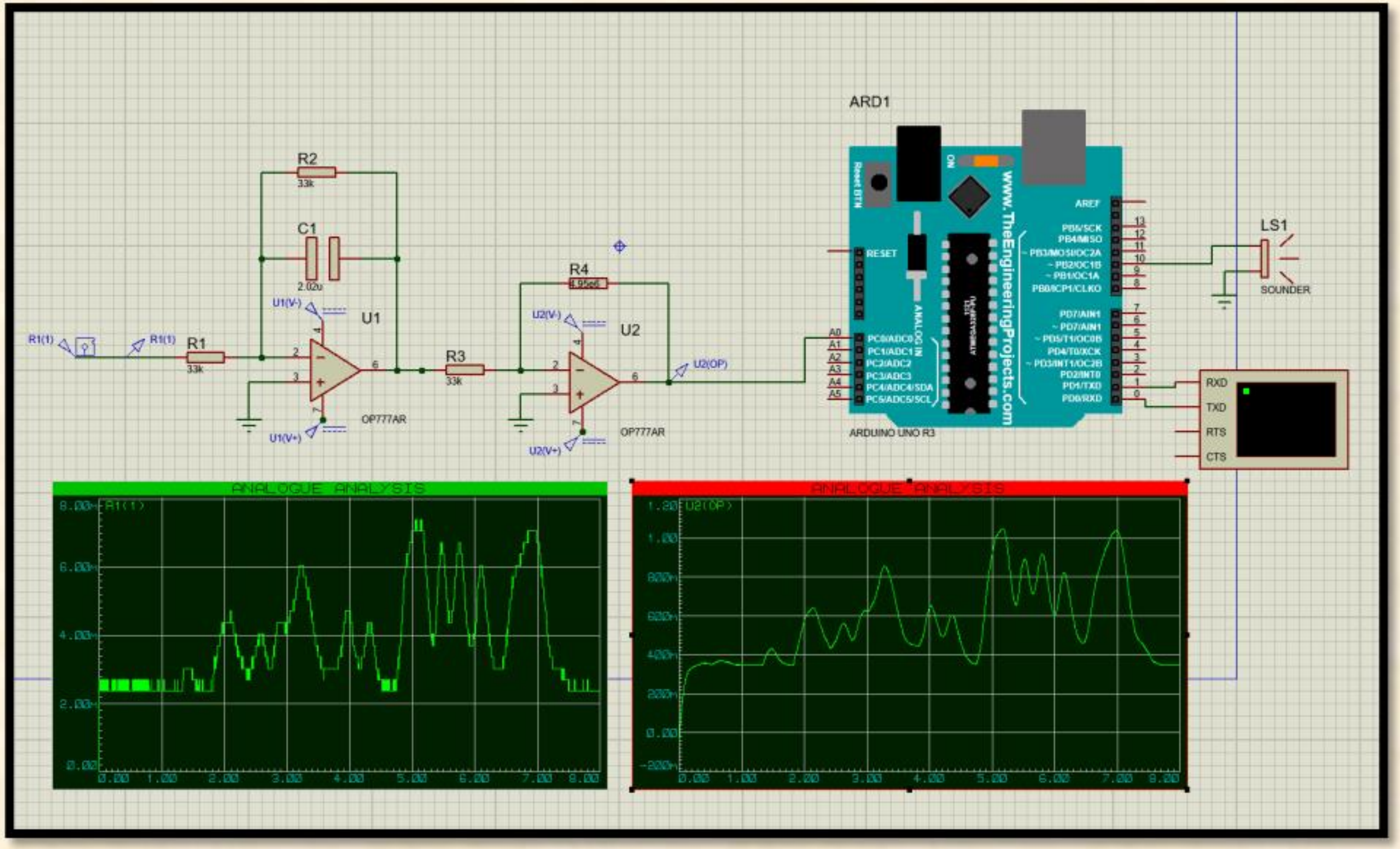
We first designed it on tinker cad software so it becomes easy for us to implement it with real hardware and reduce the errors as possible, we also tried our best to make it look clean and prevent wire crossing each other, we also used to know color for +Vcc or any positive signal and -Vcc or any negative signal as red and black colors



Arduino:

After that we downloaded a new software Proteus 8 so we can simulate an Arduino, it wasn't easy because it wasn't built in it so we added it from external source as new library and we designed the circuit and then add the output signal to the A0 port and then we read the analog port in the code to measure the weight based on the input signal so we made our load cell works as a weight measure



Virtual Terminal

```
Weight: 0.00 grams
Weight: 0.00 grams
Weight: 0.00 grams
Weight: 0.00 grams
Weight: 0.00 grams
Weight: 0.00 grams
Weight: 649.40 grams
Weight: 1170.74 grams
Weight: 1170.74 grams
Weight: 551.65 grams
Weight: 30.30 grams
Weight: 95.47 grams
Weight: 616.81 grams
Weight: 584.23 grams
Weight: 193.22 grams
Weight: 877.48 grams
Weight: 1170.74 grams
Weight: 1398.83 grams
Weight: 2050.50 grams
Weight: 2702.18 grams
Weight: 2180.84 grams
Weight: 1203.32 grams
Weight: 356.14 grams
Weight: 62.89 grams
Weight: 0.00 grams
Weight: 258.39 grams
Weight: 1235.91 grams
Weight: 1007.82 grams
Weight: 290.97 grams
Weight: 681.98 grams
Weight: 942.65 grams
```

Thank you