Memristors HW2

Chris Shakkour, 208157826, [christian.s@campus.technion.ac.il](mailto:christian.s@campus.technion.ac.il)

Nadi Najjar, 211610704, [nadi.najjar@campus.technion.ac.il](mailto:nadi.najjar@campus.technion.ac.il)

Contents

[Question 1 – Sneak path 2](#_Toc122334741)

[Part-a 2](#_Toc122334742)

[Part-b 2](#_Toc122334743)

[Question 2 – Sneak path and mitigation 5](#_Toc122334744)

[Part-a: Disconnected terminals 5](#_Toc122334745)

[Part-b: Grounding technique 8](#_Toc122334746)

[Part-c: Diode selector 11](#_Toc122334747)

[Question 3 – Write disturbance and mitigation 14](#_Toc122334748)

[Part-a 14](#_Toc122334749)

[Part-b 15](#_Toc122334750)

[Part-c 16](#_Toc122334751)

[Part-d 16](#_Toc122334752)

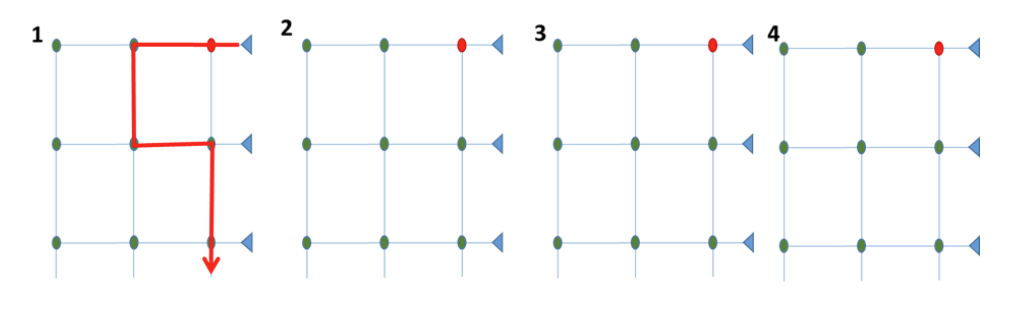
[Part-e 16](#_Toc122334753)

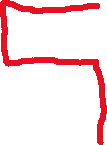
[Part-f 17](#_Toc122334754)

# Question 1 – Sneak path

## Part-a

1. All 3 turn sneak paths:

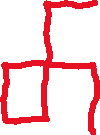
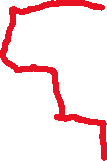




1. All 5 turn sneak paths:

Chart, line chart

Description automatically generated



1. there does not exist paths with 7 turns since we only have 6 rails hence to make 7 turns we would have to cross more than once in at least one rail.
2. The total number of sneak paths are presented above, hence a total of 8 sneak paths.

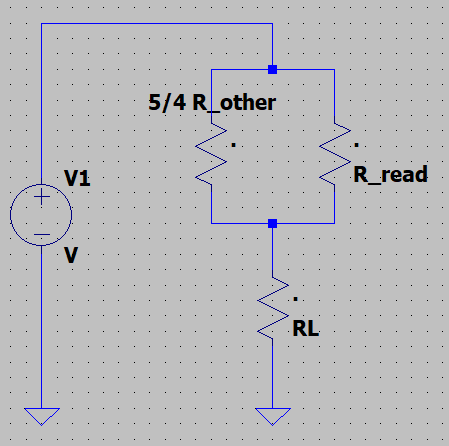
## Part-b

1. The criterion is as follows: I1\_min > l0\_max. we request that the minimal current be sufficient to switch the memristance.
2. All memristors except the one we intend to read are at high resistance meaning R\_off the one we are reading shall be put at R\_on so that most of the current flows thru this memristor, hence the current thru RL is lowest.
3. The opposite case from I0\_min will lead to sensing the highest current, all memristors are at R\_on except the memristor we are reading that should be at R\_off, the minimal current is flown thru R\_off hence the maximal current is flows thru the RL resistor.
4. Following equivalent schematics:

Schematic

Description automatically generated with medium confidence

After reduction we get the following schematics:



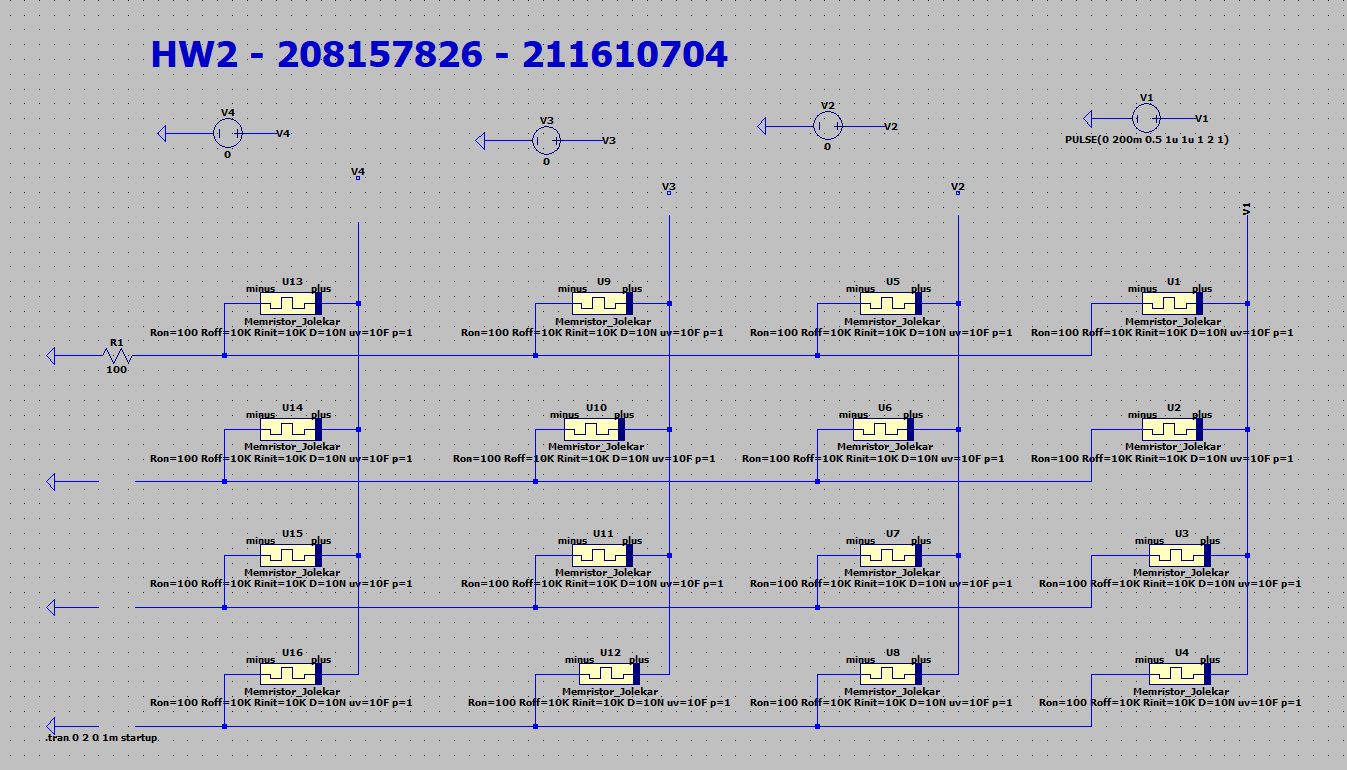
1. To calculate the current thru RL, we will use KVL Laws: and substitute R\_off for I\_min and R\_on for I\_max.
2. Assuming R\_on is way smaller than R\_off we get that R\_off is not effective on the circuit since the smaller resistor in the series has more effect on the total series resistance hence, we can say that the series resistance is equal to the resistance of R\_on.
3. The new criterion given is as follows, first let’s calculate the resistance of both :

From the equations above we conclude that

# Question 2 – Sneak path and mitigation

Following 4x4 Joglekar’s memory array in three formats.

## Part-a: Disconnected terminals



x

x

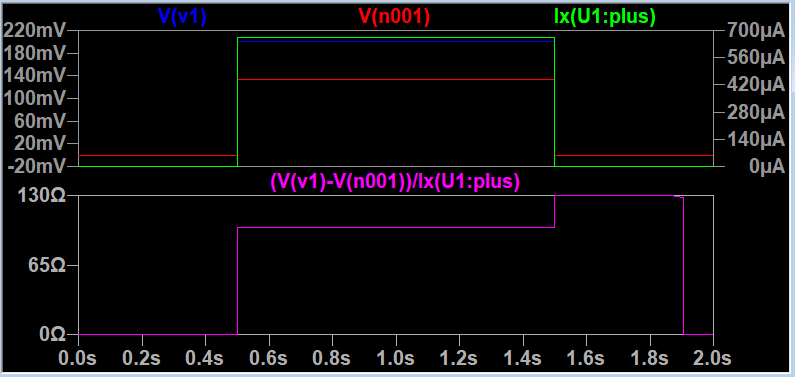
x

x

x

x

To measure the resistance on the memristor we will apply a high pulse of one second as follows:



To calculate the resistance of the Memristor we divide the Voltage across the memristor by dividing both terminal voltage over the current that flows into the memristor to get the exact resistance in a point in time see example above, V(v1) is the voltage on the positive terminal, V(n001) is the voltage across the negative terminal, and Ix(U1:plus) is the current flowing thru the memristor.

1. All resistors at R\_OFF

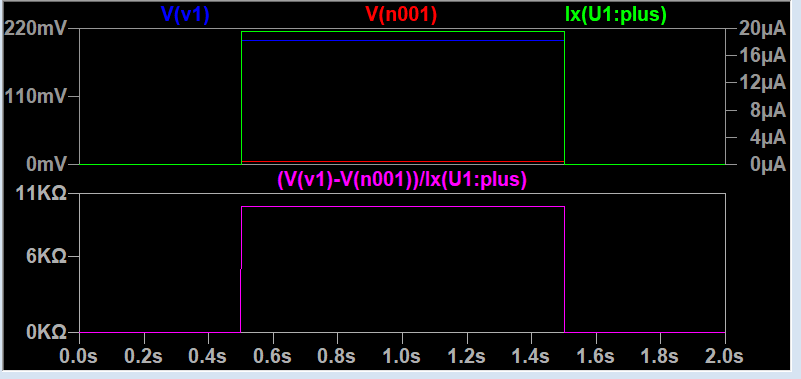


Figure : Right-top Memristor resistance pulse measurement.

1. All R\_ON except the one we are reading:

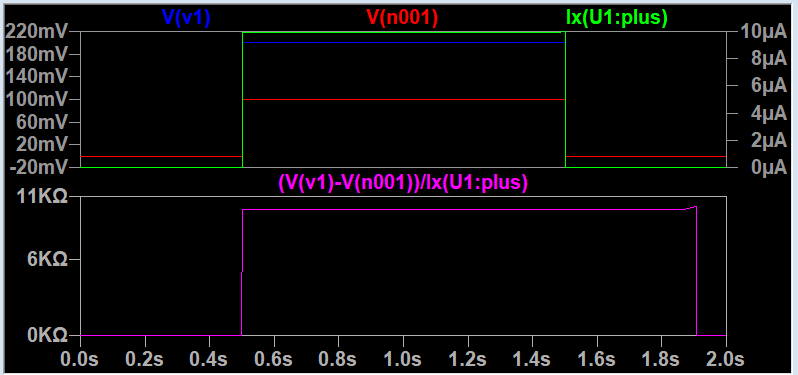


Figure 2: Right-top Memristor resistance pulse measurement.

1. All at R\_OFF except the one we are reading:

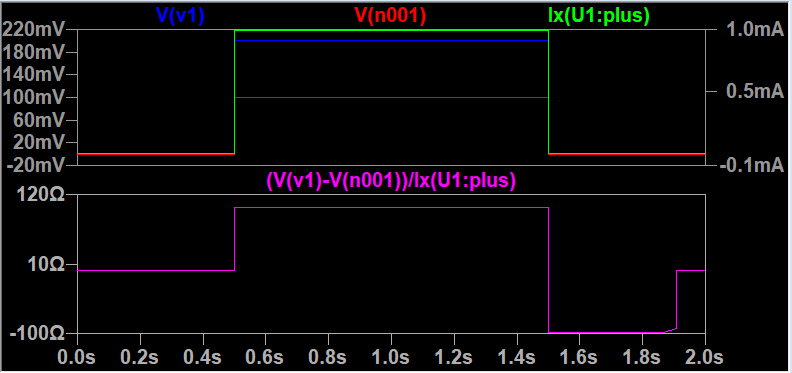


Figure 3: Right-top Memristor resistance pulse measurement.

1. All memristors at R\_ON:

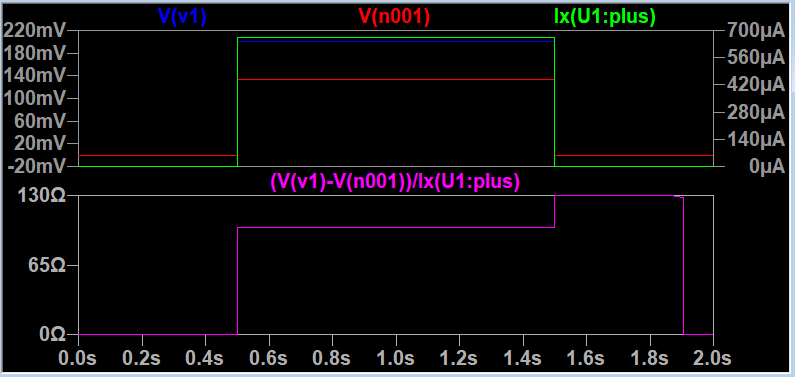
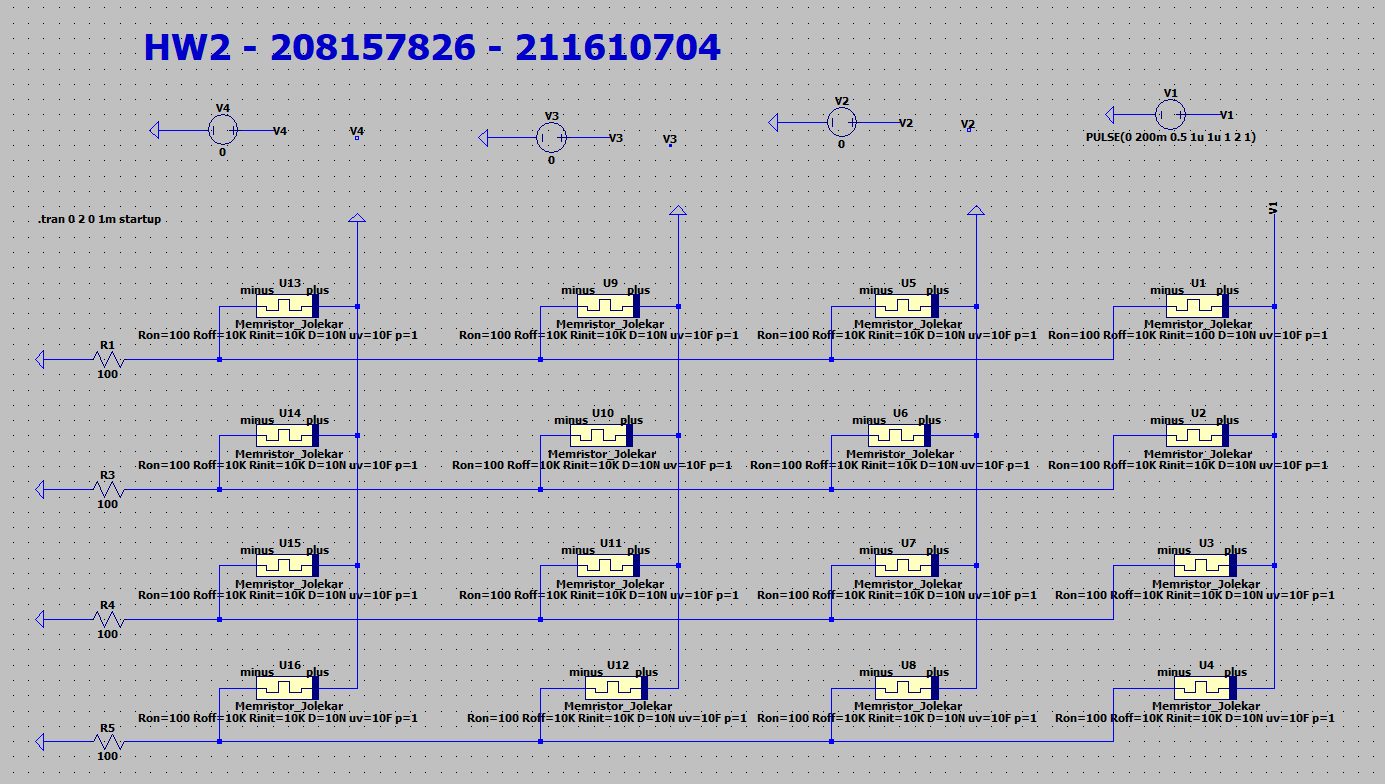


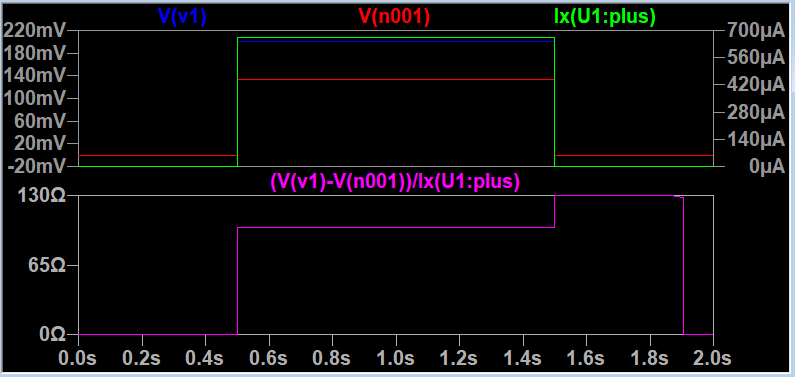
Figure 2: Right-top Memristor resistance pulse measurement.

## Part-b: Grounding technique

We apply a ground to all free terminals like shown below both WL and BL.



To measure the resistance on the memristor we will apply a high pulse of one second as follows:



To calculate the resistance of the Memristor we divide the Voltage across the memristor by dividing both terminal voltage over the current that flows into the memristor to get the exact resistance in a point in time see example above, V(v1) is the voltage on the positive terminal, V(n001) is the voltage across the negative terminal, and Ix(U1:plus) is the current flowing thru the memristor.

1. All resistors at R\_OFF

Chart

Description automatically generated

Figure : Right-top Memristor resistance pulse measurement.

1. All R\_ON except the one we are reading:

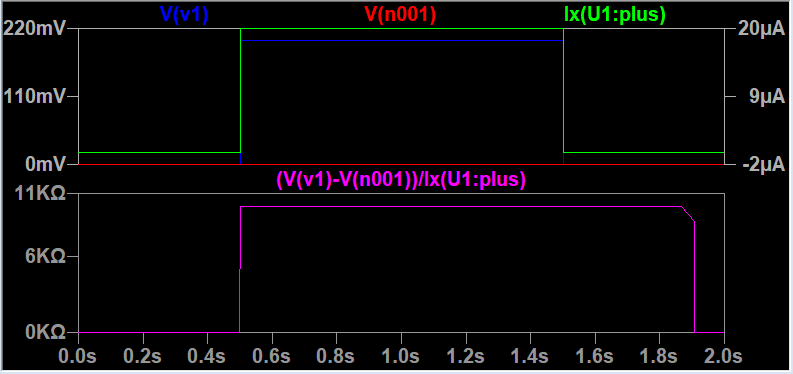


Figure 2: Right-top Memristor resistance pulse measurement.

1. All at R\_OFF except the one we are reading:

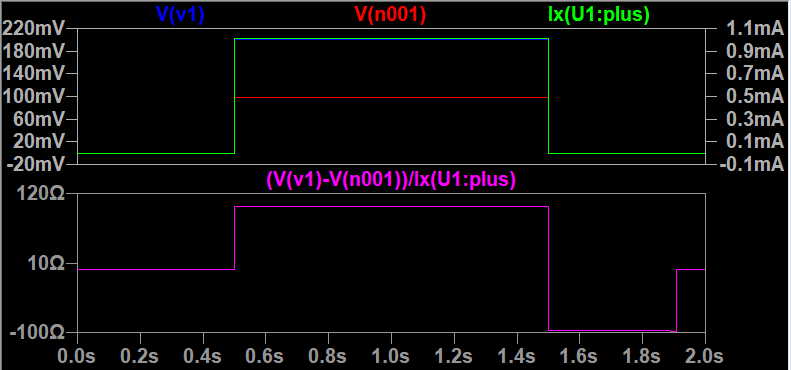


Figure 3: Right-top Memristor resistance pulse measurement.

1. All memristors at R\_ON:

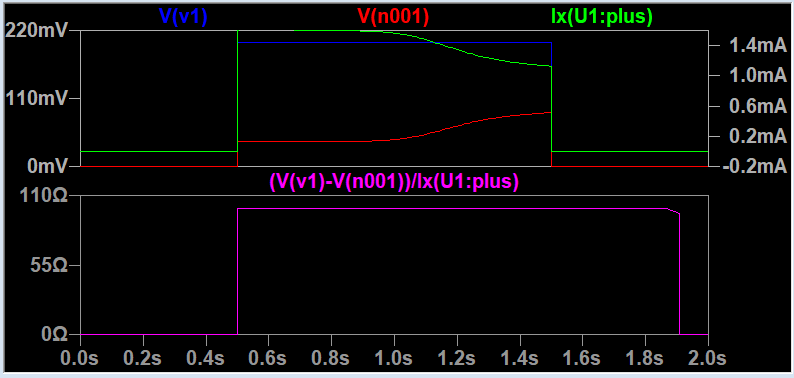
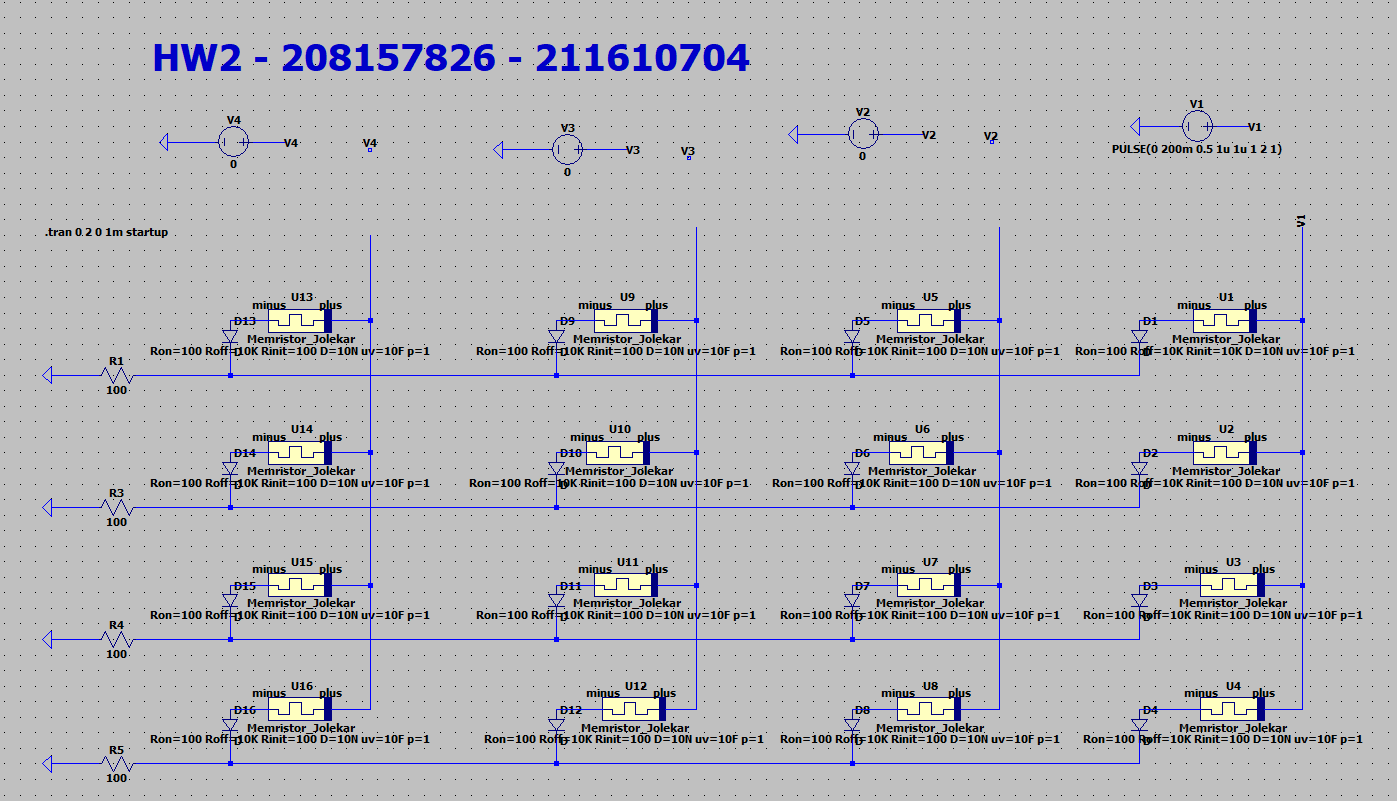


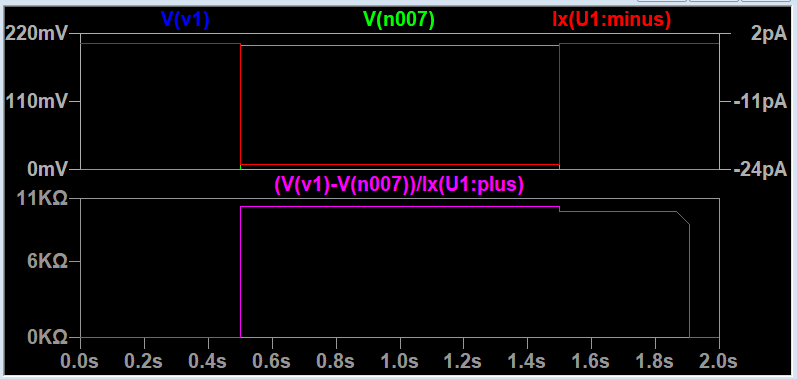
Figure 4: Right-top Memristor resistance pulse measurement.

## Part-c: Diode selector

We plug a diode to all negative terminals of the memristors.



To measure the resistance on the memristor we will apply a high pulse of one second as follows:



To calculate the resistance of the Memristor we divide the Voltage across the memristor by dividing both terminal voltage over the current that flows into the memristor to get the exact resistance in a point in time see example above, V(v1) is the voltage on the positive terminal, V(n007) is the voltage across the negative terminal, and Ix(U1:plus) is the current flowing thru the memristor.

1. All resistors at R\_OFF

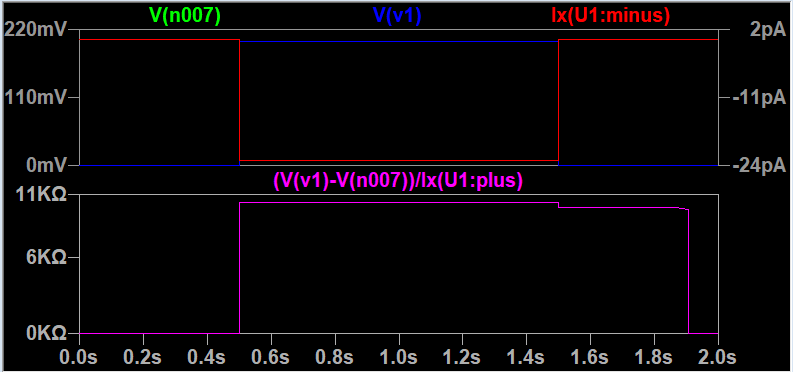


Figure : Right-top Memristor resistance pulse measurement.

1. All R\_ON except the one we are reading:

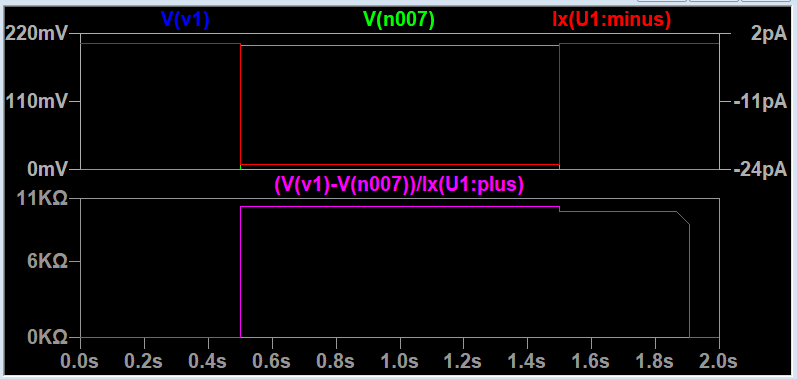


Figure 2: Right-top Memristor resistance pulse measurement.

1. All at R\_OFF except the one we are reading:

Chart

Description automatically generated with low confidence

Figure 3: Right-top Memristor resistance pulse measurement.

1. All memristors at R\_ON:

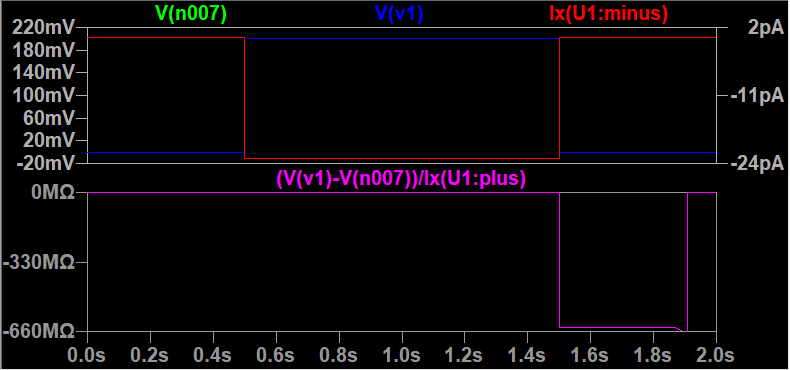


Figure 4: Right-top Memristor resistance pulse measurement.

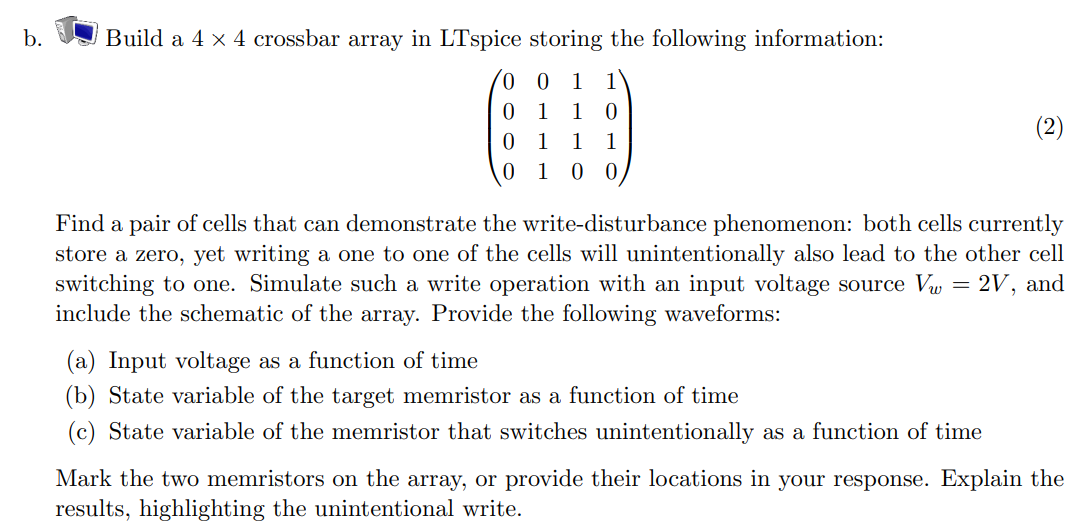
# Question 3 – Write disturbance and mitigation

## Part-a



Since now the mobility is higher it is easier to transition from one side to the other and the noise margin is much smaller and the metastability is greater hence, to ensure a clean transition we would start with values greater and smaller than the values on the edges where W/D = [0, 1].

## Part-b



**Circuit Schematic:** Timeline

Description automatically generated



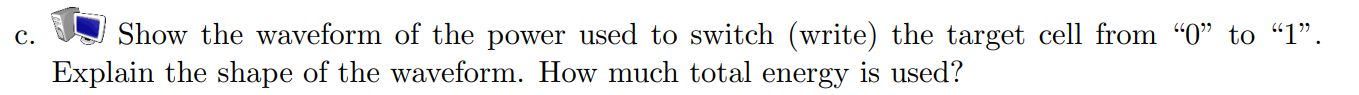
Plot of the input voltage, state variable of U9 and state variable of U13:



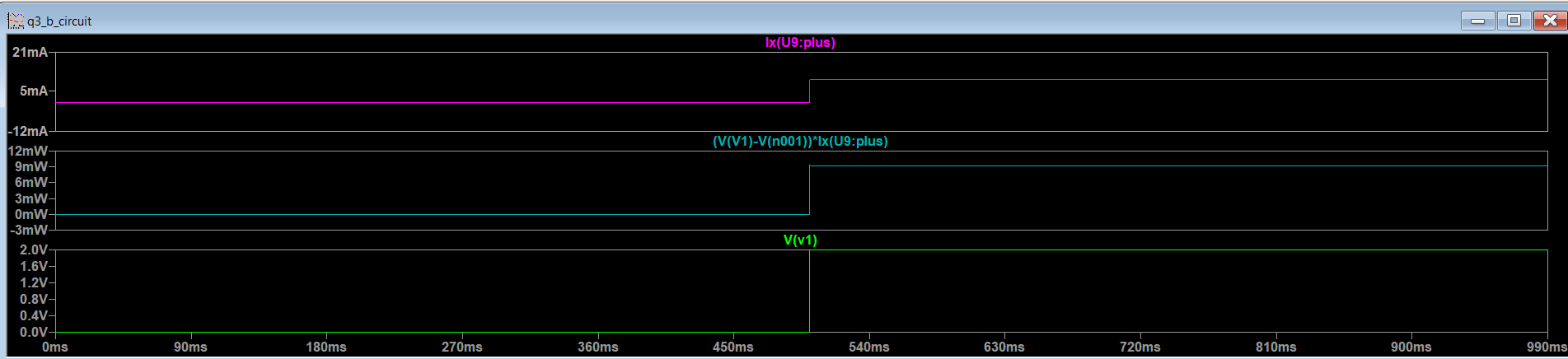
We can see from U9 state variable that we successfully wrote the value of ‘1’ on it so we can see the transaction between 0 and 1.

Unfortunately, we can see that U13 is also affected by the write procedure, since there is a sneak path that pass through U13, and from the plot of U13 state variable that we wrote the value of ‘1’ in it.

## Part-c

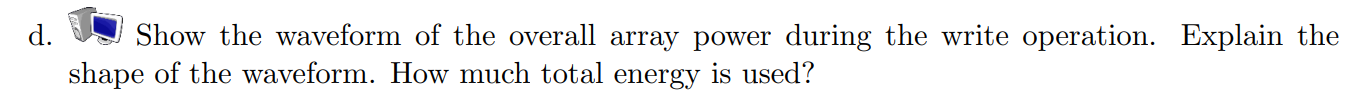


Plot of the power used to switch U9 from ‘0’ to ‘1’ (blue marked line):

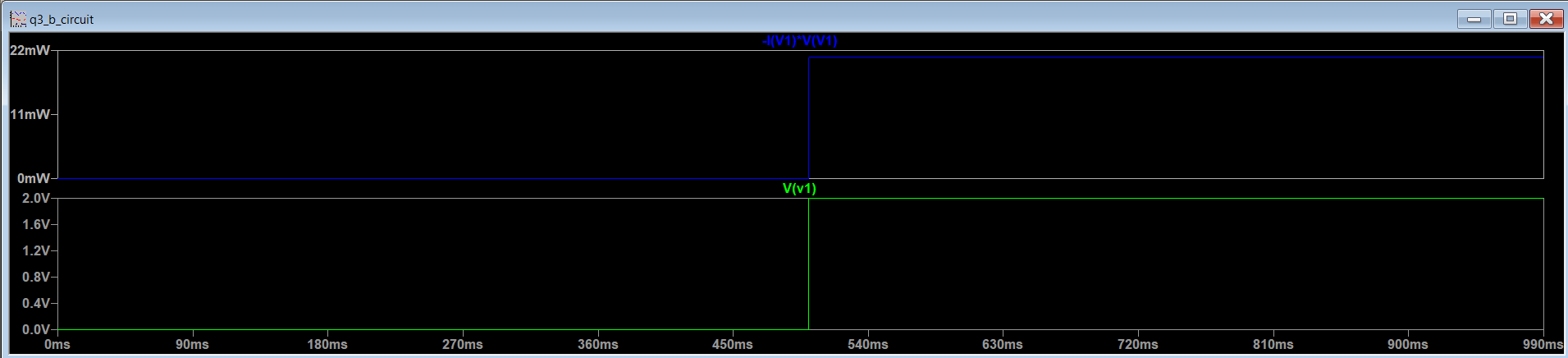


Since the input voltage applied to U9 (for write procedure) is a pulse, we got a usage of power that looks like a pulse, before the switching its 0mW and during the switching we used about 9mW of power.

## Part-d

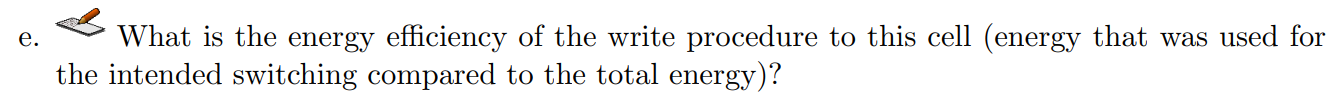


Plot of the power used in the array during the write procedure (blue marked line):



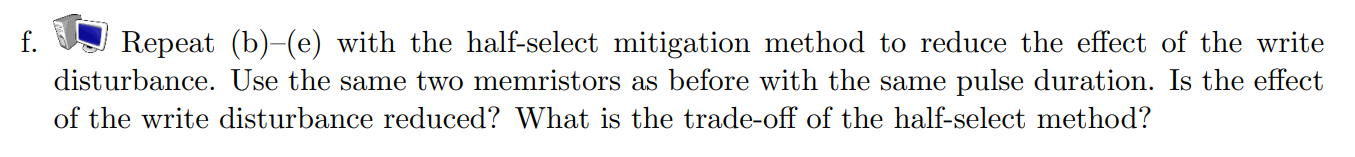
We will get the same shape of the power used in U9 for the same reason mentioned in 3.c, but the value of the power used in the writing procedure is about 22mW, that includes the unintended write procedure done on U13 Memristor.

## Part-e



The power used to switch U9 from ‘0’ to ‘1’ is 10mW and the total power wasted in the circuit is 22mW (from previous sections), so the efficiency is calculated here:

## Part-f



**f.b-plot**

A screenshot of a computer

Description automatically generated with medium confidence

**f.c-plot**

A screenshot of a computer

Description automatically generated with medium confidence

**f.b-plot**

Graphical user interface, text, application

Description automatically generated

We can see from f.b plot (the plot of state variable of U13) that the U13 Wont be written unintentionally, so the half-select method worked successfully to prevent the sneak path the passes through U13.

Power consumption through the write process of U9 in the array increased to almost 50mW so we got an efficiency of: