ELECTRONICS WORKSHOP 2

Spring 2023
UG2

Non-linear Energy Operator

HARDWARE REALIZATION AND MORE

Brahad Kokad - 2021112004

video presentation link: https://tinyurl.com/34hrhpvy

26/04/2023

MOTIVATION

Many naturally occurring signals contain erratic spikes and sudden changes which can be triggers or effect of something very significant, for example in brainwave and ECG signals. Detecting these spikes and their properties is thus very essential for modern medicinal science.

Systems that can detect these in real time with good accuracy are desired. NEOs (non-linear energy operators) have a very important property that they can detect spikes by virtue of their mathematical form itself.
Energy operators basically measure the cross energy between a signal and its derivatives. A mathematical operator that is used to compute the energy of a signal in a given frequency band.

Other than this energy operators have applications in a variety of fields such as speech processing and image processing.
Due to this wide range of possible applications, making an electronic circuit to realize the NEO is extremely favourable.

Project flow and stages of implementation

- Literary study.
- Block design of NEO with operators involved.
- Building subcircuits for operators.
- Building the complete circuit using these subcircuits.
- Verification and testing: providing different input signals and analyzing the outputs.

LITERARY SURVEY

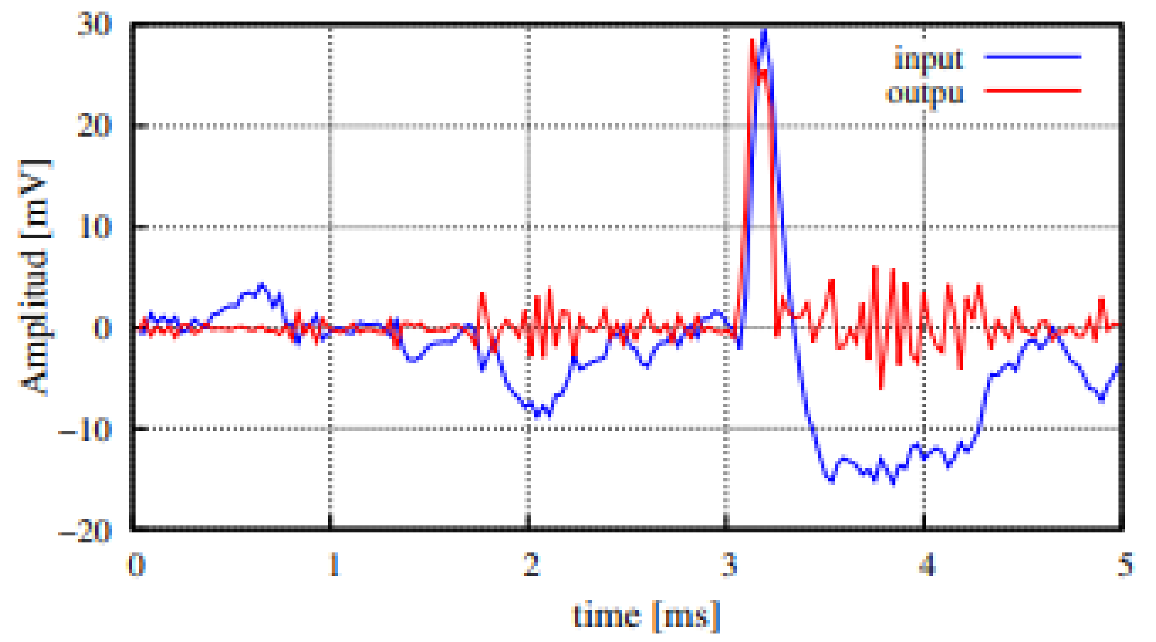


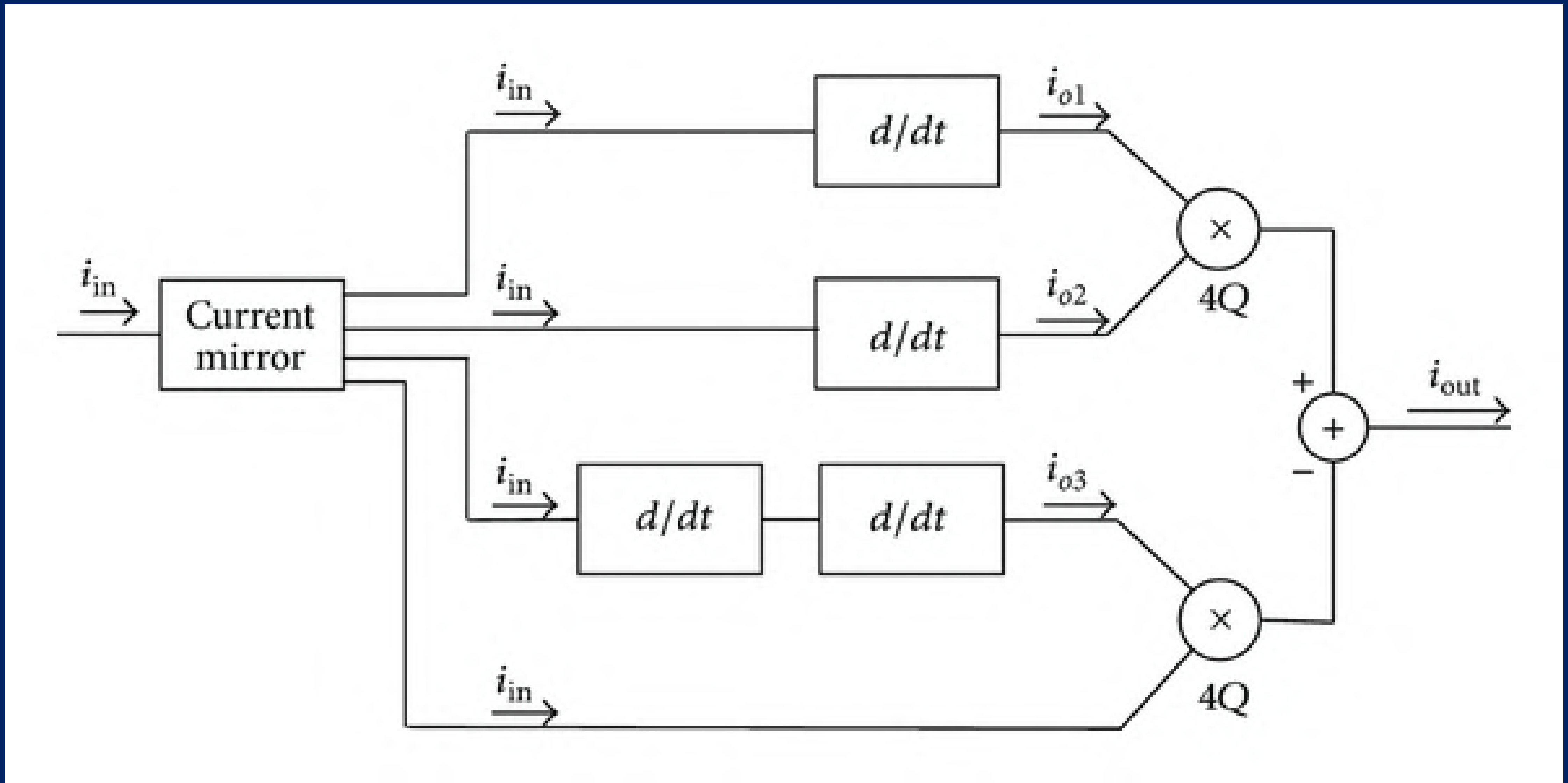
Fig. 7. Response of the circuit to real human neural signal. The output signal was scaled so that its maximum coincides with the input maximum.

NEO and EOs in general were completely new to me and I read several research papers to understand what they are supposed to be. In conclusion, my understanding is that EOs are essentially mathematical operations done on a signal and can be defined by a general equation for kth order. NEO is when k = 2.

Implementing this on hardware then essentially boils down to manipulating the mathematical equation into standard operations namely differentiation, multiplication and addition.

The adjacent figure is from one of the papers I've read and it shows what lies in the core of our idea. We want to emphasize peaks (occurrences of changes in signal that are sudden *and* significantly large), and we also want our output to be unaffected by smaller and slower changes, and to be noise resistant enough.

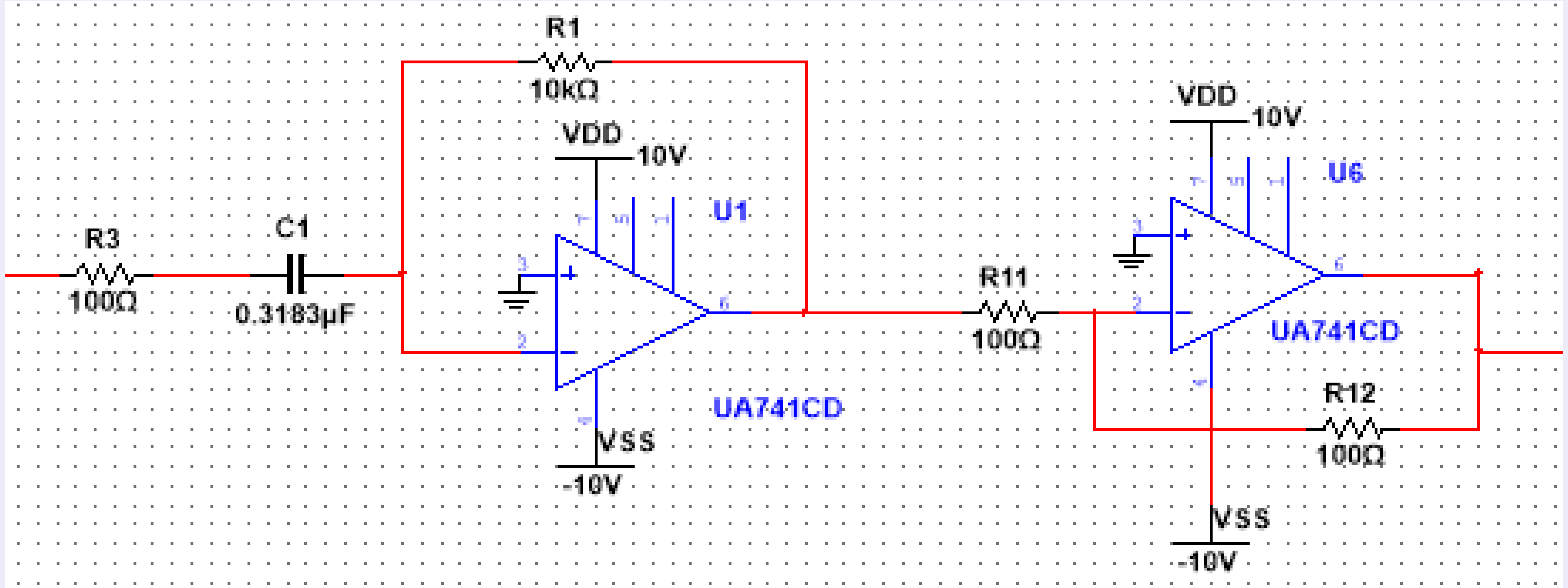
The block diagram ideation of this is given in the next slide and successive slides have a breakdown of implementing each individual block.



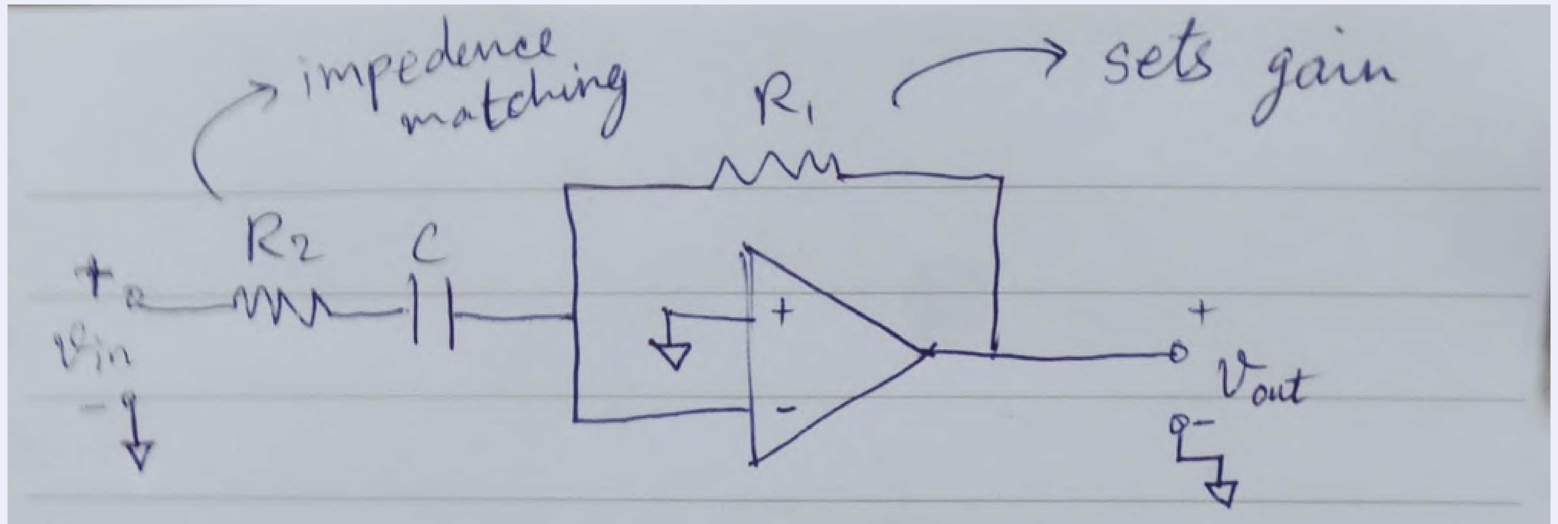
BLOCK DIAGRAM MODEL

DIFFERENTIAOR BLOCK

This block is implemented using an op-amp in a feedback loop system. The differentiator in itself gives out an inverting output, this has been corrected by cascading an op-amp based active inverter. This has been preferred over the usual CMOS based inverter so as to have the flexibility of controlling output gain levels.



Working of Differentiator



$$V_{out} = -RC \times \frac{dV_{in}}{dt}$$

$$\underbrace{i_c = i_{R_1}}_{} = \frac{V_- - V_{out}}{R_1} = \frac{-V_{out}}{R_1}$$

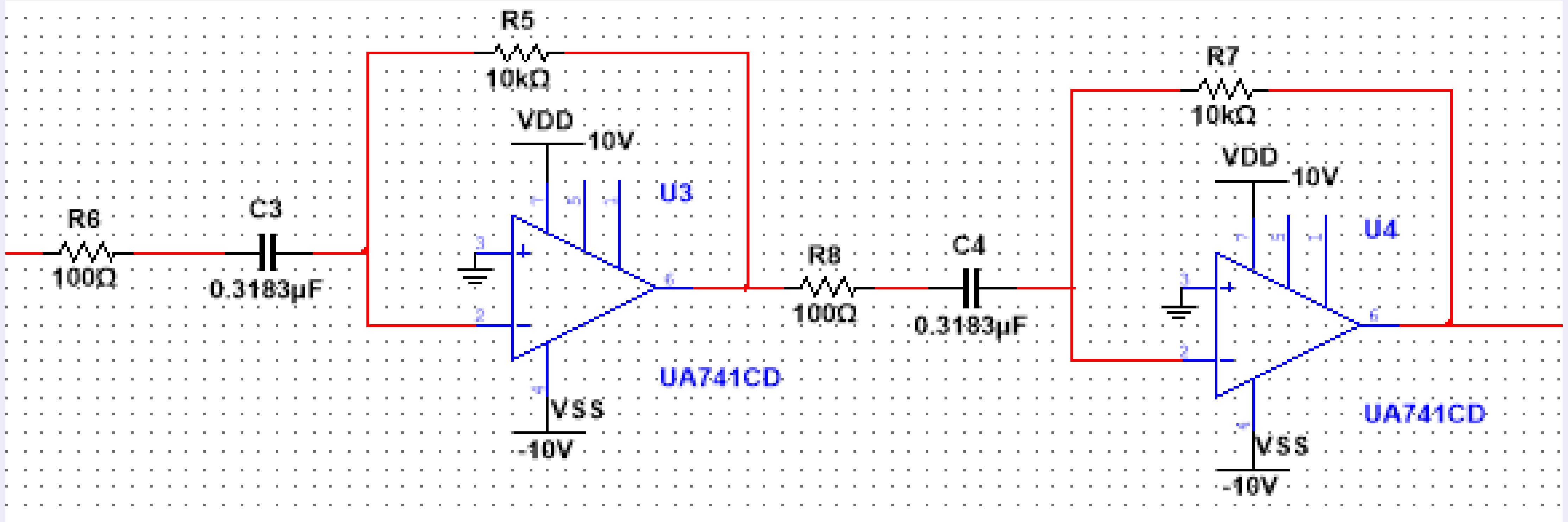
current through op-amp terminal is zero

$$\therefore V_{out} = -R_1 C \times \frac{dV_c}{dt}$$

$$V_{out} = -R_1 C \times \frac{dV_{in}}{dt}$$

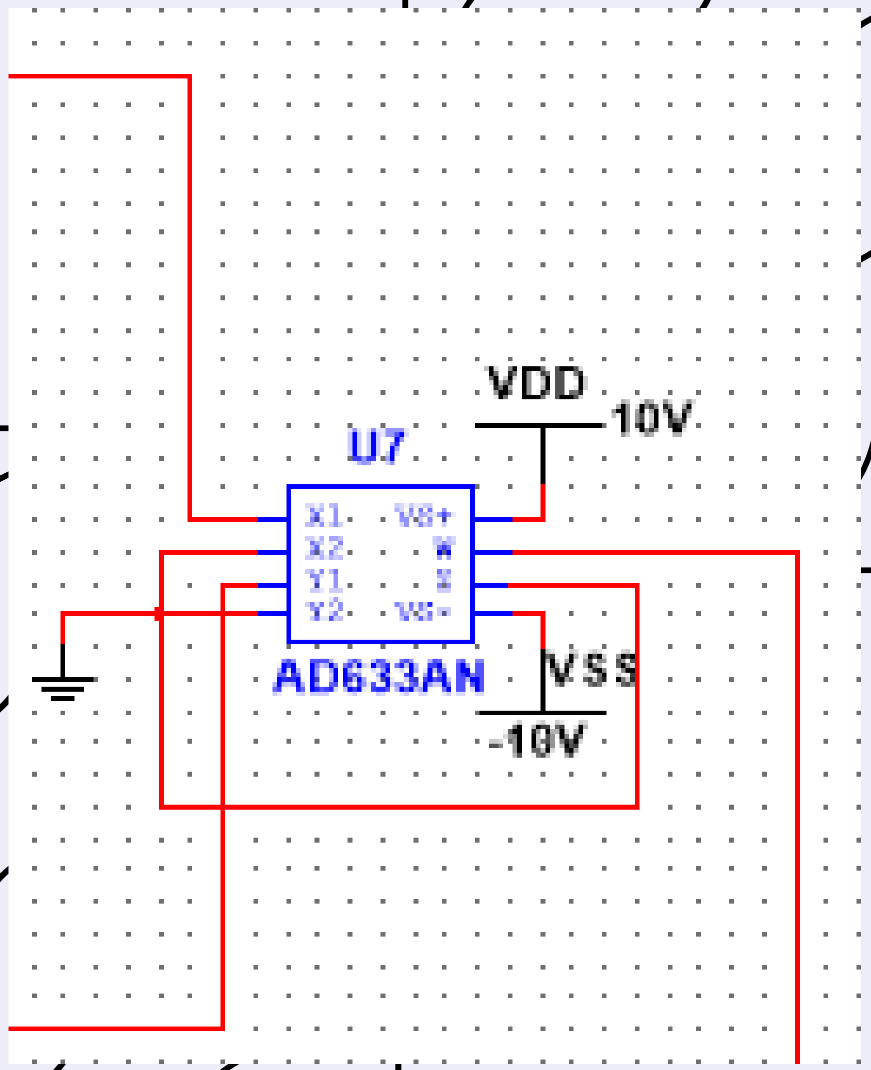
DOUBLE DIFFERENTIATOR BLOCK

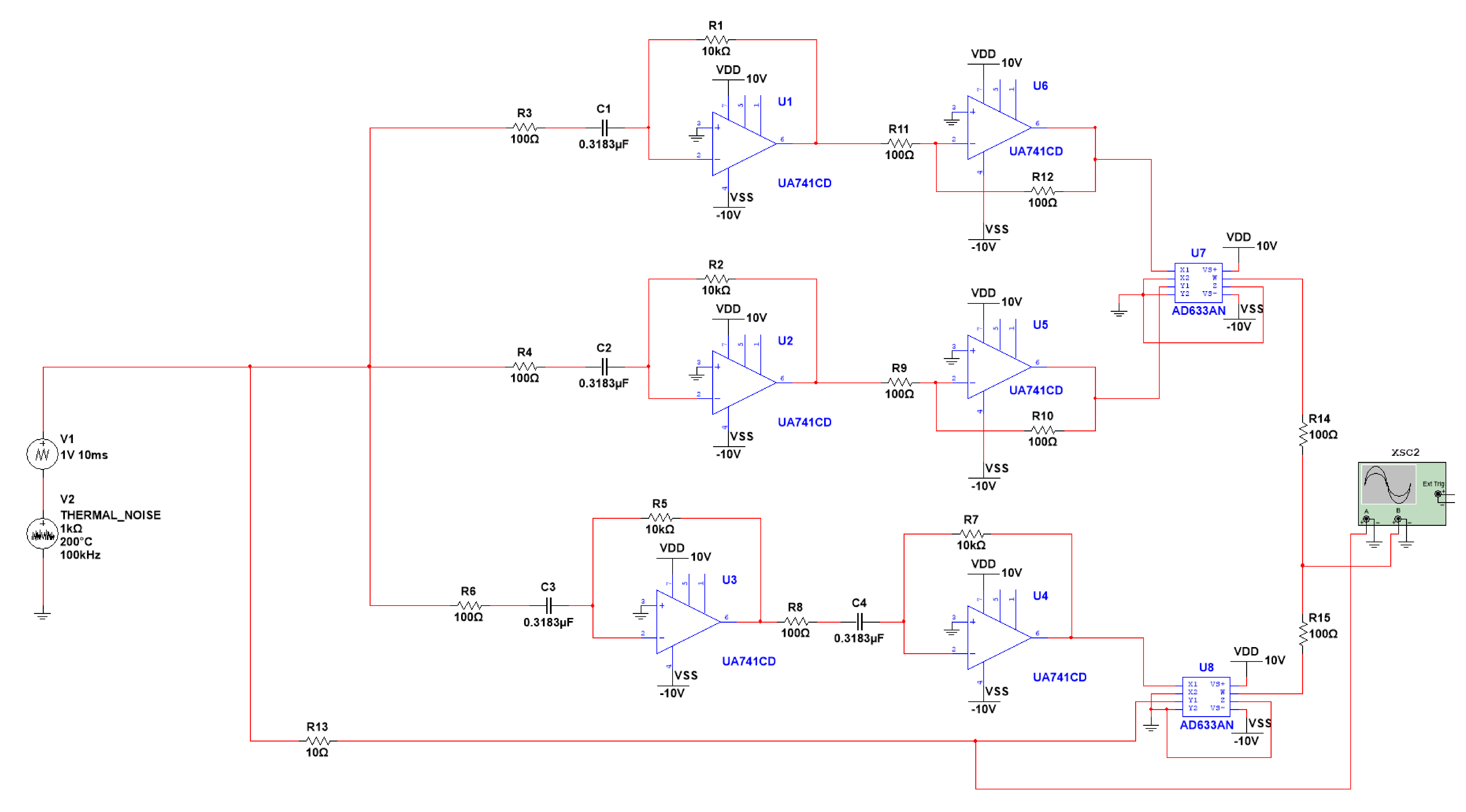
This is essentially two differentiator blocks cascaded.



MULTIPLIER BLOCK

Multiplication of signals is done by using the AD633 IC



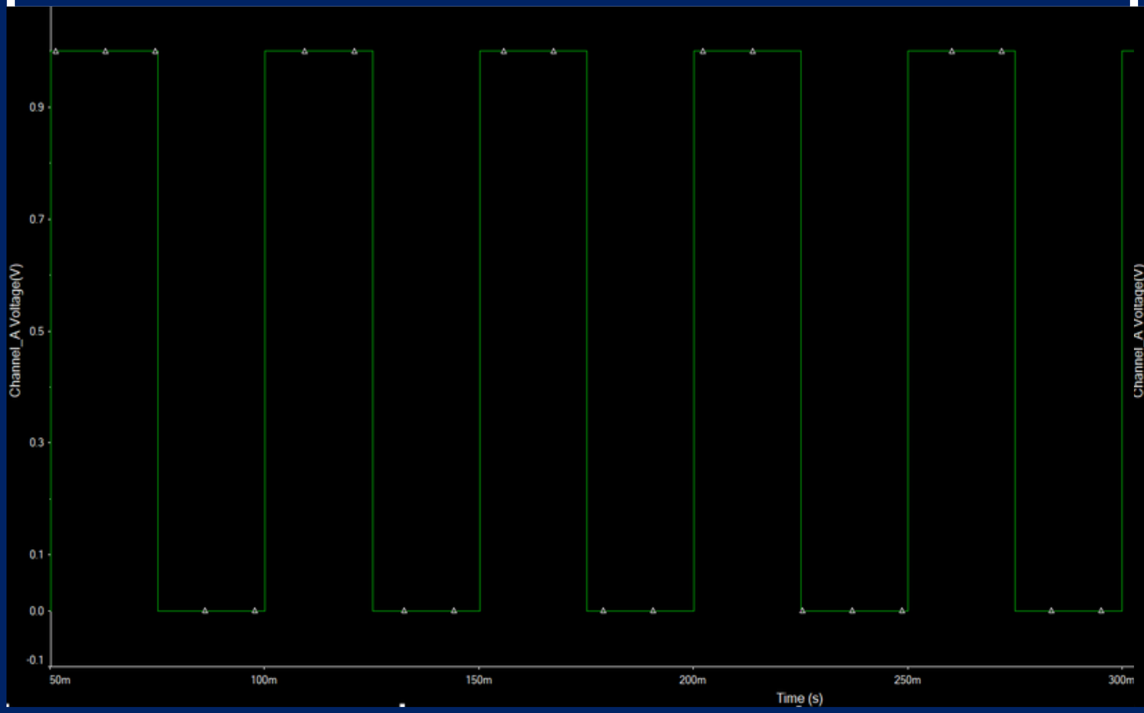


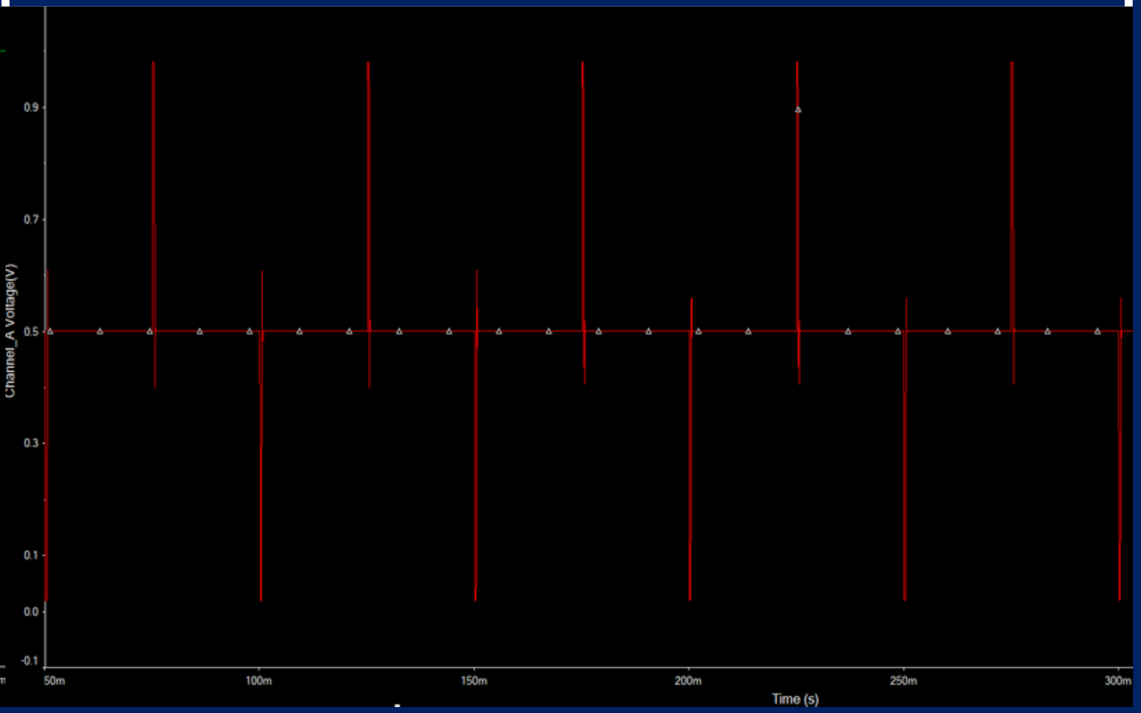
R1
10kΩ
VDD
10V
U1
R3
100Ω
C1
0.3183µF
UA741CD
VSS
-10V
VDD
10V
U6
R11
100Ω
UA741CD
R12
100Ω
VSS
-10V
VDD
10V
U7
AD633AN
VSS
-10V
R2
10kΩ
VDD
10V
U2
R4
100Ω
C2
0.3183µF
UA741CD
VSS
-10V
VDD
10V
U5
R9
100Ω
UA741CD
R10
100Ω
VSS
-10V
R14
100Ω
XSC2
Ext Trig
V1
1V 10ms
V2
THERMAL_NOISE
1kΩ
200°C
100kHz
R5
10kΩ
VDD
10V
U3
R6
100Ω
C3
0.3183µF
UA741CD
VSS
-10V
R8
100Ω
C4
0.3183µF
R7
10kΩ
VDD
10V
U4
UA741CD
VSS
-10V
R15
100Ω
VDD
10V
U8
AD633AN
VSS
-10V
R13
10Ω

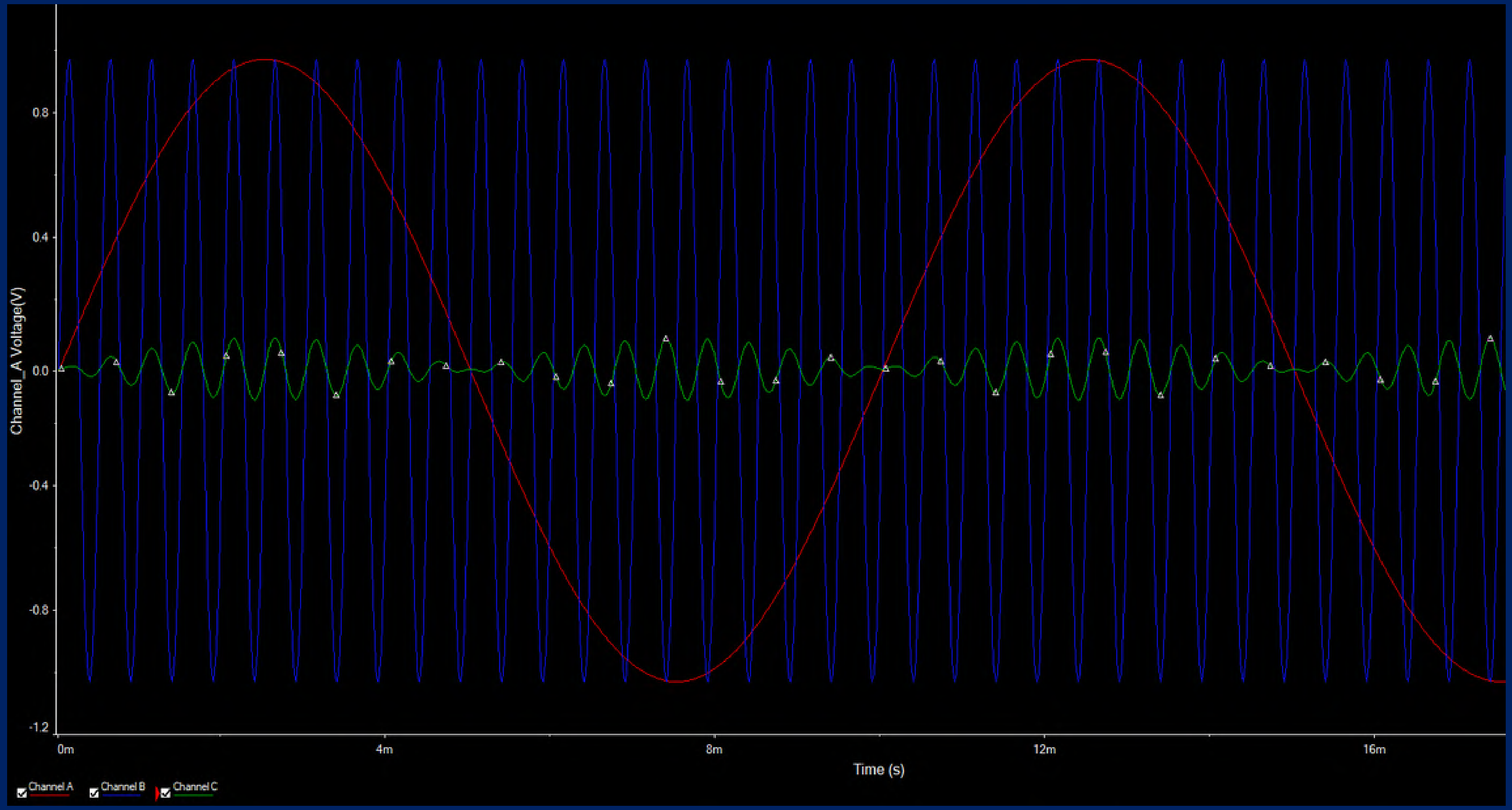
Simulation Results

Differentiator (inverted)

INPUT	OUTPUT



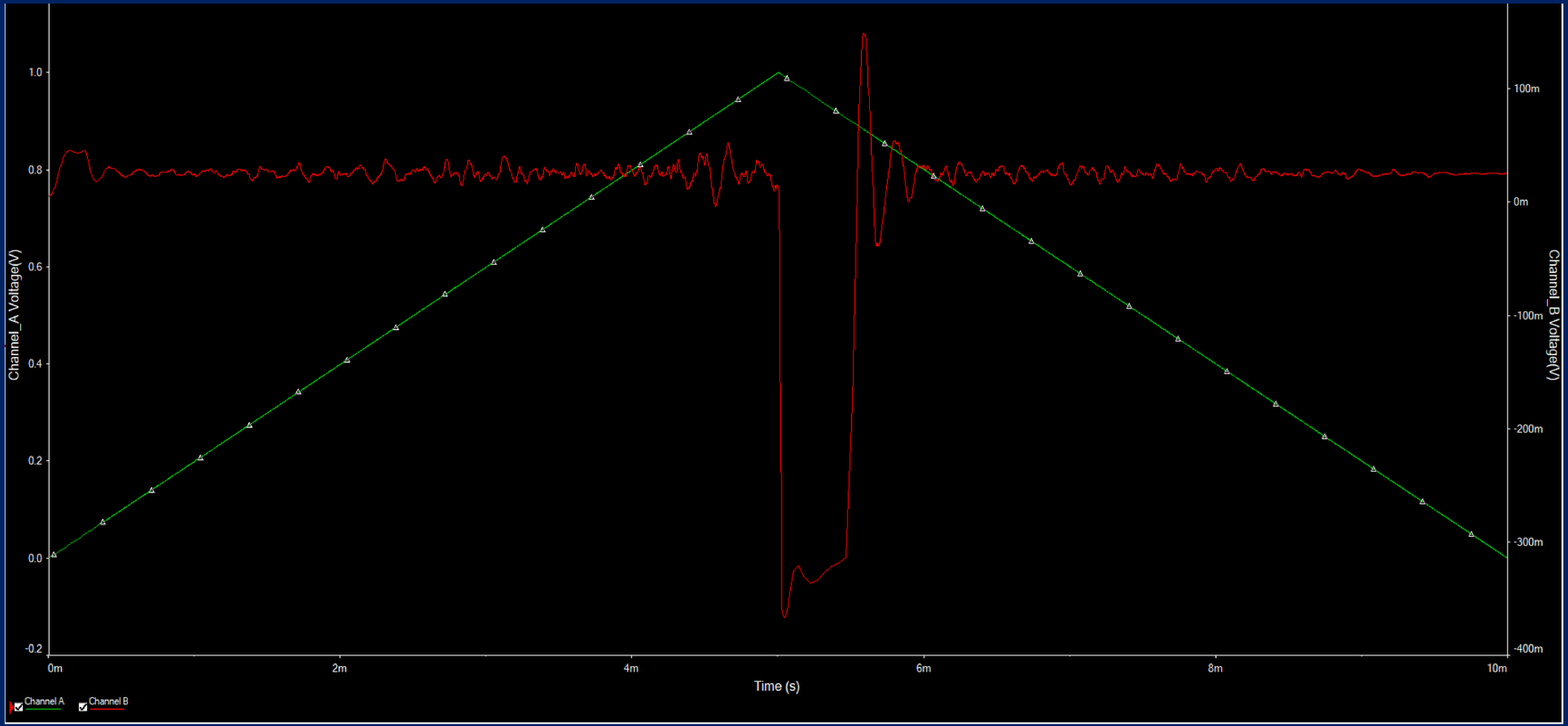




Inputs: red and blue, output: green

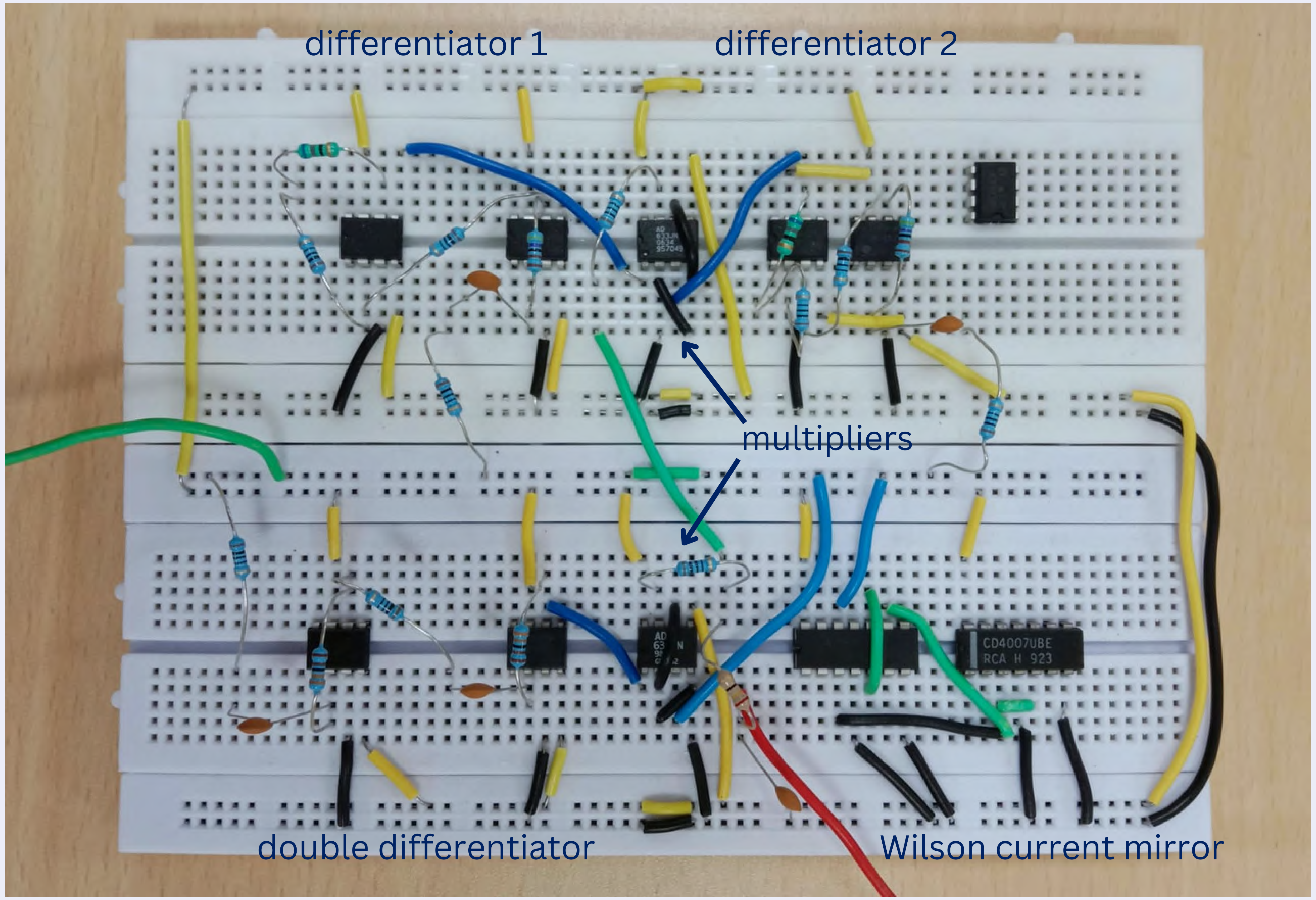
Multiplier

Final I/O



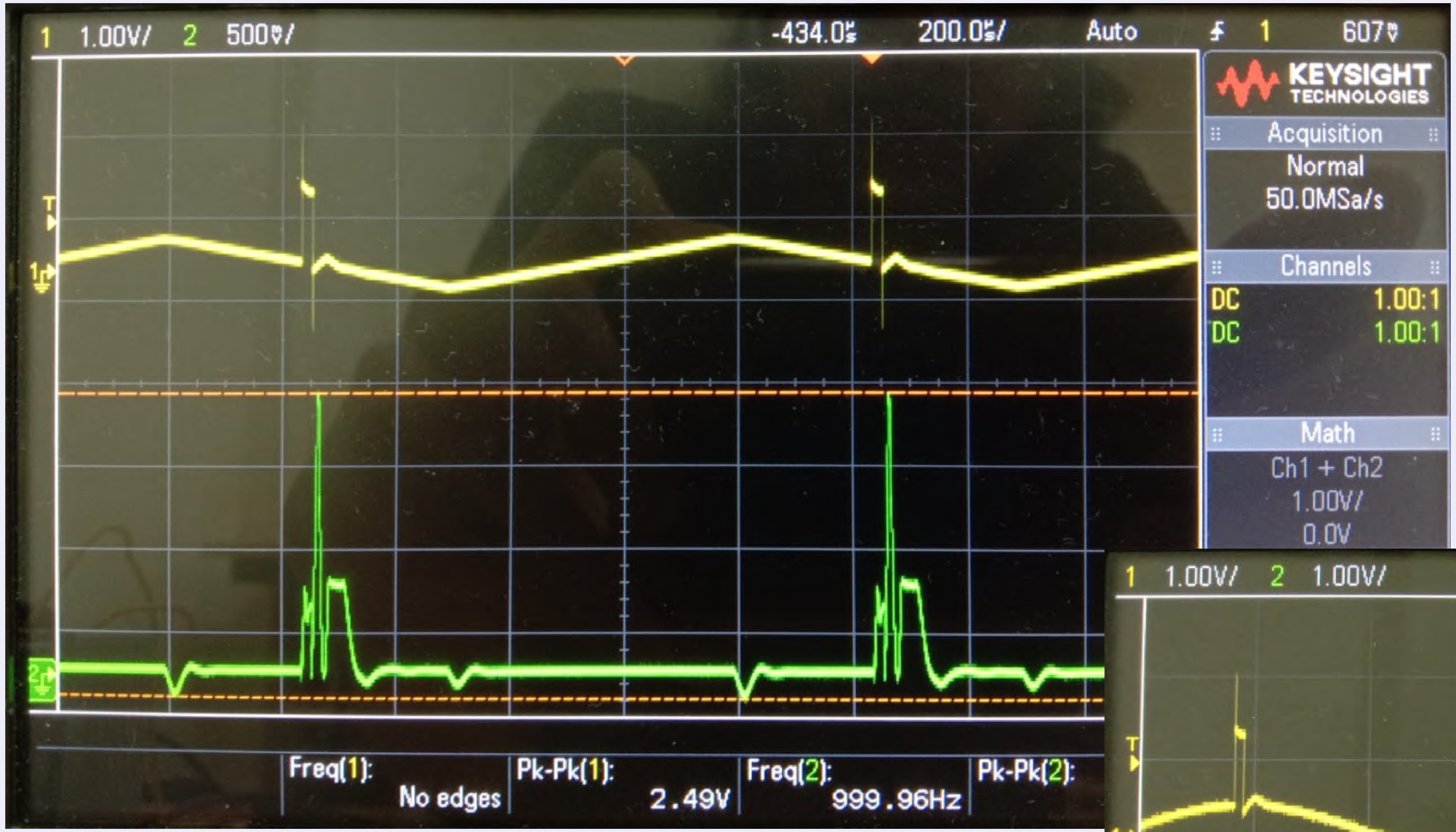
Input: green, output: red

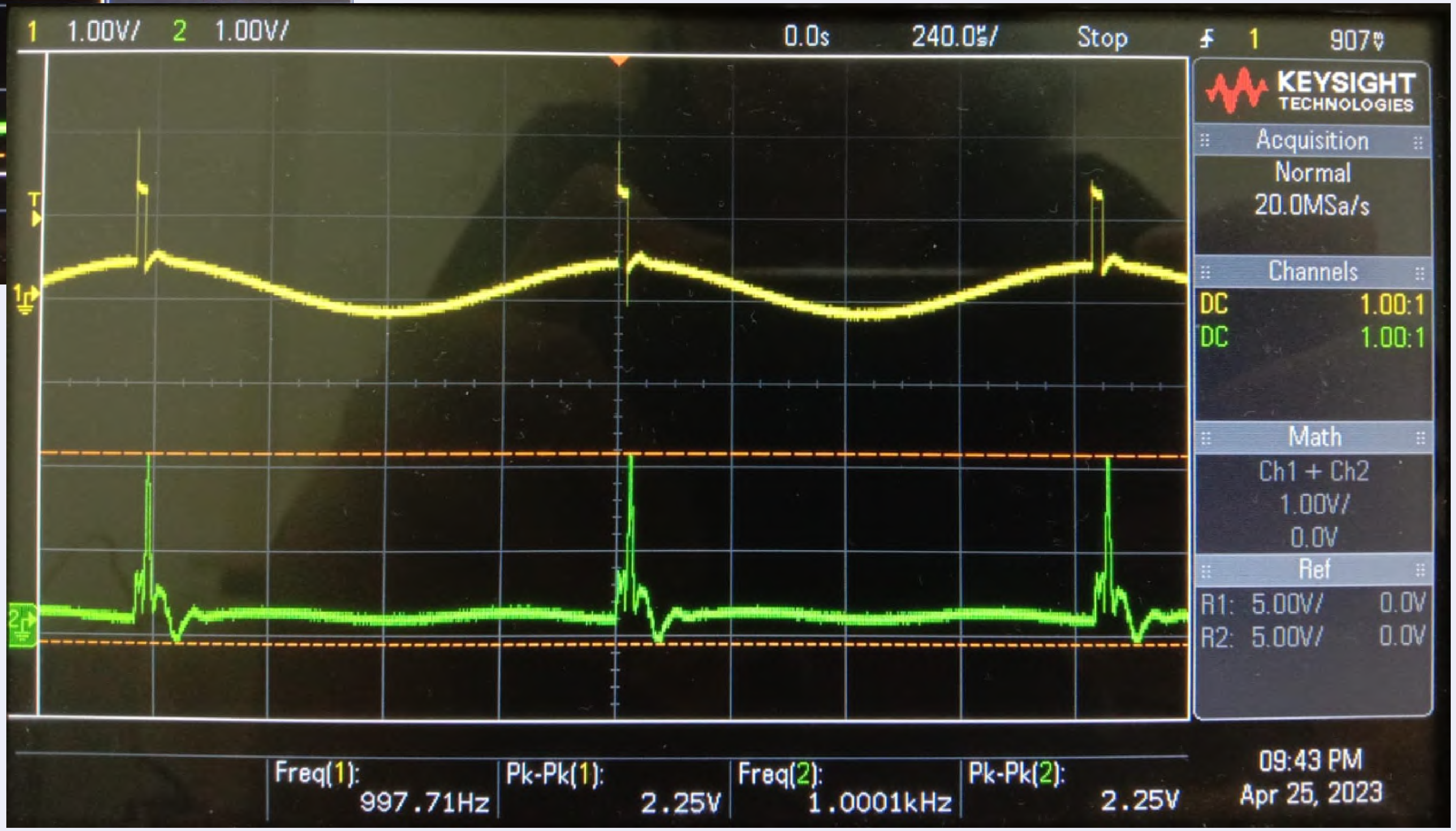
BREADBOARD IMPLEMENTATION



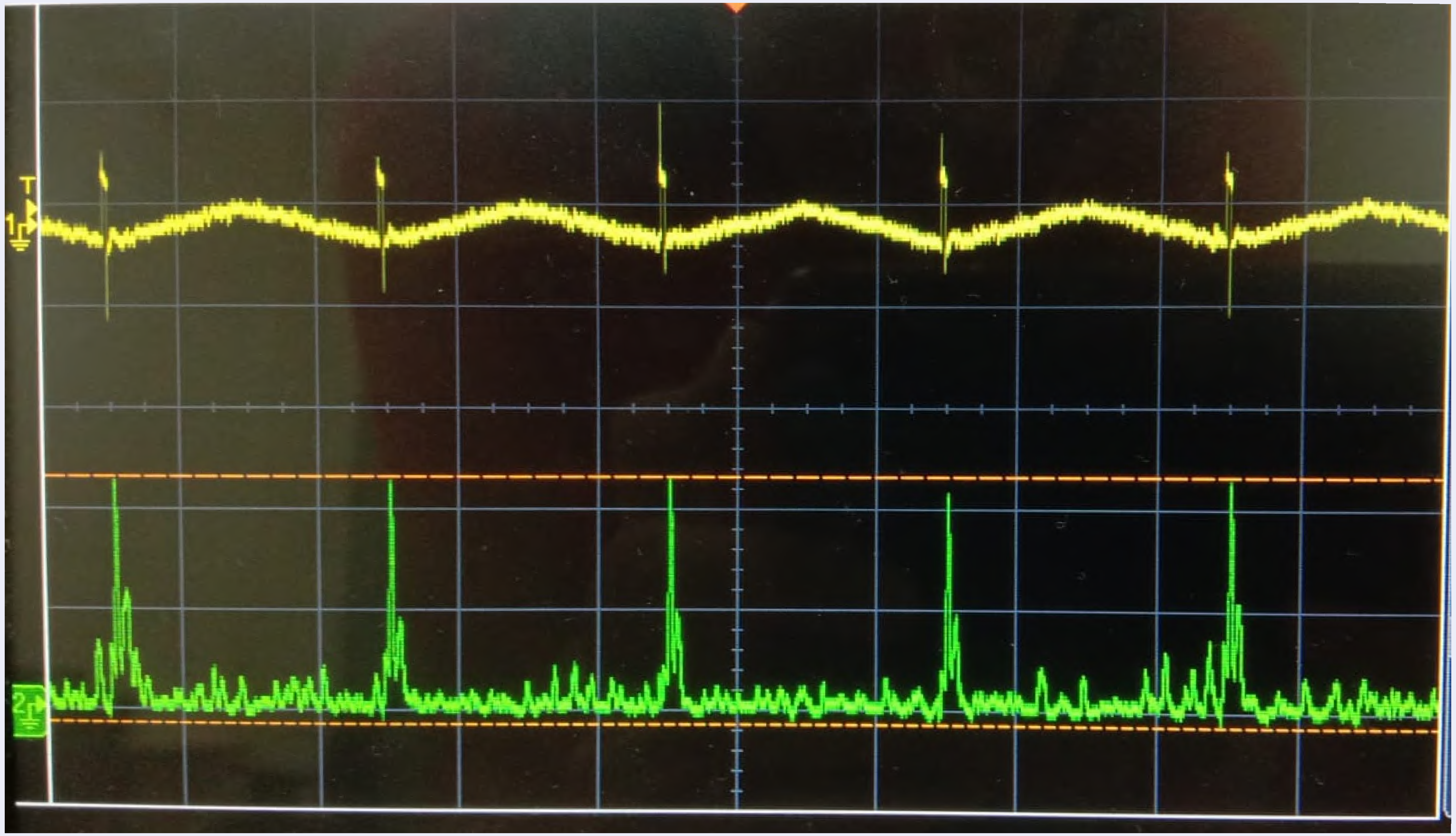
Hardware Results

Without noise



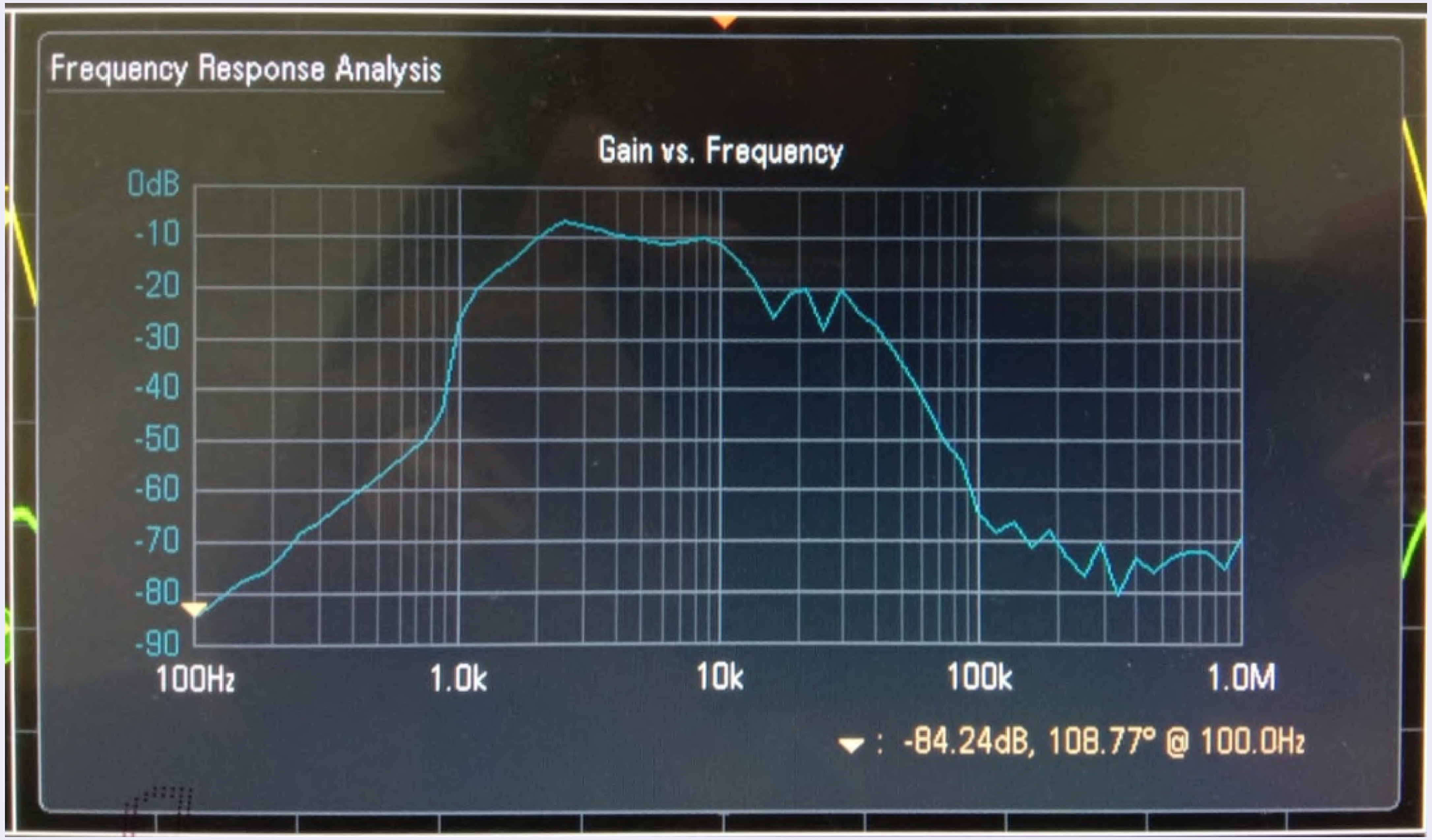


With noise



As you can see, the input is overloaded with distortion that I created by superimposing a random triangular wave on top of our *actual* spikes (pulse with duty cycle 1%), and that is further corrupted with noise. And the output shows that the circuit is successfully able to detect the *actual* spikes while neglecting other changes, and thus **I call it a success**.

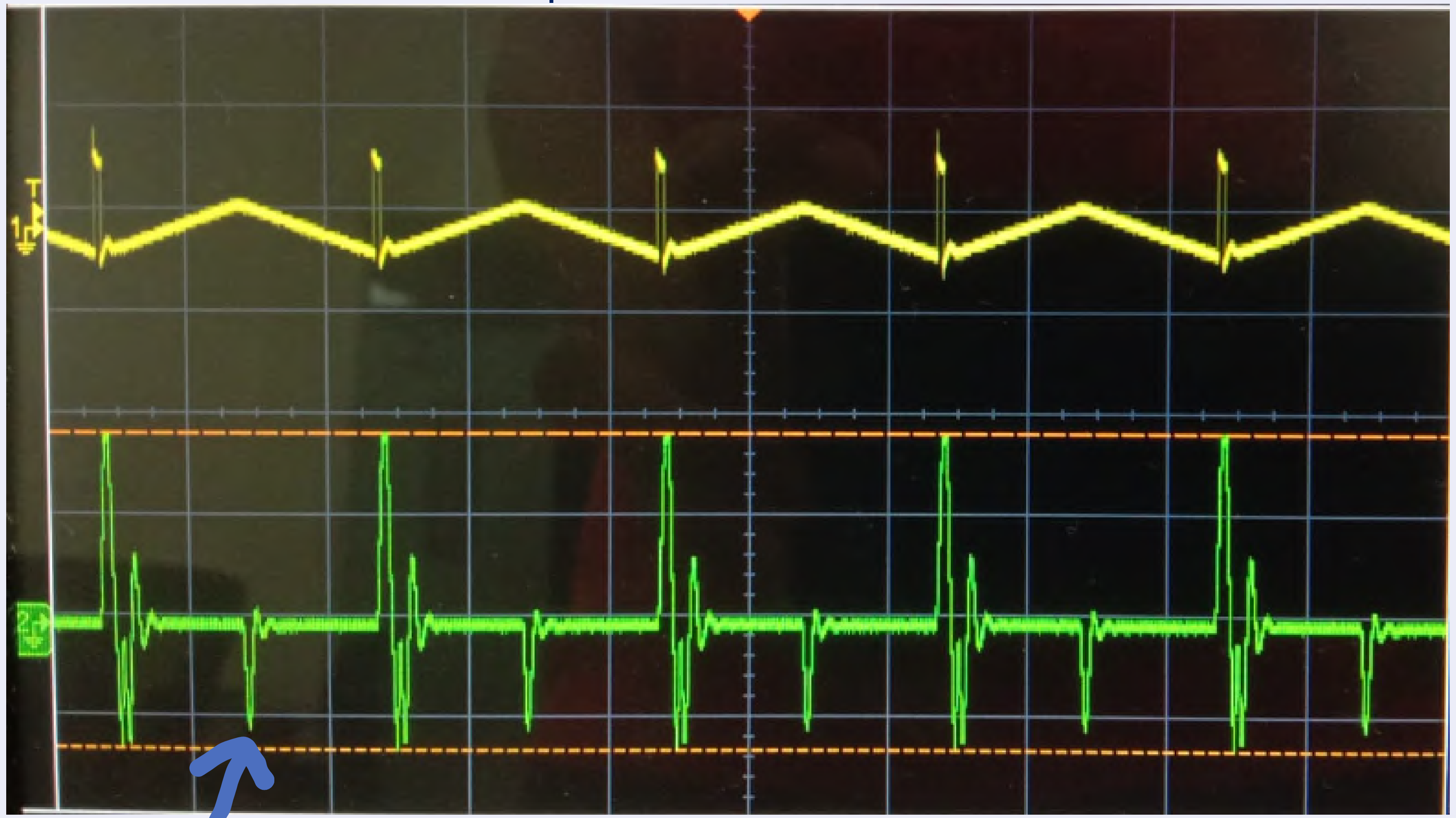
Frequency analysis



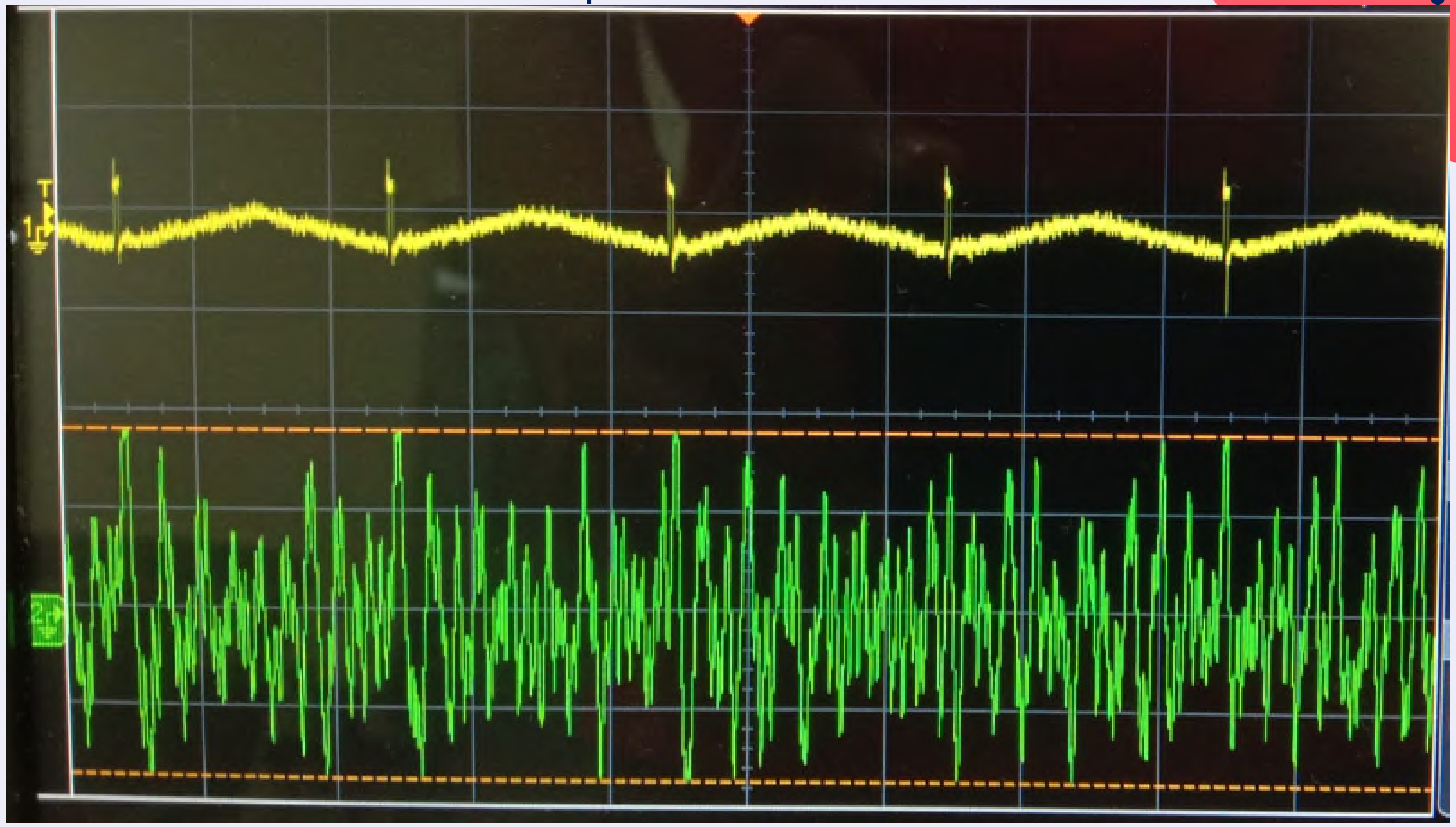
It can be seen from the Bode plot above that the circuit best performs in the range of say 1-12kHz. However, this frequency plot is misleading for our use case, this is because even tough the output signal has significant magnitudes in certain frequency range, it does not guarantee correctness. I've characterized the circuit by trying different sorts of input signals and have found that the circuit is almost always correct for 100Hz - 3000Hz. This is more than sufficient for our use case because no biological signal will actually change several thousand times a second.

V/S Only double differentiator

input without noise



input with noise



significant false alert

Clearly, if we use only the double differentiator as spike detector then it works much worse than ideal. the output is only barely okay (even had false alerts) when the input signal has high SNR but it's totally useless when input signal has high SNR.

KEY TAKEAWAY

The circuit is great at recognizing changes that are significant *and* sudden (fast enough*). Notice in the plot in last slide how the swings of the triangular wave are significant in magnitude but the change isn't fast enough. On the other hand, the noise is fast but the aggregate change introduced by it in a vicinity is not significant enough. Thus, for both these kinds of disturbances, the circuit turns out to be very robust and does not give false positives.

POSSIBLE IMPROVEMENTS

- Making the circuit more noise resistant. For instance, using an advanced low pass filter design at output (as compared to the simple RC used now).

- Introducing some way to have adaptive thresholding in the circuit, this will make the circuit more versatile by reducing constraints on i/o swings.

- Once we have a robust and reliable way of thresholding, we can implement something like a flip-flop to keep a count of the spikes occurring and based on this we do further analysis such as finding patterns in the spikes, etc.

- Making the circuit perform better at frequencies below 100Hz

REFERENCES

- Spike Detection for Integrated Circuits: Comparative Study, A. Sarje and P. Abshire
- Low SNR Neural Spike Detection Using Scaled Energy Operators for Implantable Brain Circuits, IEEE 2017
- A CMOS Implementation of the Discrete Time Nonlinear Energy Operator Based on a Transconductor-Squarer Circuit, LASCAS 2016
- Signal processing using the Teager Energy Operator and other nonlinear operators, University of Oslo Department of Informatics
- Several other papers from references of above listed ones
- Some websites that I don't remember (surfed the internet a lot because this whole thing was entirely new to me)

Thank You!