
Supplementary information

Logic-in-memory based on an atomically thin semiconductor

In the format provided by the
authors and unedited

Supplementary Information: Logic-in-Memory Based on an Atomically Thin Semiconductor

Guilherme Migliato Marega^{1,2}, Yanfei Zhao^{1,2}, Ahmet Avsar^{1,2}, Zhenyu Wang^{1,2}, Mukesh Tripathi^{1,2}, Aleksandra Radenovic³, Andras Kis^{1,2*}

¹*Electrical Engineering Institute, École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland*

²*Institute of Materials Science and Engineering, École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland*

³*Institute of Bioengineering, École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland*

**Correspondence should be addressed to: Andras Kis, andras.kis@epfl.ch*

Supplementary Note 1. Operation of the Floating Gate Memory

The operation of the floating-gate transistor (FGFET) is based on the Fowler-Nordheim tunnelling of charges from the MoS₂ into the embedded metal gate. The tunnelling probability can be modulated by shifting the Fermi level of the semiconductor by the gate electric field, and it is possible due to the reduced thickness of the tunnel oxide. Because the floating-gate is composed of platinum, its high work function enables electron extraction from the floating-gate to the semiconductor. When applying a negative gate voltage, this results in the creation of positive charges in the intermediate metal. Similarly, applying a positive gate voltage, injection of an electron into the floating-gate is possible. However, due to the presence of an asymmetry between the energy gap for injection and extraction, taking into account the semiconductor's gap, electron injection is more difficult than extraction. Extended Data Figure 3a. shows the band diagram of the different materials in comprising the memory before being brought into contact. The values for electron affinity, band gap and work-function were extracted from the literature¹⁻³. Extended Data Figures 3b and c illustrate the programming mechanism for both electron injection and extraction as well as their effect on the memory threshold voltage.

Supplementary Note 2. Memory Endurance Test

Endurance test is performed by applying symmetrical 100 ms programming/erasing pulses, we estimated an endurance higher than 10^4 complete cycles. The endurance is similar to reported commercial FGMOS devices. Extended Data Figure 4. shows the same device for 7.5V and 10 V pulses.

Supplementary Note 3. Programmable Inverter

The FGFET can be seen as a device with two analog functions: the floating-gate acting as a non-volatile modulation of the channel conductance and the FET as a current-voltage converter. Therefore, when this device is integrated in an inverter configuration, the control of of conductance enables control of logical states within the inverter. The estimate for the noise margin for the inverter programmed with $V_{\text{PROG}} = 8.5$ V is shown on Extended Data Figure 5.

Supplementary Table 1. Values of noise margin and auxiliary parameters as a function of the programming voltage.

V_{PROG} (V)	V_{OL} (V)	V_{IL} (V)	V_{IH} (V)	V_{OH} (V)	NM_{L} (V)	NM_{H} (V)
8.25	0.29	0.13	0.5	0.87	-0.16	0.37
8.50	0.31	0.41	0.77	0.89	0.10	0.12
8.75	0.28	0.57	0.93	0.85	0.29	-0.08

Supplementary Note 4. Programmable NOR operation

Due to the ability of the memory to retain a programmed state (1,2,3), the functionality of logic circuits grows exponentially with the number of devices. Each new device adds 3 additional combinations of states creating a total of 3^N programmable states where N is the number of memory elements. Therefore, even a circuit with two memories can be used to implement 9 different programming states. Although there is a high number of states available, we can have an overlap of logic functionality between two different programmed states. Extended Data Figure 6a shows a two-memory NOR topology. Because of the increase functionality given by the memories, this circuit is able to represent 5 different logic functions: 11 – Constant HIGH; 33 – Constant LOW; 21 – Inverter A; 12 – Inverter B; 22 – NOR A,B. Each of the Boolean functions is one subset of the main logic function. These results are summarized in Extended Data Figure 6b.

Supplementary Table 2 summarises the programming conditions for achieving the different Boolean functions.

Supplementary Table 2. Programming Table for Programmable 2-input NOR operation

V_{PROGA} (V)	Q_1	X_A (Q)	V_{PROGB} (V)	Q_2	X_B (Q)	Logic Function	Operation Name
7.5V - 9V	2	INA	7.5V - 9V	2	INB	$Y = \text{NOT}(X_A^{(2)} + X_B^{(2)}) = \text{NOT}(\text{INA} + \text{INB})$	NOR
7.5V - 9V	2	INA	+12.5V	1	LOW	$Y = \text{NOT}(X_A^{(2)} + X_B^{(1)}) = \text{NOT}(\text{INA} + 0) = \text{NOT}(\text{INA})$	NOT A
+12.5V	1	LOW	7.5V - 9V	2	INB	$Y = \text{NOT}(X_A^{(1)} + X_B^{(2)}) = \text{NOT}(0 + \text{INB}) = \text{NOT}(\text{INB})$	NOT B
+12.5V	1	LOW	+12.5V	1	LOW	$Y = \text{NOT}(X_A^{(1)} + X_B^{(1)}) = \text{NOT}(0 + 0) = 1$	HIGH
-12.5V	3	HIGH	-12.5V	3	HIGH	$Y = \text{NOT}(X_A^{(3)} + X_B^{(3)}) = \text{NOT}(1 + 1) = 0$	LOW

Similarly, we show with the devices described in the logic-in-memory fabrication process the realization of a 3-input NOR logic gate. Extended Data Figure 7a shows the schematic of the circuit and Extended Data Figure 7b several programmable logic functions of this gate.

Supplementary Table 3. Programming table for Programmable NOR operation

Q_1	X_A	Q_2	X_B	Q_3	X_C	Logic Function	Operation Name
2	INA	2	INB	2	INC	$Y = \text{NOT}(X_A^{(1)} + X_B^{(2)} + X_C^{(3)}) = \text{NOT}(\text{INA} + \text{INB} + \text{INC})$	NOR A B C
2	INA	2	INB	1	LOW	$Y = \text{NOT}(X_A^{(1)} + X_B^{(2)} + 0) = \text{NOT}(\text{INA} + \text{INB})$	NOR A B
2	INA	1	LOW	2	INC	$Y = \text{NOT}(X_A^{(1)} + 0 + X_C^{(3)}) = \text{NOT}(\text{INA} + \text{INC})$	NOR A C
1	LOW	2	INB	2	INC	$Y = \text{NOT}(0 + X_B^{(2)} + X_C^{(3)}) = \text{NOT}(\text{INB} + \text{INC})$	NOR B C
2	INA	1	LOW	1	LOW	$Y = \text{NOT}(X_A^{(1)} + 0 + 0) = \text{NOT}(\text{INA})$	NOT A
1	LOW	2	INB	1	LOW	$Y = \text{NOT}(0 + X_B^{(2)} + 0) = \text{NOT}(\text{INB})$	NOT B
1	LOW	1	LOW	2	INC	$Y = \text{NOT}(0 + 0 + X_C^{(3)}) = \text{NOT}(\text{INC})$	NOT C
1	LOW	1	LOW	1	LOW	$Y = \text{NOT}(0 + 0 + 0) = \text{NOT}(0) = 1$	HIGH
3	HIGH	3	HIGH	3	HIGH	$Y = \text{NOT}(1 + 1 + 1) = \text{NOT}(1) = 0$	LOW

Supplementary Note 5. Universal Logic-In-Memory Unit

Our Logic-in-memory concept relies on exploring the subset operations of a principal logic operation. In Extended Data Figure 8a, we present our circuit concept that enables a comprehensive logic functionality. This structure combines both NOR and NAND operation enlarging the set of operations attainable by this structure. By adding an input and output interface, in Extended Data Figure 8b, we can further extend its functionality and logic operation. Extended Data Figure 8c-e shows how three essential logic gates are produced from

the basic unit cell. NAND and NOR are directly derived from the main Boolean equation, XOR equation can be extracted applying De Morgan's laws. Further operations are described in Supplementary Table 4.

Supplementary Table 4. Programming table for a full set of programmable operations for a 2-input logic. (^ represents the NOT operation).

N	IN ₁	Q ₁	IN ₂	Q ₂	IN ₃	Q ₃	IN ₄	Q ₄	Boolean Equation	Boolean function
1	A	1	B	2	A	3	B	3	OUT = LOW	LOW
2	A	2	B	3	A	3	B	2	OUT = $\wedge(A + B)$	NOR A,B
3	$\wedge A$	2	B	3	A	3	B	2	OUT = $\wedge(\wedge A + B)$	NOR $\wedge A$,B
4	A	2	B	3	A	3	$\wedge B$	2	OUT = $\wedge(A + \wedge B)$	NOR A, $\wedge B$
5	$\wedge A$	2	B	3	A	3	$\wedge B$	2	OUT = $\wedge(\wedge A + \wedge B)$	AND A,B
6	A	2	B	3	A	1	B	1	OUT = $\wedge(A)$	NOT A
7	A	2	B	2	$\wedge A$	2	$\wedge B$	2	OUT = $A \otimes B$	XOR A,B
8	$\wedge A$	2	B	3	A	1	B	1	OUT = A	A
9	A	3	B	2	A	1	B	1	OUT = $\wedge(B)$	NOT B
10	A	3	$\wedge B$	2	A	1	B	1	OUT = B	B
11	$\wedge A$	2	B	2	A	2	$\wedge B$	2	OUT = $\wedge A \otimes B$	XNOR A,B
12	A	2	$\wedge B$	2	A	1	B	1	OUT = $\wedge(A.\wedge B)$	NAND A, $\wedge B$
13	$\wedge A$	2	B	2	A	1	$\wedge B$	1	OUT = $\wedge(\wedge A.B)$	NAND $\wedge A$,B
14	A	2	B	2	A	1	B	1	OUT = $\wedge(A.B)$	NAND A,B
15	$\wedge A$	2	$\wedge B$	2	A	1	B	1	OUT = A+B	OR A,B
16	A	2	B	1	A	1	B	1	OUT = HIGH	HIGH

Similarly to a 2-input cell, we can intuitively expand the structure to a 3-input cell. Because a 3-input cell has a 2-input cell as a subset, it has a higher parallelism and density of operations. Supplementary Table 5 shows the combination of programming states that produces a desired logic operation.

Supplementary Table 5. Programming Table for basic block 3-input programmable operation for IN1 = IN4 = IN7 = A; IN2 = IN5 = IN8 = B; IN3 = IN6 = IN9 = C.

[illegible]

Supplementary Note 6. Performance comparison with other devices

For comparing our approach, we take similar nonvolatile and reconfigurable circuits based on emerging memories and architectures. Supplementary Table 6 summarizes the various metrics, the energy consumed per operation and reconfigurability of logic-in-memory based on various emerging technologies for different emerging memories. While both FeFETs and ReRAMs have an energy performance similar to our 2D FGMOS but with much more aggressive scaling, it is important to notice that they are based on sequential programming and reading of signals. In contrast, this programming/reading scheme requires time-multiplexing interfaces which increases the difficulty for cascading many logical stages. Therefore, the construction of large and complex structures based on FeFETs and ReRAM may have to overcome several design challenges. Related to MTJ memories, although the density of logical functions is the highest, it has similar problems to the ones mentioned for FeFETs and ReRAMs as well as orders of magnitude higher energy consumption. Finally, while our proof-of principle circuits are based on relatively large transistors, this can be overcome with further scaling⁴, also expected to result in the reduction of supply voltage and power dissipation. Furthermore, our solution addresses all the Boolean operations and it can easily be cascaded to form larger circuits.

Supplementary Table 6. Comparison table of logic-in-memory circuits from different emerging memories and architectures. Values calculated from available data in references⁵⁻⁷.

	This Work (2-input case)	MTJ⁵	ReRAM⁶	FeFET⁷
Technology node	1 μm	100 nm	50 nm	28 nm
VDD (V)	1	0.1	-0.3	1.5
Energy per operation (EOP) [J/op]	11nJ/op	25 $\mu\text{J}/\text{op}$	15nJ/op	22.5 nJ/op
Devices/Logical Operations	4/8	1/4	2/2	1/2
Input Domain	Voltage	Voltage + H field	Voltage	Voltage
Output Domain	Voltage	Resistance	Current	Voltage

Supplementary Note 7. (S)TEM Characterization of Monolayer MoS₂

The ADF-STEM image, Extended Data Figure 11, shows the clean and large field of view of the monolayer MoS₂-2H crystal lattice structure. Due to its annular dark image contrast dependency, the intensities of the atomic columns are directly proportional to the atomic number ($Z^{1.6-2}$). Thus, Mo-atomic sites seem brighter than the two sulphur atoms which are on top of each other. The inset panel in Extended Data Figure 11 shows a Fourier transform (FFT) of the real-space, which further confirms the crystalline monolayer MoS₂ structure.

The panel b shows the Wiener filtered STEM image, which is magnified from the same region as in panel a. Imaging was done at a low acceleration voltage (80 kV), which is below the threshold for electron beam induced knock-on damage of MoS₂. The zoomed-in image shows the perfect quality of monolayer MoS₂ flake. However, only a few intrinsic S vacancies can be seen in the image. The intensity profile in both panels along the line shows the peak positions of the Mo- and S-atoms, respectively. Panel c shows the simulated STEM image (also see Methods) of monolayer MoS₂.

Supplementary Note 8. TEM Characterization of the Device Cross-Section

To verify the logic-in memory device structure with the vertically stacked MoS₂ 2D channel, deposited floating gates and oxide layers, cross-sectional transmission electron microscopy was performed (see also Methods for details on lamella preparation and TEM imaging). TEM image shown in Extended Data Figure 12a shows the overview of fabricated device structure on top of the SiO₂/Si substrate. Extended Data Figure 12b shows a zoomed-in image close to the MoS₂ 2D channel and Ti/Au drain-source contact region.

Further magnified image in Extended Data Figure 12c highlights stacked layers of Pt floating and bottom gates, HfO₂ tunnel and blocking oxide layers, below the MoS₂ 2D channel. The interfaces between the channel, oxides and floating gates shows a clear and homogeneous structure. No significant voids or deformations were observed in the device.

Supplementary Note 9. Programmer

To achieve more complex circuits and structures, we have developed a closed-loop programmer to overcome device variation caused by the fabrication process, material growth and environment condition. In Extended Data Figure 9a and b, we show the hardware connections needed for programming a 4-memory structure and a 9-memory structure. The electronic switches represented in blue and green are used to individually address each floating-gate memory for programming and testing. The reconfiguration of the base circuit allow programming each memory by connecting the gate to a higher voltage output and the source to ground, and subsequently, tested by connecting the FGMOS in an inverter-like topology for verifying if its transfer curve corresponds to the desired state.

This hardware is supported by a LABVIEW code represented in Extended Data Figure 9c. For the programmer block diagram, we perform programming of each desired state to the corresponding memory. If a closed-loop option is chosen, the software will iterate until it finds an error lower than the specified. The error is calculated at 0 gate voltage in an inverter

configuration. If the difference between the output voltage of the inverter and the state value at this point is higher than the desired error, the programmer initiates a new programming iteration of the memory. For this programmer, we choose a levelling programming approach in which a triangle wave signal is applied to the FGMOS gate. After the programming, the time-dependent logic test is realized to verify the circuit behaviour. In this stage, the switches are open except the one connected to the output of the system.

Extended Data Figure 9d shows the performance of the programmer for the 9-memory structure shown in Figure 4 of the paper. In this example, we show the programmer setting the states $Q_{1-9} = 222\ 111\ 111$ to the structure for a 3-Input NAND operation. The left figure shows the transfer curve of each of the memories in the circuit. The ones set to state 1, show a constant output HIGH as desired. The first try memories where the state was set to 2, have an inverter-like transfer curve. The 3-input NAND performance is finally verified in the test block.

References

1. Ji, C. H., Lee, J. Y., Kim, K. T. & Oh, S. Y. Enhanced photocurrent in organic photodetectors by the tunneling effect of a hafnium oxide thin film as an electron blocking layer. *RSC Adv.* **9**, 29993–29997 (2019).
2. Michaelson, H. B. The work function of the elements and its periodicity. *J Appl Phys* **48**, 6 (1977).
3. Moun, M., Singh, A. & Singh, R. Study of Electrical Behavior of Metal-Semiconductor Contacts on Exfoliated MoS₂ Flakes. *Phys. Status Solidi A* **215**, 1800188 (2018).
4. Desai, S. B. *et al.* MoS₂ transistors with 1-nanometer gate lengths. *Science* **354**, 99–102 (2016).
5. Suh, D. I., Kil, J. P., Kim, K. W., Kim, K. S. & Park, W. A Single Magnetic Tunnel Junction Representing the Basic Logic Functions—NAND, NOR, and IMP. *IEEE Electron Device Lett.* **36**, 402–404 (2015).
6. Borghetti, J. *et al.* ‘Memristive’ switches enable ‘stateful’ logic operations via material implication. *Nature* **464**, 873–876 (2010).
7. Breyer, E. T., Mulaosmanovic, H., Mikolajick, T. & Slesazek, S. Reconfigurable NAND/NOR logic gates in 28 nm HKMG and 22 nm FD-SOI FeFET technology. in *2017 IEEE International Electron Devices Meeting (IEDM)* 28.5.1–28.5.4 (IEEE, 2017). doi:10.1109/IEDM.2017.8268471.