Memristors HW2

Chris Shakkour, 208157826, christian.s@campus.technion.ac.il

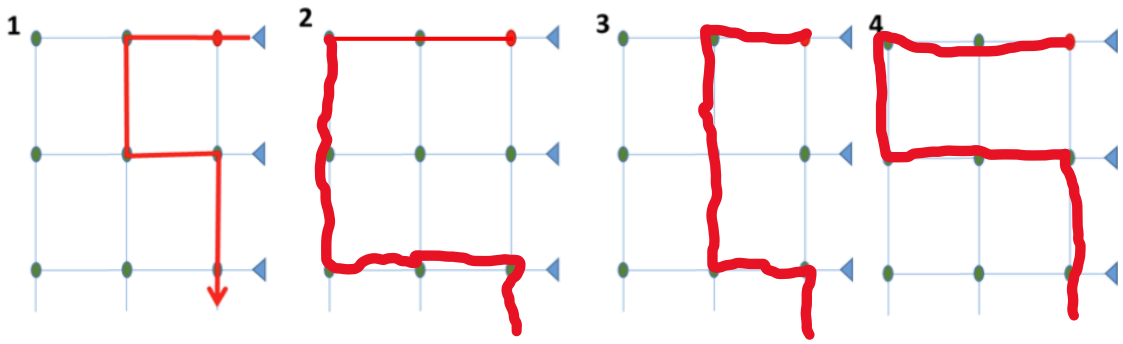
Nadi Najjar, 211610704, nadi.najjar@campus.technion.ac.il

Contents

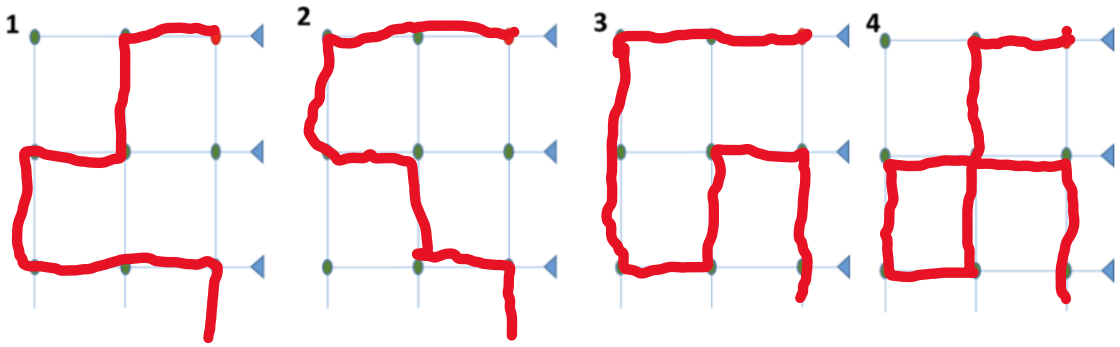
Question 1 – Sneak path

Part-a

a. All 3 turn sneak paths:



b. All 5 turn sneak paths:



c. 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.

d. The total number of sneak paths are presented above, hence a total of 8 sneak paths.

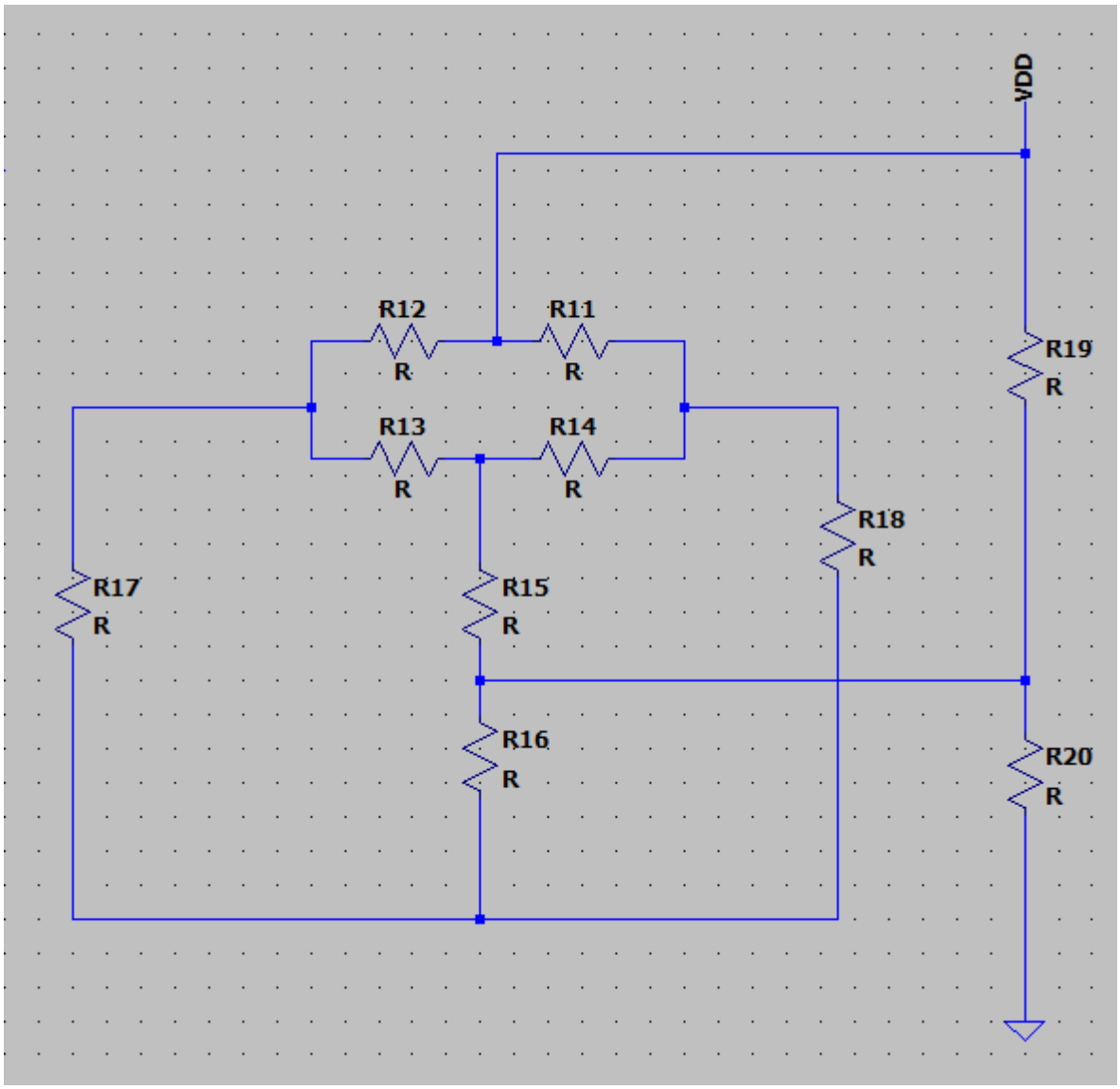
Part-b

a. The criterion is as follows: I1_min > I0_max. we request that the minimal current be sufficient to switch the memristance.

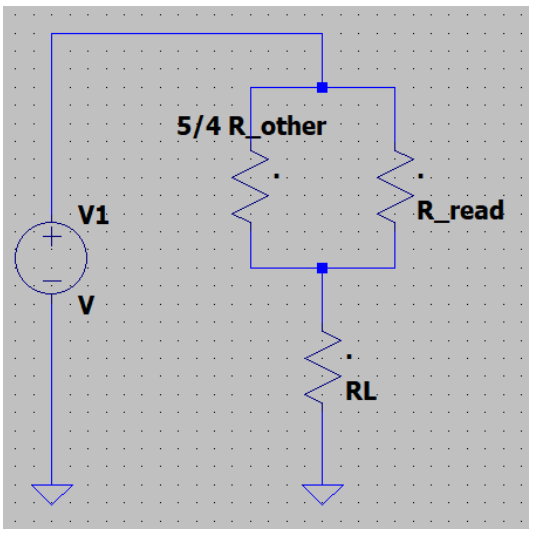
b. All memristors except the one we intend to read are at high resistance meaning R_off the one we are reading shall be put at R_on so that most of the current flows thru this memristor, hence the current thru RL is lowest.

c. 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.

d. Following equivalent schematics:



After reduction we get the following schematics:



e. To calculate the current thru RL, we will use KVL Laws: and substitute R_off for I_min and R_on for I_max.

$$I_{min} = \frac{V}{RL + (R_{read} || \frac{5}{4} R_{other})} = \frac{V}{RL + (R_{on} || \frac{5}{4} R_{off})}$$

$$I_{max} = \frac{V}{RL + (R_{read} || \frac{5}{4} R_{other})} = \frac{V}{RL + (R_{off} || \frac{5}{4} R_{on})}$$

f. 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.

$$I_{min} = \frac{V}{RL + (R_{read} || \frac{5}{4} R_{other})} = \frac{V}{RL + (R_{on} || \frac{5}{4} R_{off})} \rightarrow R_{on} \ll R_{off} \rightarrow \frac{V}{RL + R_{on}}$$

g. The new criterion given $R_{on} \ll R_{off}$ is as follows, first let's calculate the resistance of both I_{min}, I_{max}:

$$I_{min} \rightarrow R_{on} \ll R_{off} \rightarrow \frac{V}{RL + R_{on}}$$

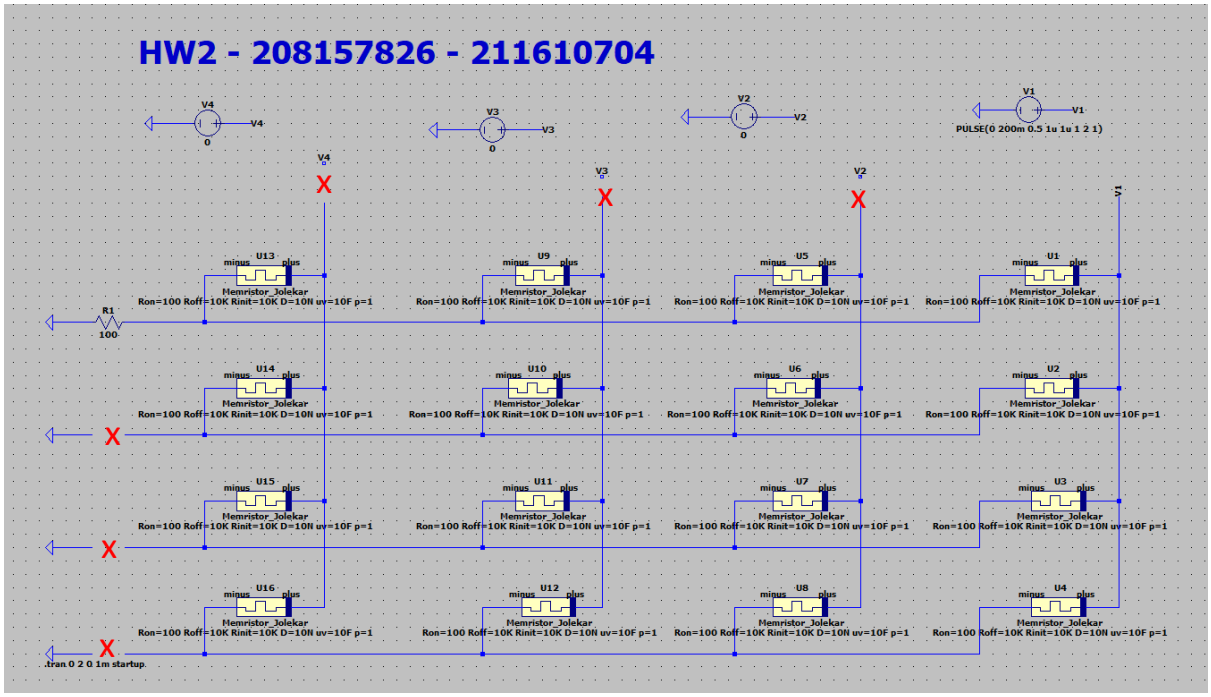
$$I_{max} \rightarrow R_{on} \ll R_{off} \rightarrow \frac{V}{RL + \frac{5}{4} R_{on}}$$

From the equations above we conclude that $I_{min}, > I_{max}$

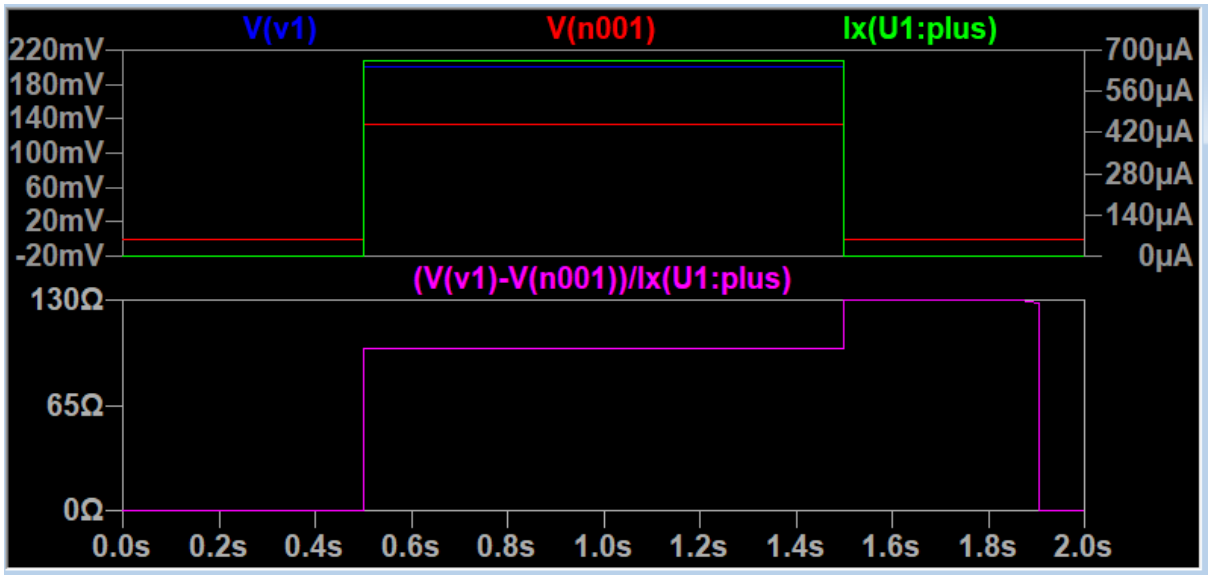
Question 2 – Sneak path and mitigation

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

Part-a: Disconnected terminals



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.

a. All resistors at R_OFF

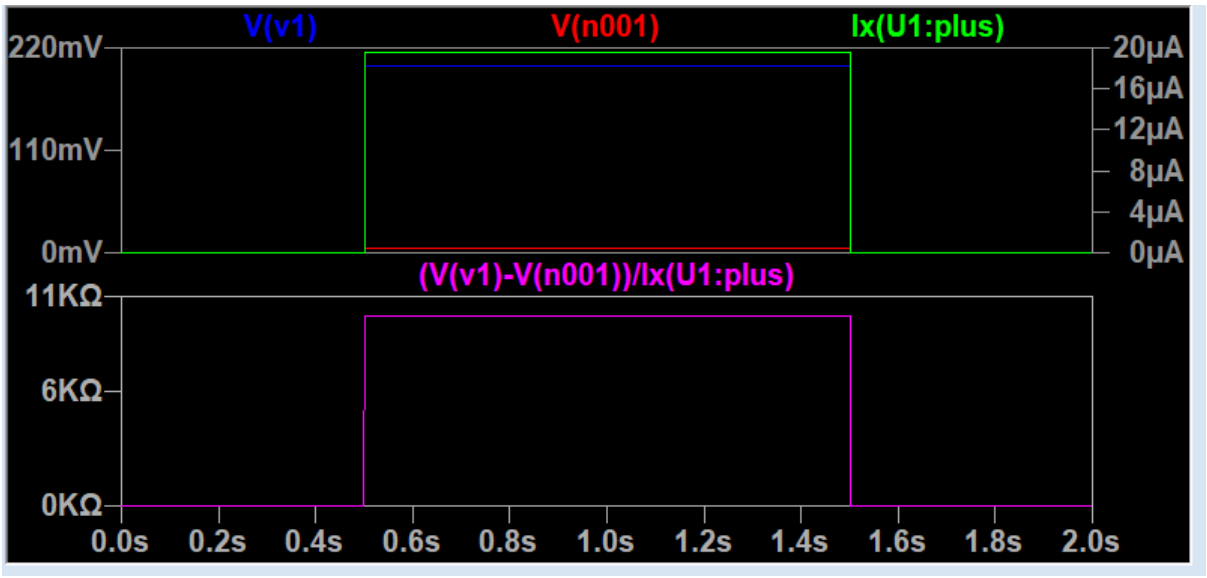


Figure 1: Right-top Memristor resistance pulse measurement.

b. All R_ON except the one we are reading:

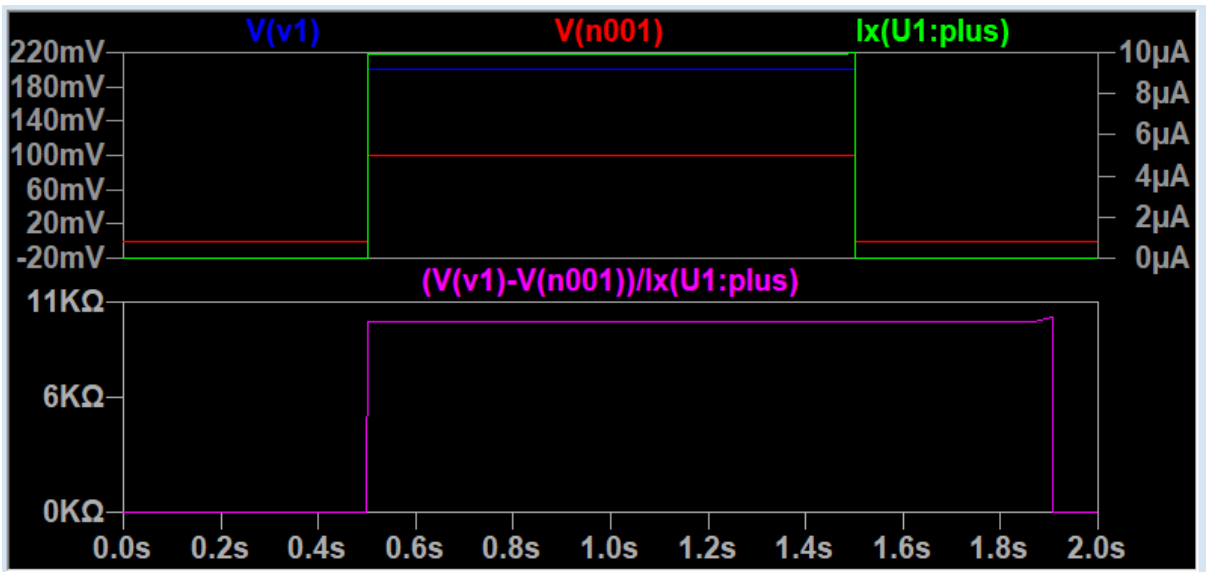


Figure 2: Right-top Memristor resistance pulse measurement.

c. All at R_OFF except the one we are reading:

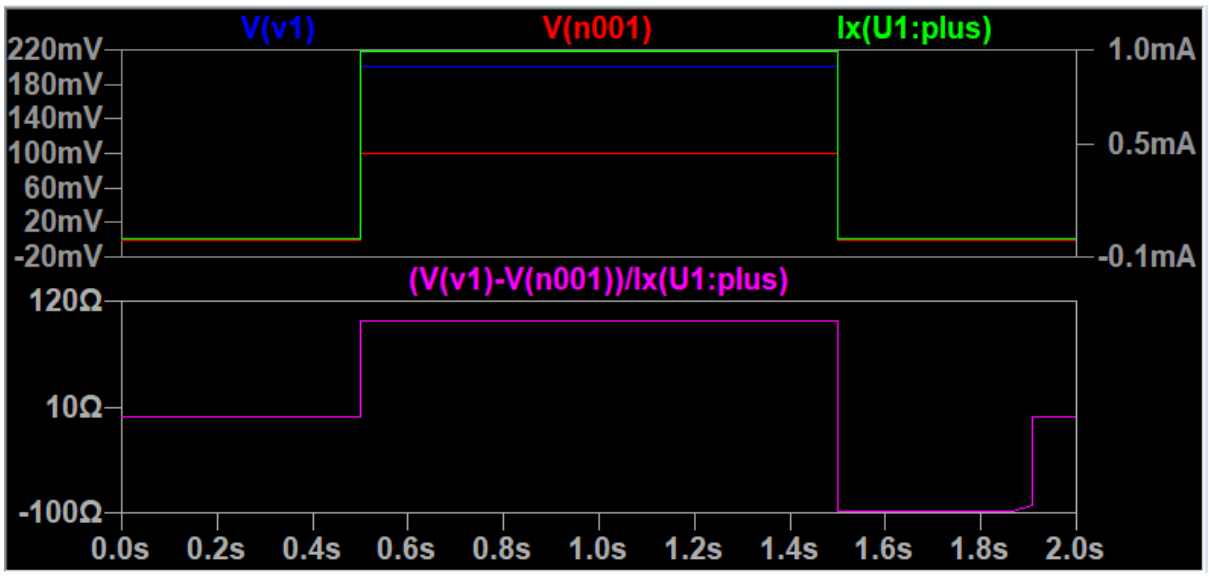


Figure 3: Right-top Memristor resistance pulse measurement.

d. All memristors at R_ON:

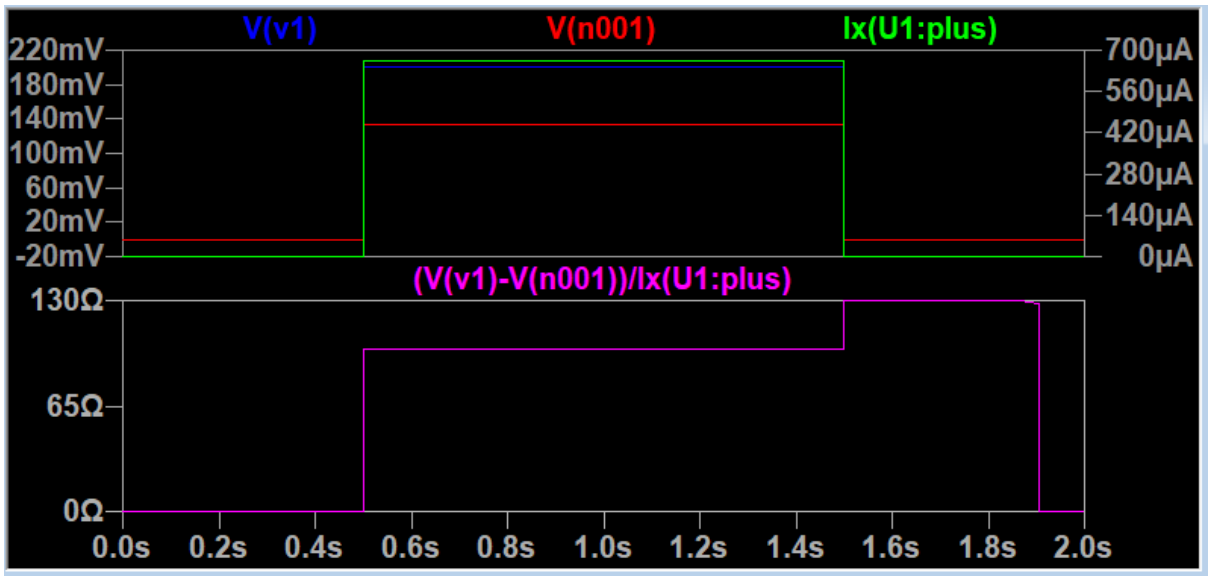
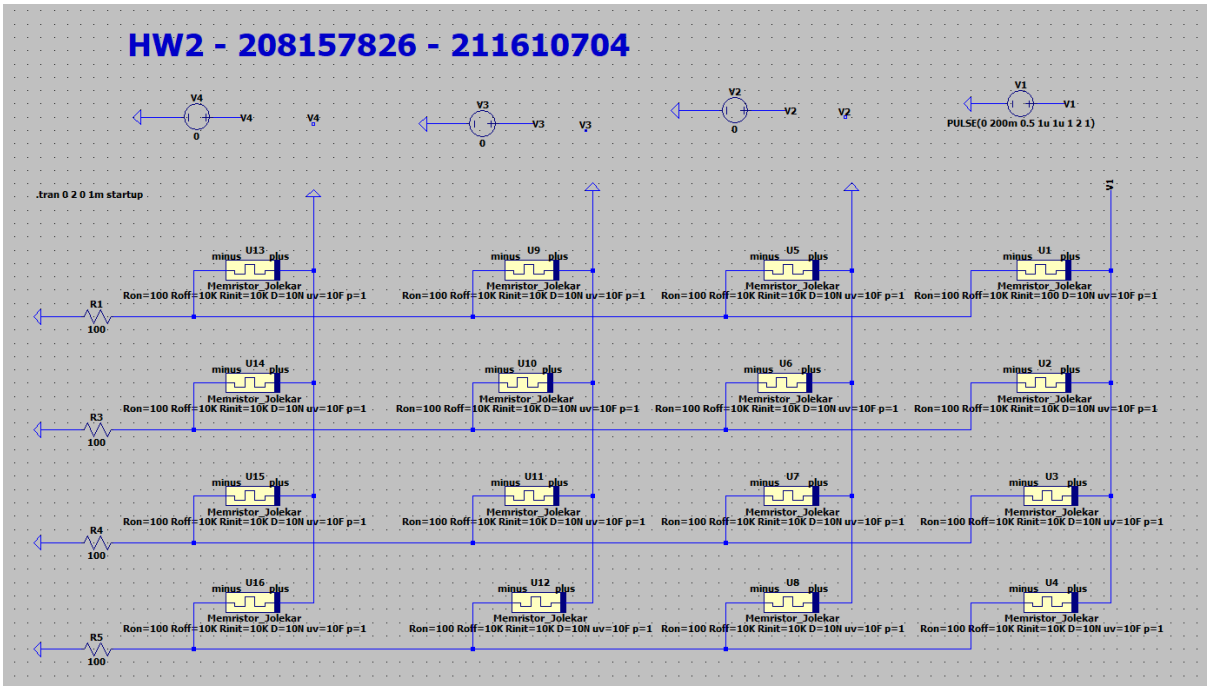


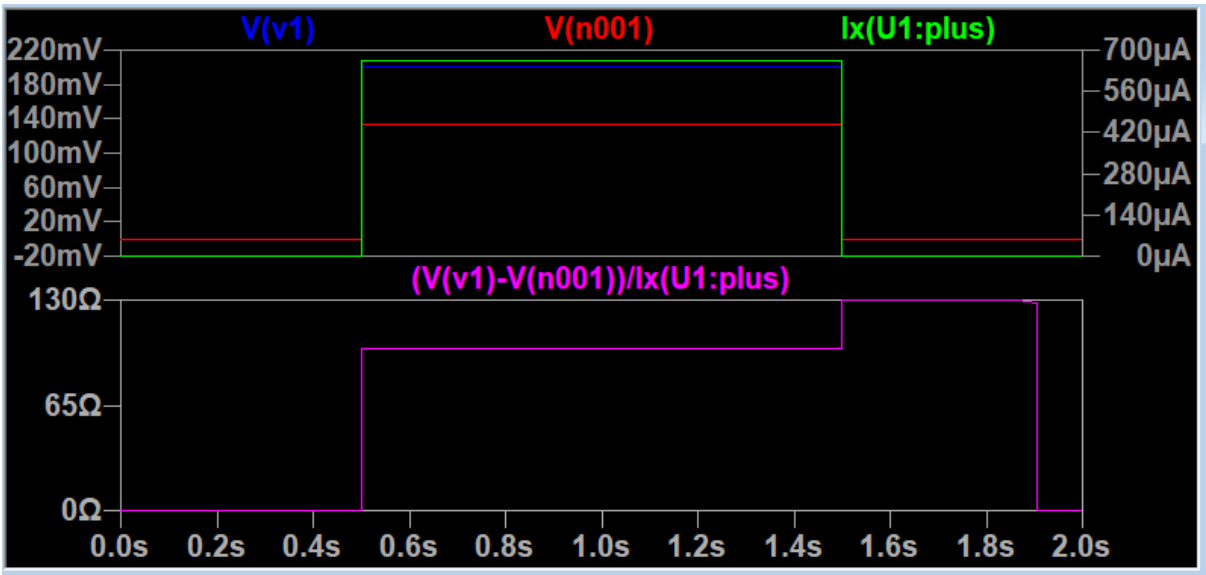
Figure 2: Right-top Memristor resistance pulse measurement.

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.

a. All resistors at R_OFF

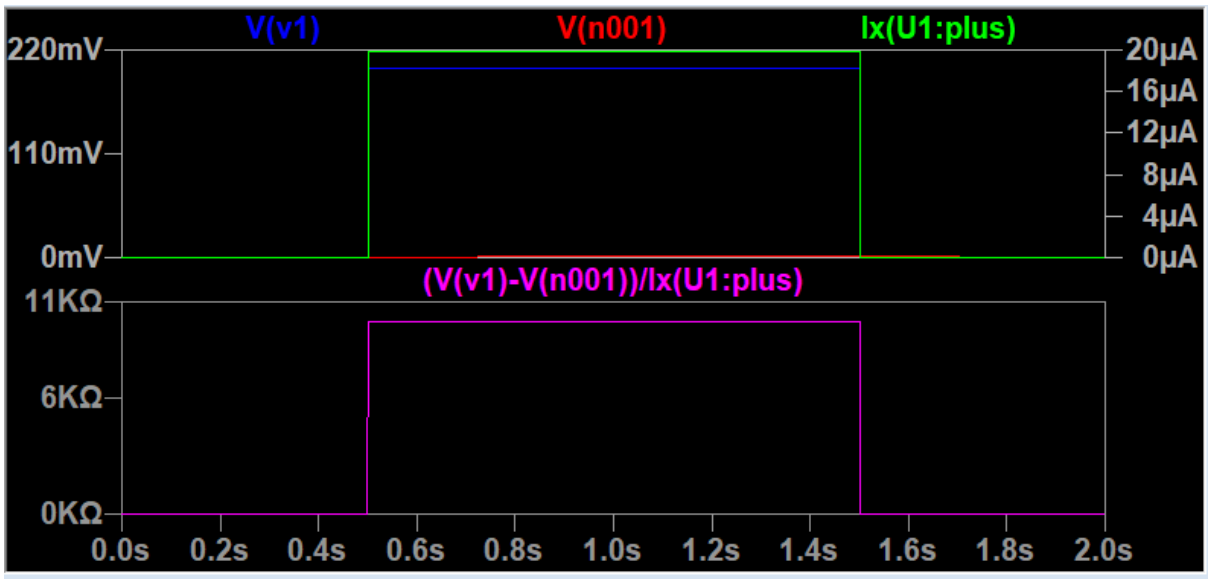


Figure 2: Right-top Memristor resistance pulse measurement.

b. All R_ON except the one we are reading:

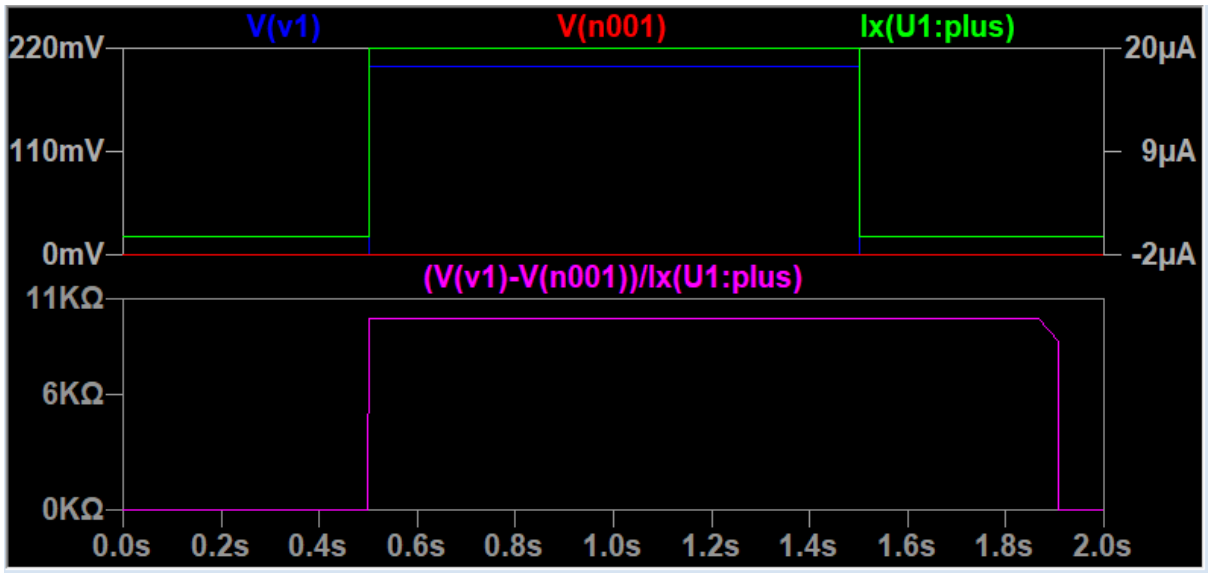


Figure 2: Right-top Memristor resistance pulse measurement.

c. All at R_OFF except the one we are reading:

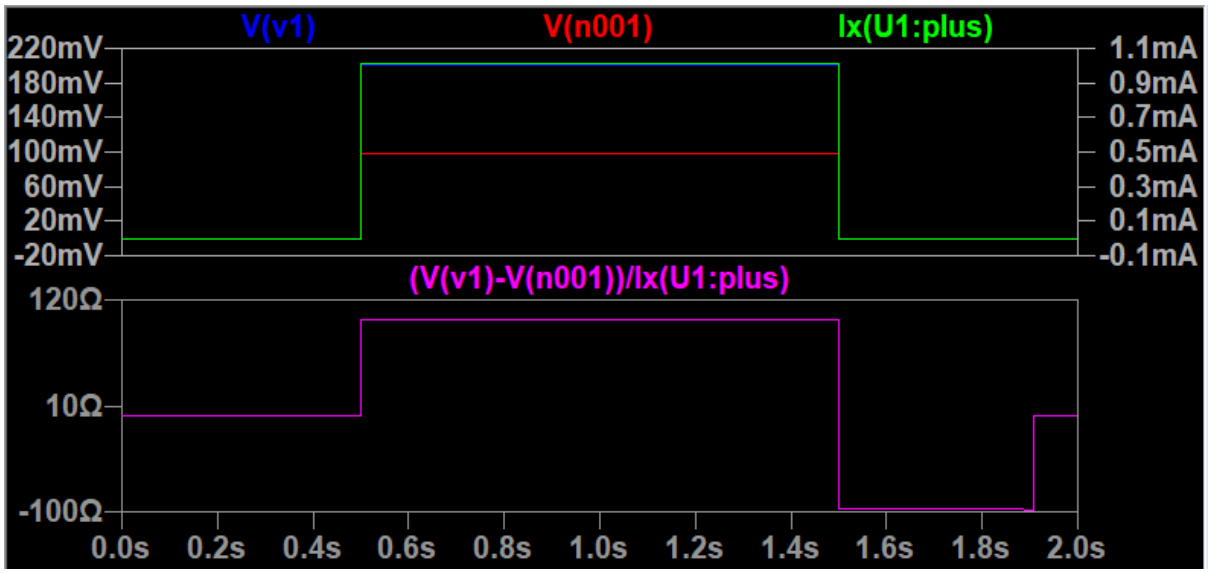


Figure 3: Right-top Memristor resistance pulse measurement.

d. All memristors at R_ON:

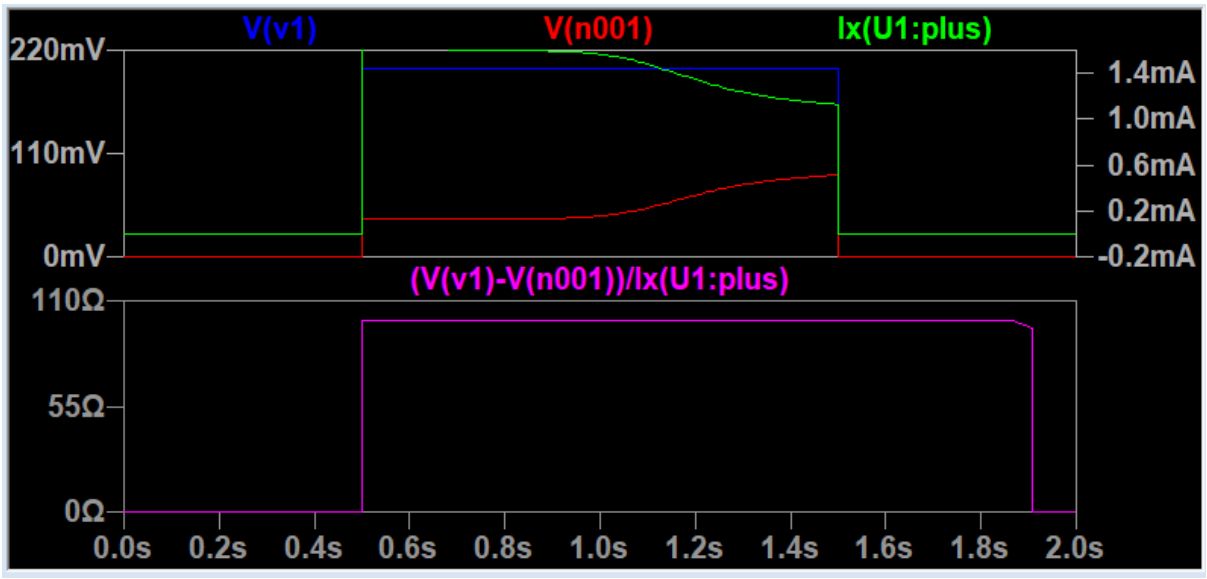
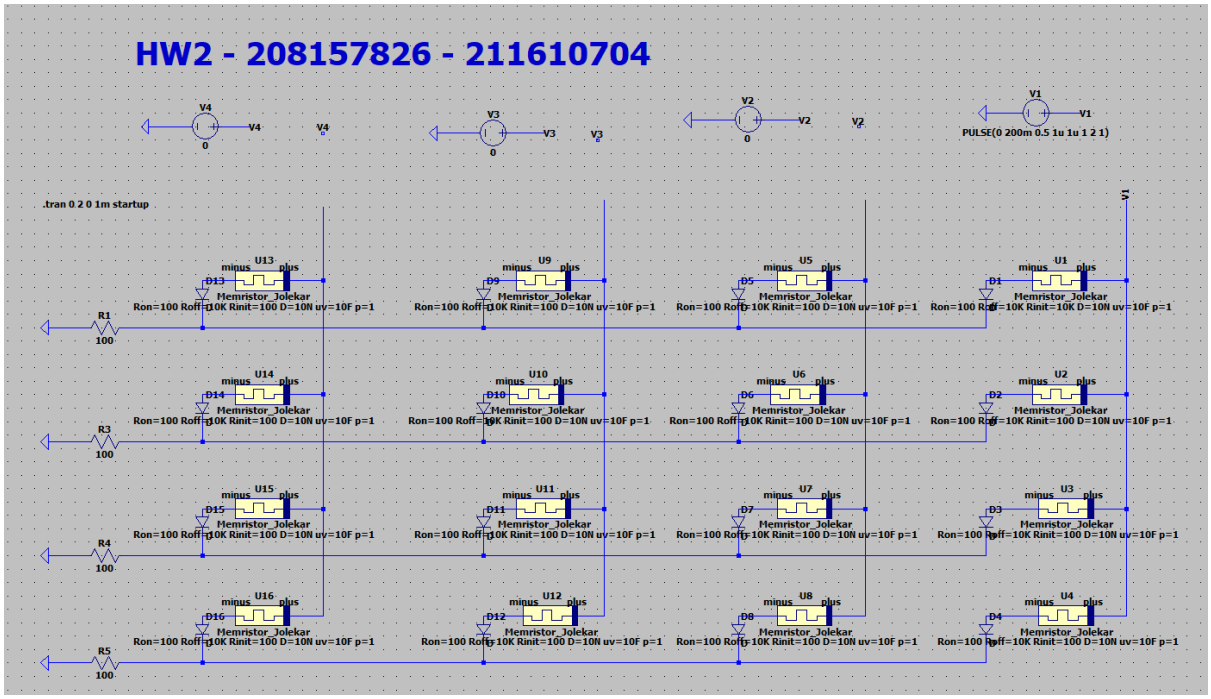


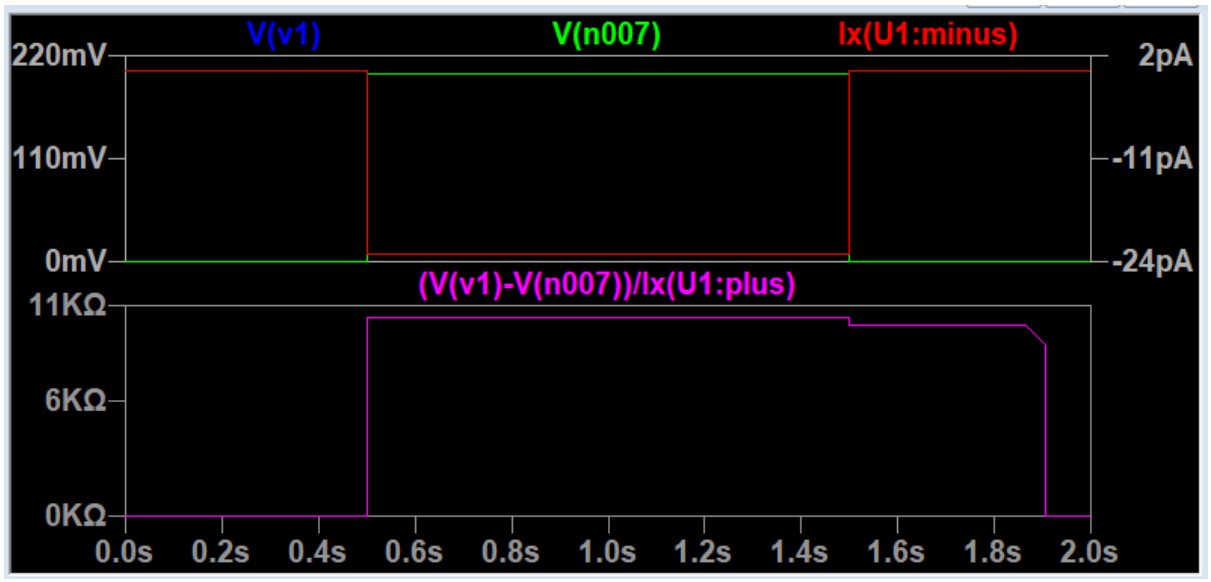
Figure 4: Right-top Memristor resistance pulse measurement.

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.

a. All resistors at R_OFF

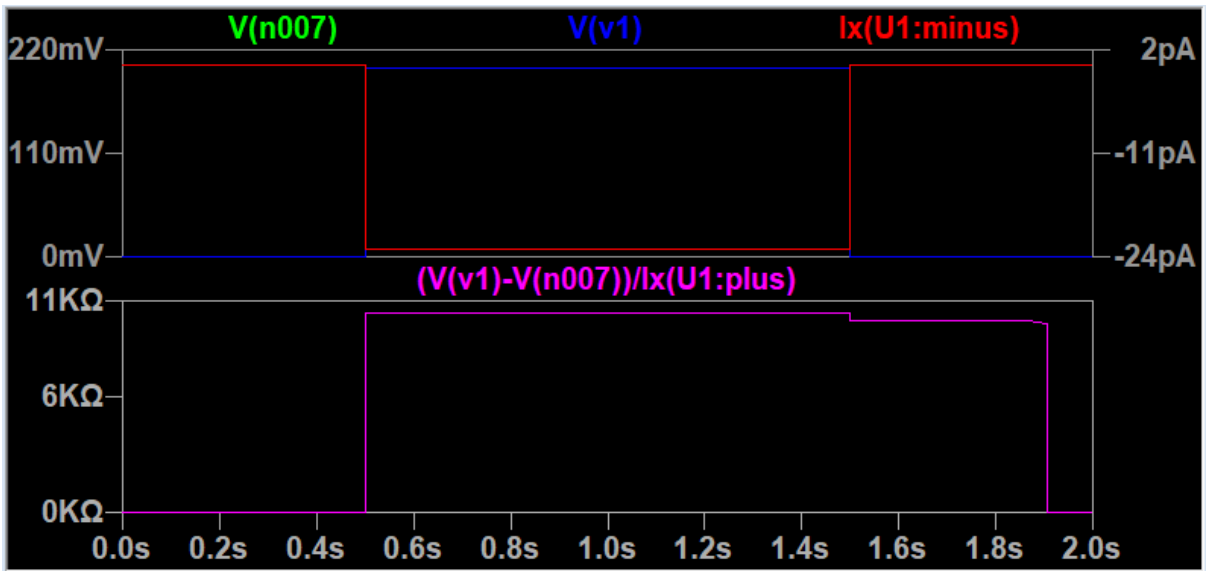


Figure 3: Right-top Memristor resistance pulse measurement.

b. All R_ON except the one we are reading:

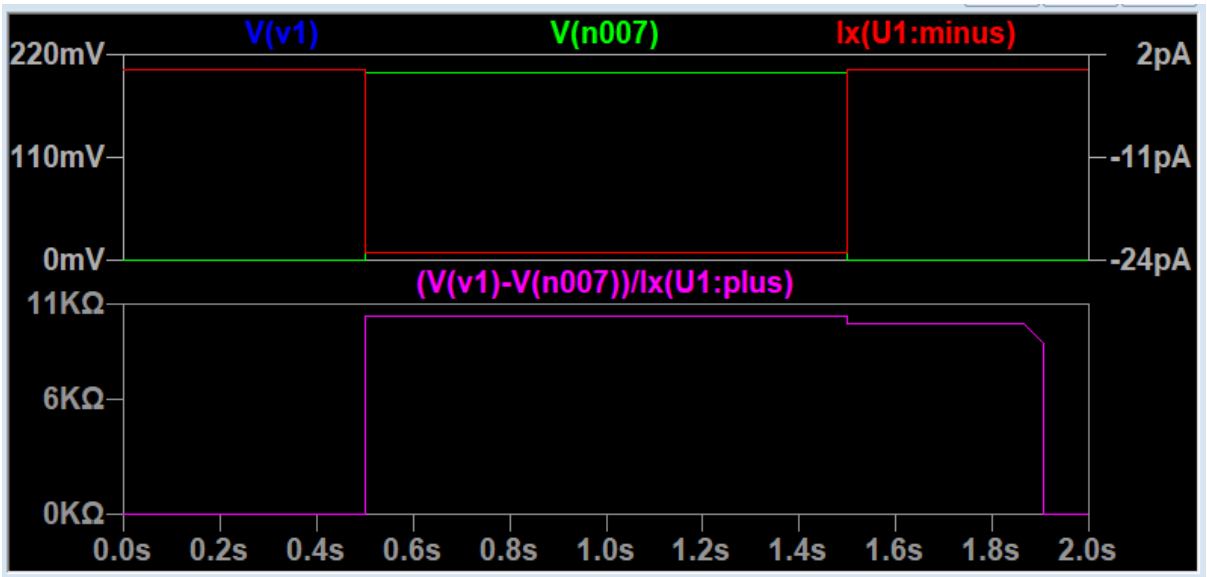


Figure 2: Right-top Memristor resistance pulse measurement.

c. All at R_OFF except the one we are reading:

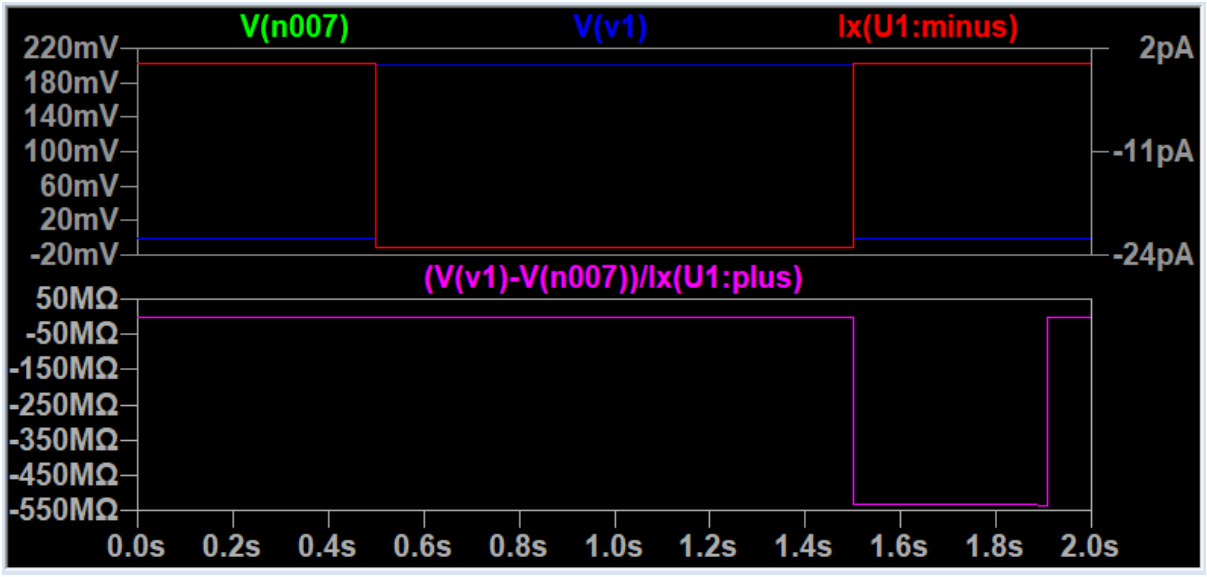


Figure 3: Right-top Memristor resistance pulse measurement.

d. All memristors at R_ON:

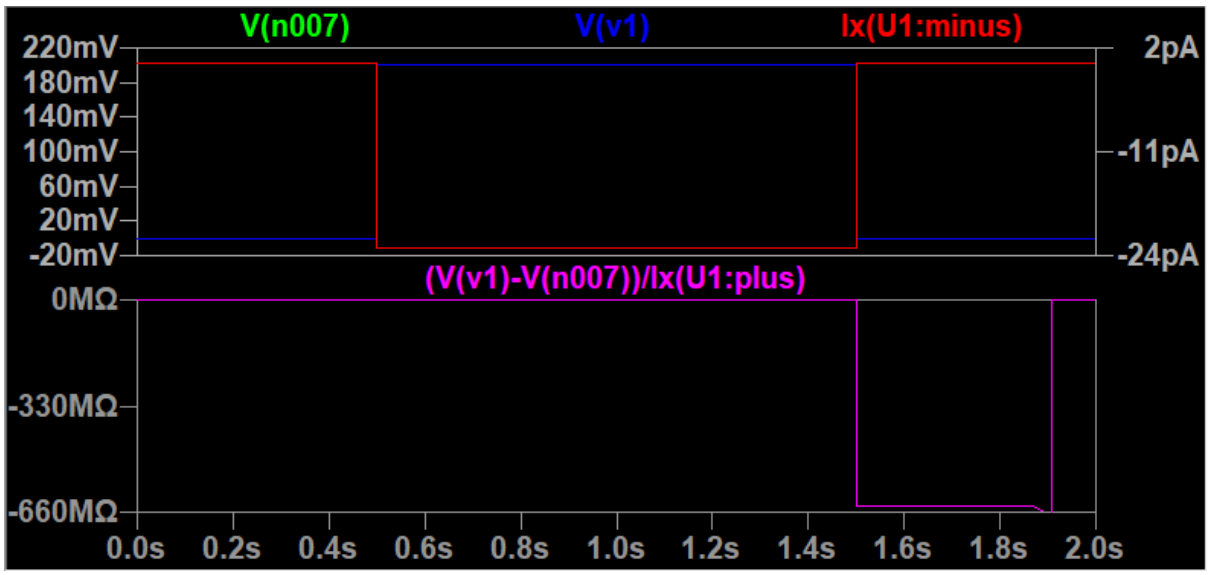


Figure 4: Right-top Memristor resistance pulse measurement.

Question 3 – Write disturbance and mitigation

Question 3 – Write Disturbance and Mitigation

In this question, you will investigate the write disturbance phenomenon and its mitigation techniques in LTspice by designing a 4×4 memory array. Use the Joglekar model from HW1, with the following parameters (notice that the parameters are different from Question 2):

Parameter	Value
R_{ON} (Ron)	100Ω
R_{OFF} (Roff)	$10k\Omega$
D (D)	$1N$
μ (uv)	$1000F$
p (p)	1

where R_{init} is chosen according to $9k\Omega$ for "0" and 200Ω for "1".

Part-a

a. Why is R_{init} chosen as either 200Ω or $9k\Omega$ instead of 100Ω or $10k\Omega$?

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

b. Build a 4×4 crossbar array in LTspice storing the following information:

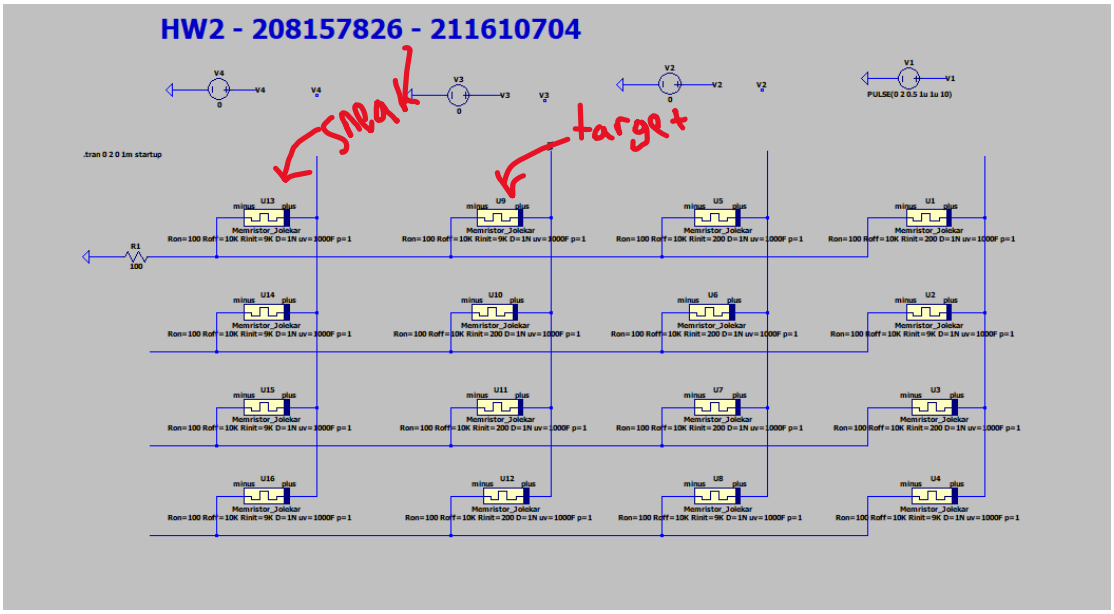
$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{2}$$

Find a pair of cells that can demonstrate the write-disturbance phenomenon: both cells currently store a zero, yet writing a one to one of the cells will unintentionally also lead to the other cell switching to one. Simulate such a write operation with an input voltage source $V_w = 2V$, and include the schematic of the array. Provide the following waveforms:

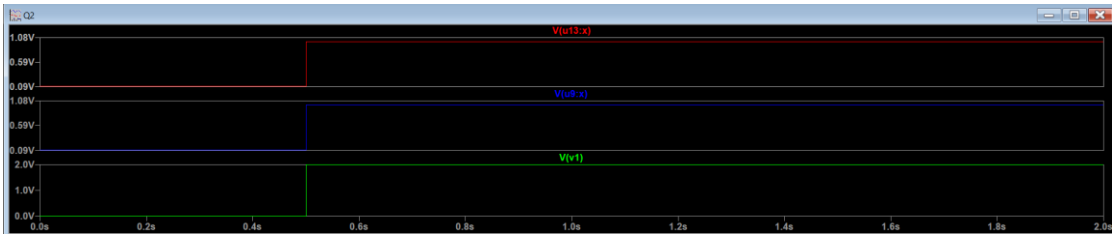
(a) Input voltage as a function of time
(b) State variable of the target memristor as a function of time
(c) State variable of the memristor that switches unintentionally as a function of time

Mark the two memristors on the array, or provide their locations in your response. Explain the results, highlighting the unintentional write.

Circuit Schematic:



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



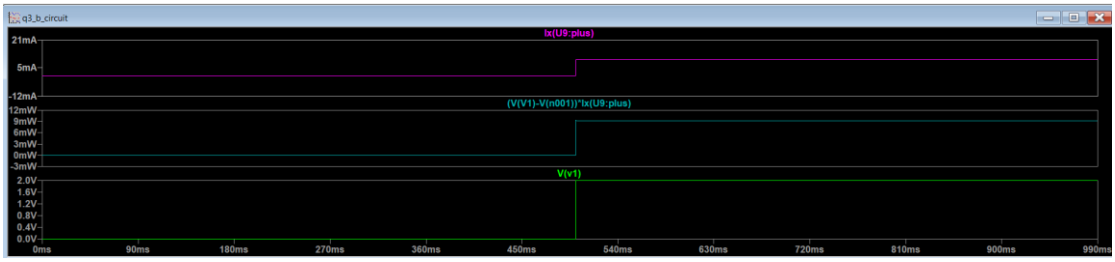
We can see from U9 state variable that we successfully wrote the value of '1' on it so we can see the transaction between 0 and 1.

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

c. Show the waveform of the power used to switch (write) the target cell from "0" to "1". Explain the shape of the waveform. How much total energy is used?

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

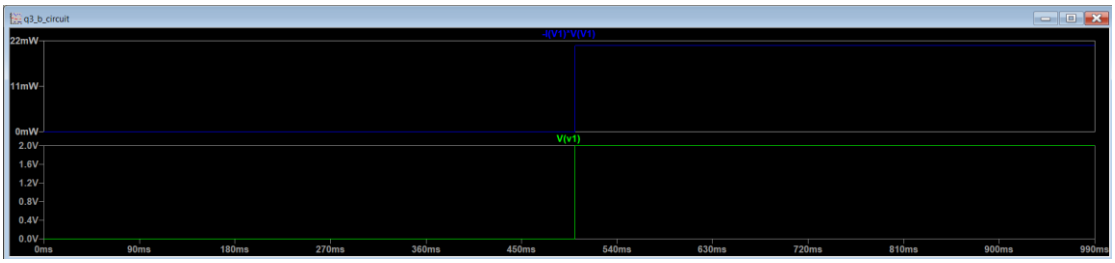


Since the input voltage applied to U9 (for write procedure) is a pulse, we got a usage of power that looks like a pulse, before the switching its 0mW and during the switching we used about 9mW of power.

Part-d

d. Show the waveform of the overall array power during the write operation. Explain the shape of the waveform. How much total energy is used?

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

e. What is the energy efficiency of the write procedure to this cell (energy that was used for the intended switching compared to the total energy)?

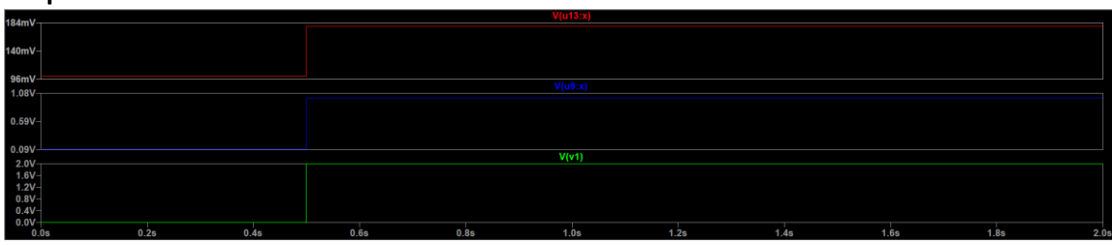
The power used to switch U9 from '0' to '1' is 10mW and the total power wasted in the circuit is 22mW (from previous sections), so the efficiency is calculated here:

$$Efficiency = \frac{P_{U9}}{P_{arr}} = \frac{10mW}{22mW} \cong 45\%$$

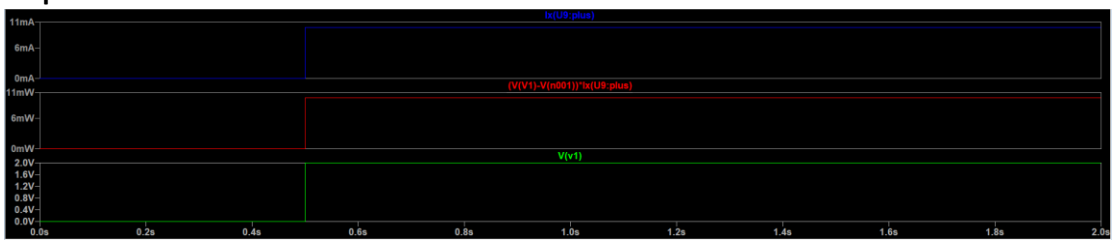
Part-f

f. Repeat (b)–(e) with the half-select mitigation method to reduce the effect of the write disturbance. Use the same two memristors as before with the same pulse duration. Is the effect of the write disturbance reduced? What is the trade-off of the half-select method?

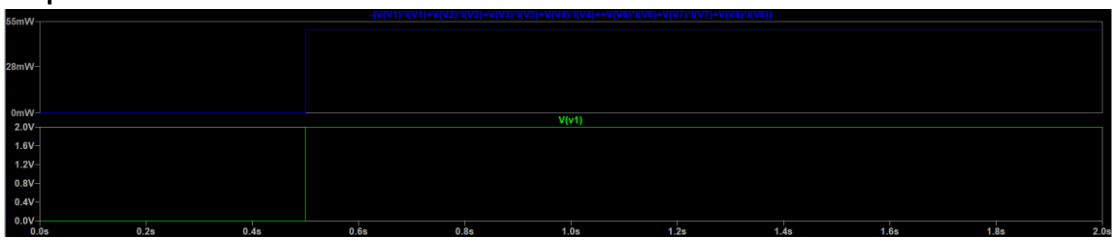
f.b-plot



f.c-plot



f.b-plot



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:

$$Efficiency = \frac{P_{U9}}{P_{arr}} = \frac{10mW}{50mW} \cong 20\%$$