Semester project 3
„Neuro-transistor simulation with memristor-based memcapacitor using LTSPICE“

Topic: Realization of the neuron functionality with a pseudo-memcapacitive transistor using LTSPICE
Task: Implement a single memristor-based memcapacitor at the gate of a MOSFET transistor in LTSPICE and find a pulse input setting which realizes the neuron firing after integration through the transistor. Extend the simulation to a 3x3 multi-channel neuro-transistor as shown in Figure 1 and test the correct functionality. Finally, sweep and find optional parameters for capacitances and pulse timing by using the LTSPICE in a loop via python.
Supervisor: Dr. Richard Schroedter, richard.schroedter@tu-dresden.de

Task description:
1. Read the paper Schroedter, et al. 2022 [1] and search relevant literature.
2. Make you familiar with the memristor model equations and check the model in LTSPICE: https://github.com/SchroedterR/Memristor
3. Create a voltage input time series (triangle shape with different amplitudes) like in Schroedter, et al. 2022 [1], and use it to verify the LTSPICE model.
4. Create a voltage pulses input time series to test the potentiation and depression curves of the model. Check whether you can reproduce the dynamic route map (DRM) as given in the paper Schroedter, et al. 2022 [1].
5. Change the LTSPICE model, such that the resistances Rs and Rp are part of the model file *.sub
6. Implement a method to run your LTSPICE code via a higher language like Python, e.g. by using the library PyLTSpice (https://github.com/nunobrum/PyLTSpice). When using PyLTSpice, you might need to fix a small error by uncommenting it: “#+ cmd_line_switches”. Use this method for all further tasks, where you need to sweep many variables.
7. Implement the memristor-based memcapacitor using a capacitor Cp in parallel to the memristor and a larger capacitor Cs in series of both. Choose realistic parameters, e.g. Cp=10pF and Cs=100pF. Plot the current-voltage curves of the resulting memcapacitive circuit for a sine wave input using different memristor states.
8. Add the gate of a NMOS at the membrane node between the two capacitors to form a neuro-transistor. You can start with the NMOS definition: “NMOS(LEVEL=3 L=26u W=94u Vto=0.2 Tox=22n TPG=0 Uo=798 THETA=42m PHI=66)”. Find out, the meaning of every parameter in the BSIM documentation (e.g. see also this documentation: https://ltwiki.org/LTspiceHelp/LTspiceHelp/M_MOSFET.htm).
9. Find a setting of voltage input pulses, e.g. 1V and 2us, where you see a significant difference in the source-drain output current (“firing”), for the case when the memristor is in ON state and for the case when the memristor is in OFF state.
10. Extend the circuit to 3 memristor-based memcapacitors at the same gate and find again a setting, where the neuron fires, when all memristors are ON and it does not fire, when the memristors are OFF.
11. Extend the circuit to a 3x3 multi-channel neuro-transistor, see Figure 1, by adding 3 NMOS transistors in series.
12. Run several sweeps of the 3x3 multi-channel neuro-transistor by varying the used capacitances and pulse timing parameters to find an optimal setting for firing and non-firing operation depending the memristor states.
13. Analyze and test how the programming/writing of the individual memristor states can be performed for the 3x3 Multi-channel neuro-transistor circuit.
14. Discuss how many of these neuro-transistors can operate in a series connection to build a spiking neural network.
15. **Comment** each individual line of code in your implementation, such that the code can be understood without reading the report. Add a **Readme file** that explains how to use your implementation code.
16. **Submit** the commented implementation **code** as one ZIP file to your supervisor **before the examination**.

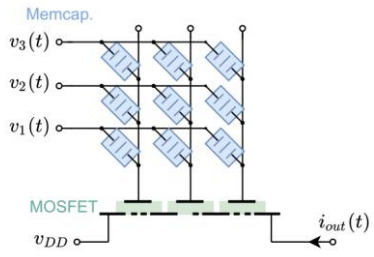

Figure 1: 3x3 Multi-channel neuro-transistor

Literature:
[1] Schroedter, et al. “SPICE Compact Model for an Analog Switching Niobium Oxide Memristor.” IEEE MOCAST, 2022, https://doi.org/10.1109/MOCAST54814.2022.9837726