European Commission

FVLLMONTI

Call: **H2020-FETPROACT-2020-01**

Grant Agreement no. **101016776**

Deliverable D04.01

Library of optimized VNWFET-based logic cells

Start date of the project: 1st January 2021
Duration: 60 months
Project Coordinator: Cristell MANEUX - University of Bordeaux

Contact: Cristell MANEUX - cristell.maneux@ims-bordeaux.fr

DOCUMENT CLASSIFICATION

Title	Library of optimized VNWFET-based logic cells
Deliverable	D04.01
Estimated Delivery	31/08/2023 (M30+2)
Date of Delivery Foreseen	29/08/2023 (M30+2)
Actual Date of Delivery	01/09/2023 (M30+2)
Authors	Guilhem Larrieu – P2 – LAAS Sylvain Pelloquin – P2 – LAAS Alberto Bosio – P3 – ECL-INL Bastien Deveautour – P3 – ECL-INL Sara Mannaa – P3 – ECL-INL Cédric Marchand – P3 – ECL-INL **Ian O'Connor – P3 – ECL-INL** Konrad Streich – P3 – ECL-INL Cigdem Cakirlar – P6 – NLB Jens Trommer – P6 – NLB
Approver	Cristell Maneux – P1 – UBx
Work package	WP4
Dissemination	PU (Public)
Version	V1.0
Doc ID Code	D4.01_FVLLMONTI_P3-ECL-INL-20230831
Keywords	Logic circuit design, standard cells, non-volatile logic, reconfigurable logic, benchmarking, synthesis, technology, physical design automation

DOCUMENT HISTORY

Version status	V1.0
Date	01/09/2023
Document revision	V0.1 – 26/05/2023 – Initial draft V0.2 – 14/07/2023 – Document restructuring V0.3 – 21/07/2023 – Added Crossbar sections V0.4 – 28/08/2023 – Reviewed JL1 JL2 logic sections V0.5 – 31/08/2023 – Text review V0.6 – 31/08/2023 – Final figures

DOCUMENT ABSTRACT

This document describes a set of logic cells based on conventional complementary static logic, pass-transistor logic (PTL) non-volatile logic (NVL) and ambipolar logic design styles using:

- single-stack and two-stack n- and p-type JL-VNWFETs;
- single-stack reconfigurable logic blocks based on PC-VNWFETs;
- Fe-NVM integrated into a mixed two-layer stack of n- and p-type JL- and PC-VNWFETs.

Logic functions of interest include fundamental building blocks for logic and arithmetic used in CNNs (addition, multiplication): standard logic functions, XOR, crossbar, runtime reconfiguration, non-volatile properties. The library identifies single-bit logic operations, in the form of schematics, leading towards GDSII-ready layout for WP1 and behavioral models for T4.3 and WP5 annotated with timing, volume and energy estimations.

This version of D04.01 is intended to serve as a reference document, containing a detailed description of functionality and performance metrics of all logic cell designs, i.e. elementary (1-bit) logic circuits for typical boolean operations (INV1, NAND2, NOR2, XOR2) for CMOS-like and PTL design styles and for combinations of FVLLMONTI technological flavors (1-2-3 stacked gates, chemical-/electrostatic-doping, absence or presence of ferroelectric layer in gate stack).

The dataset corresponding to this deliverable and specified in D6.6 (Data Management Plan) is defined by the Argos OpenAIRE Logic standard cell library based on FVLLMONTI technology. It will be made publicly available at the end of the project via Zenodo and is intended to contain multiple representations of data obtained principally via simulation:

- reference circuit schematics using clear symbols to indicate the type and technology of VNWFET used (text files – pdf);
- physical design related representations of logic cells such as sticks diagrams, GDSII, 3D-representations (image files – png);
- Pareto-front data extracted from actual simulations using compact models and parasitic extractions illustrating quantitative tradeoffs between delay, energy consumption, volume / footprint, reliability (spreadsheet files - xlsx).

Data is organized according to the following convention:
[LogicFunction]_[Datawidth]_[DesignStyle]_[TechnologyVariant]_v[YYYYMMDD] where:

- LogicFunction represents the Boolean operation (e.g. NAND, NOR, XOR, MAJ ...);
- DataWidth represents the number of bits in the operands (1bit, 2bit, 4bit ...);
- DesignStyle represents the design approach (e.g. ComplementaryStatic, DynamicN, DynamicP, PTL ...);
- TechnologyVariant represents the technology options (e.g. JL1, JL2, AP, JLFE ...)

This information will be used mainly in WP4 to generate:

- a scaled down version of N^{2}C^2 in D04.04 scheduled for M35;
- GDSII-ready logic cell and NVM bitcell layouts in D04.02 scheduled for M35;
- a second version of the virtual scalable N^{2}C^2 in D04.05b scheduled for M44.

It will also be used in WP5 to enable architectural exploration in D5.2.
To reflect the progress in technology-related WPs as the FVLLMONTI project advances, the content of this deliverable and dataset will be updated to capture new data, opportunities and limitations according to the state of logic circuit development.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101016776.

TABLE OF CONTENT

LIST OF FIGURES AND TABLES

LIST OF ACRONYMS / GLOSSARY

D: Deliverable

M: Month of the project

P: Partner

PU: Public

V: Version

WP: Work Package

APn: Ambipolar (VNWFET) with n stacked gates ($1 \leq n \leq 2$)

APFEn: Ambipolar Ferroelectric (VNWFET) with n stacked gates ($1 \leq n \leq 2$)

CStatic: Complementary Static (logic)

CNVL: Complementary Non-Volatile Logic

DTCO: Design Technology Co-Optimization

FeFET: Ferroelectric Field Effect Transistor

JLn: Junctionless (VNWFET) with n stacked gates ($1 \leq n \leq 3$)

JLFEn: Junctionless Ferroelectric (VNWFET) with n stacked gates ($1 \leq n \leq 2$)

LUT: Look Up Table

MAC: Multiply Accumulate

N^2C^2: Neural Network Compute Cube

NN: Neural Network

PStatic: P-type Static (logic)

PTL: Pass-Transistor Logic

PPTL: P-type PTL

RFET: Reconfigurable Field Effect Transistor

RFeFET: Reconfigurable Ferroelectric Field Effect Transistor

VNWFET: Vertical Nanowire Field Effect Transistor

Xbar: Crossbar

1. Introduction

Zettascale data generation at the edge, connecting humans to machines and machines to humans, is pushing performance requirements for edge computing to ever-higher levels, while still imposing stringent constraints on energy consumption and real-time execution, as well as communication bandwidth, security and privacy [1].

This approach therefore requires more efficient hardware processors capable of performing the required computational tasks with low latency and minimal energy consumption. Needless to say, the flow to achieve this has to start from improving the performance of the basic building blocks (i.e., the transistors and the standard cells) of the circuits composing processors. Initially, it was possible to follow Moore's law [2] as regards the number of transistors integrated within such processors chips in order to achieve the target performance improvements. However, this approach is now limited by the physical properties of CMOS [3], [4] devices, reflected mainly through the increase in power consumption per chip with the continuous scaling down of the device. This has triggered the community to look toward new emerging devices and technologies that can be used in order to overcome the transistor scaling limitation.

Vertical Nanowire Field Effect Transistors (VNWFETs) are an emerging technology with significant potential to reduce footprint and consequently interconnect capacitance, thereby achieving improved energy-efficiency and being naturally compatible with advanced 3D integration approaches. As stated in its name, its operation is based on the same principle as that of the field effect transistor. The semiconductor material is a nanowire channel sandwiched between two electrodes (source and drain). This technology also relies on a Gate All Around (GAA) architecture, where the channel is surrounded by the gate stack materials in a cylindrical way [5]. Therefore, this design gives improved electrostatic control, reduced short-channel effect and better scalability opportunities.

However, while initial estimations have focused on projections and estimations, no work has so far used compact models to attempt transistor-level simulations for standard cell library characterization.

In this work, our main objective and contribution is twofold:

- to study and characterize logic cells (INV1, NAND2, NOR2 and XOR2) based on vertical nanowire transistors (with single gate JL1 or two-gate stack JL2) in terms of delay and power consumption, thus proving that this new technology can be used as the basic building block for logic gates.
- To propose novel logic and arithmetic circuits based on more advanced VNWFET technologies incorporating either ferroelectric layers for non-volatile operation, ambipolar contacts for reconfigurability, or both. Here, the goal is rather to illustrate the potential of the technology, rather than to pursue detailed characterization at this stage. The intent is to demonstrate building blocks for unconventional computing approaches and open paths for circuit-level hardware demonstrators in the latter stages of the project.

This document is organized as follows: sections 2, 3 and 4 cover the hypotheses and fundamentals of the work, covering the scope of the DTCO (Design-Technology Co-Optimization) approach, the definition of design styles and metrics used in logic design and the data management and nomenclature used throughout the project. Sections 0 and 6 cover static logic approaches (P-type to incorporate technology fabrication limitations as well as complementary for a more conventional approach and comparison), while sections 0 and 0 examines Pass-Transistor Logic and Crossbar approaches. Sections 0 and 10 examine the use of ferroelectric non-volatile devices for non-volatile logic and crossbars, while sections 11 and 12 incorporate ambipolar functionality for reconfigurability, both volatile and non-volatile.

2. DTCO scope – from technology towards circuit design

The ultimate objective of WP4 is to build a virtual prototype of the N^2C^2.

There are multiple technological variants and logic design styles that can implement the N^2C^2; and there are also multiple configurations (i.e. N^2C^2 network structures) that can execute the same operations. Each configuration is characterized by its latency (i.e. how many cycles are required to execute the operation), the area (i.e. how many N^2C^2 blocks are required) and finally the power/energy.

In order to design the N^2C^2 it is necessary to first build a library of standard logic cells, quantify relevant performance metrics and formulate as a lib file to enable synthesis

This section firstly covers the baseline device and associated compact model, then describes and captures the various technological variants explored within the project. Finally, we describe the DTCO template into which this work fits.

I. DEVICE

In this work, a vertical nanowire field effect transistor (VNWFET) [6] with a junction-less gate-all-around (GAA) architecture is adapted as shown in Figure 1. It has a homogeneously highly doped nanowire channel that is patterned into a silicon substrate, which is highly doped with boron of concentration of 10^{19} cm^{-3}. The current between the drain and source contacts is controlled by a GAA structure with an effective channel length of 14 nm. This architecture improves the electrostatic control of the channel and allows scaling problems of conventional planar transistors to be overcome.

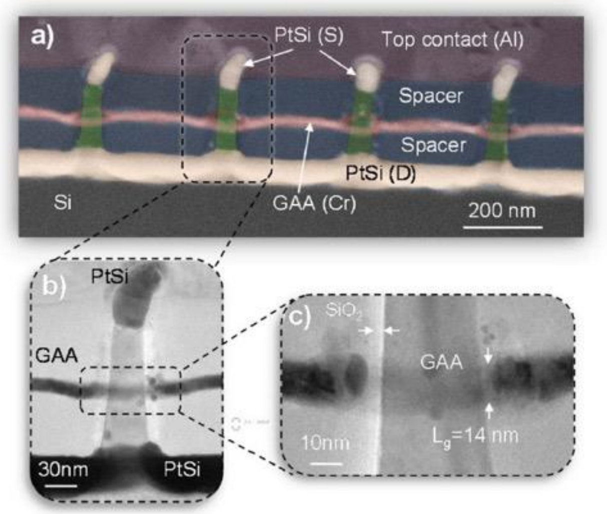


Figure 1. VNWFET device from [6]: (a) STEM image in cross section of the vertical transistor implemented in nanowire arrays, (b) single VNWFET showing its (c) gate formation

II. COMPACT MODEL

The device modeling framework for the considered VNWFET technology considers the physics of carrier transport in the junctionless architecture. In order to capture the physics of this inherently 3D technology (i.e. that of the device described in Section 2.I accurately through analytical model equations, a comprehensive SPICE simulation methodology has been developed by IMS based on a unified charge-based control model [7]. The core of the compact model accounts for an accurate description of the depletion and accumulation charges for drain current calculation in junctionless FETs. In addition, the core model is further adapted to the VNWFET technology under test through model equations for short channel effects, velocity saturation, drain-induced barrier lowering (DIBL) as well as band-to-band tunneling (BTBT) and gate-induced drain leakage (GIDL). Schottky contact formation at the source and drain access regions is also included through thermionic and leakage current branches [8][9]. Moreover, to further improve model accuracy in the low drain bias conditions, a semi-empirical field-dependent mobility model [10] has been implemented leading to very good agreement between the measurement and compact model [9]. The developed compact model and extracted model card can henceforth be exploited for circuit simulations.

III. TECHNOLOGICAL VARIANTS

The FVLLMONTI project explores several avenues of research at the technological level. In this section we detail the various options to be explored in terms of technological implementation, which are then evaluated at circuit-level to extract design data to be used in higher-level system simulations.

The baseline technology consists of the vertical nanowire field effect transistor (VNWFET) with a single gate (variant JL1, as described in Section 2.I). The design parameter at this level is essentially the number of nanowires per transistor (NW), where the pitch between nanowires is a technological parameter that will be optimized according to tradeoffs between density (footprint), inter-nanowire capacitance, reliability and yield.

Gate stacking is advantageous for logic density where multiple transistors in series are needed. This is explored at the device hardware level with a 2-gate stack (variant JL2) as well as virtually (using TCAD simulations) with a 3-gate stack (variant JL3).

Ambipolarity (electrostatic doping of the VNWFET channel) enables fine-grain logic reconfigurability but also requires a polarity gate to control the type of majority carriers in the channel. It therefore requires two gates on a single device. This can be achieved either with a 1-gate stack using a U-type configuration (variant 1AP) or with a 2-gate stack, where one of the gates is the polarity gate (variant 2AP).

The integration of a ferroelectric layer in the transistor gate stack enables non-volatile behavior (memory or configuration) directly within the transistor. This is explored at the device hardware level with a 1-gate stack (variant JLFE1) as well as virtually (using TCAD simulations) with a 2-gate stack (variant JLFE2).

Finally, the mixing of these variants is also explored virtually and used in the design of dense, fine-grain reconfigurable, non-volatile logic gates (variants 1APFE and 2APFE).

Table 1 and

Figure 2 summarize the technological variants used to build logic cell libraries in the context of the implementation of N^2C^2.

Table 1: Summary of technological variants

Variant	Gate stack	Ambipolar	Ferroelectric	hardware
JL1	1	no	no	yes
JL2	2	no	no	yes
JL3	3	no	no	no
1AP	1	yes (U)	no	yes
2AP	2	yes	no	no
JLFE1	1	no	yes	yes
JLFE2	2	no	yes	no
1APFE	1	yes (U)	yes	no
2APFE	2	yes	yes	no

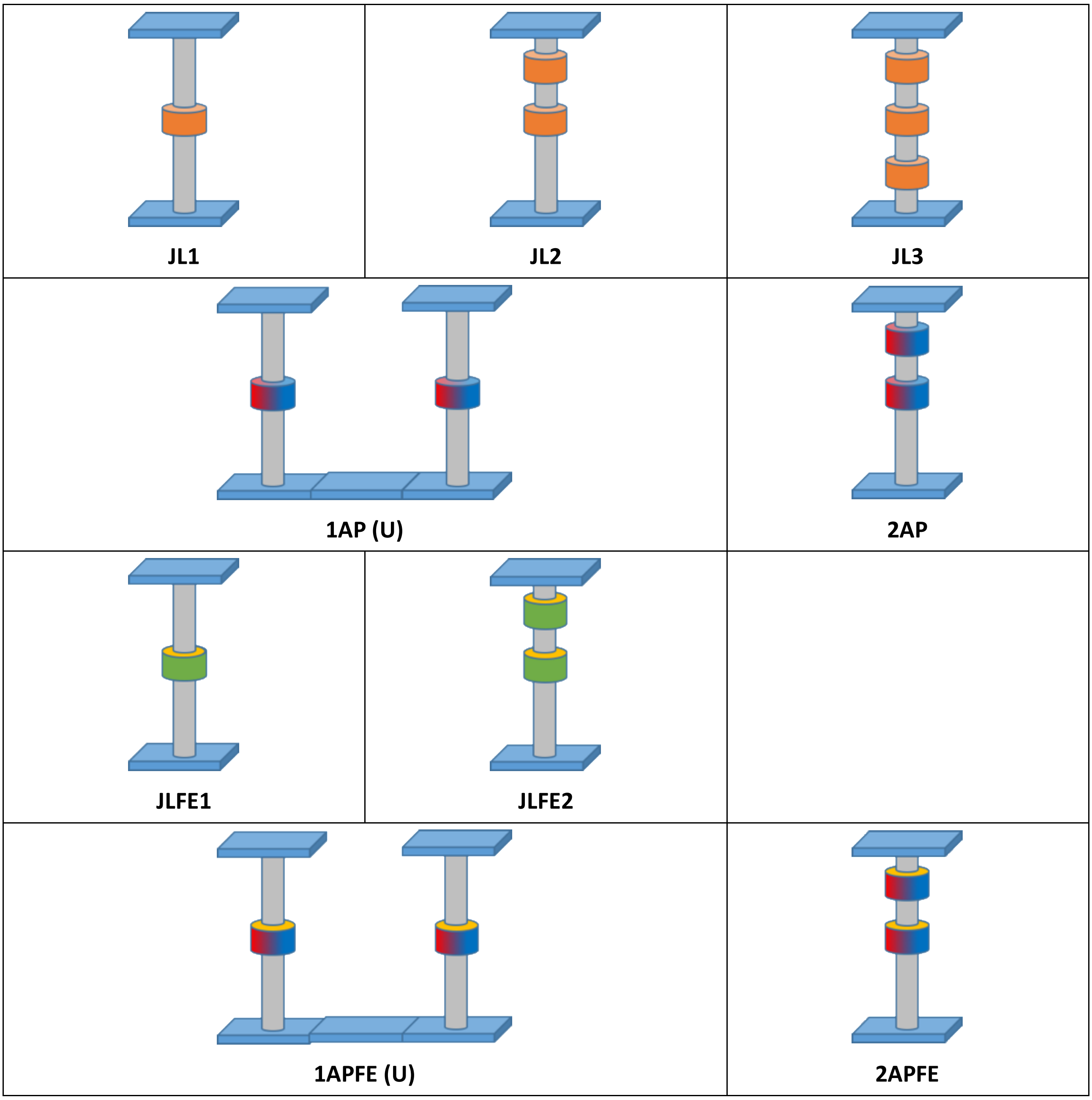


Figure 2. Representation of technological variants

IV. DTCO TEMPLATE

The number of technological variants necessitates a DTCO *template* rather than a specific DTCO flow. Further, the emerging technology nature of VNWFETs compounds the issue, since the presence of multiple facets (functional, performance, energy, temperature, reliability …) implies that one size does not fit all.

Table 2 covers the intent for all devices explored in project, including hardware prototyping and measurement ("large-scale") and virtual prototyping for prospective exploration ("scaled").

Table 2: Table indicating possible configurations of DTCO template for specific use cases

device	meas. strategy	dev. eval. method	device metrics	device model	circuit schematic	layout + pex	cct eval. method	cct. metrics	std. cell lib	synthesis	routing
large-scale JL1 large-scale JL2 large-scale 1AP		Hw exp. Data : keysight, ADS, ICCAP	I_{on}, I_{off}, V_t ...	Compact model	XX_1_Static_JL1 XX_1_Static_JL2	yes/no	ADS Spectre TCAD	EDP, footprint ...	Static_JL1 Static_JL2	ISCAS85 N2C2	manhattan octolinear
large-scale 1APFE large-scale JLFE1		Hw exp. Data : keysight, ADS, ICCAP	I_{on}, I_{off}, V_t ... $E_{write/erase}$ ...	Compact model	XX_1_NV_JLFE1	yes/no	ADS Spectre TCAD	EDP, footprint ... $E_{write/erase}$ ...	NV_JLFE1	ISCAS85 N2C2	manhattan octolinear
scaled JL1 scaled JL2 scaled JL3		TCAD	I_{on}, I_{off}, V_t ...	Compact model TCAD model	XX_1_Static_JL1 XX_1_Static_JL2 XX_1_Static_JL3	yes/no	ADS Spectre TCAD	EDP, footprint ...	Static_JL1 Static_JL2 Static_JL3	ISCAS85 N2C2	manhattan octolinear
scaled 1AP scaled 2AP		TCAD	I_{on}, I_{off}, V_t ...	Compact model TCAD model	Tile2_1_AP_1AP Tile2_1_AP_2AP	yes/no	ADS Spectre TCAD	EDP, footprint ...	AP_1AP AP_2AP	ISCAS85 N2C2	manhattan octolinear
scaled JLFE1 scaled JLFE2		TCAD	I_{on}, I_{off}, V_t ... $E_{write/erase}$ ...	Compact model TCAD model	XX_1_NV_JLFE1 XX_1_NV_JLFE2	yes/no	ADS Spectre TCAD	EDP, footprint ... $E_{write/erase}$ ...	NV_JLFE1 NV_JLFE2	ISCAS85 N2C2	manhattan octolinear
scaled 1APFE scaled 2APFE		TCAD	I_{on}, I_{off}, V_t ... $E_{write/erase}$ ...	Compact model TCAD model	Tile2_1_APNV_1APFE Tile2_1_APNV_2APFE	yes/no	ADS Spectre TCAD	EDP, footprint ... $E_{write/erase}$ ...	APNV_1APFE APNV_2APFE	ISCAS85 N2C2	manhattan octolinear

The DTCO template is shown in Figure 3. Of particular interest is the color coding, where:

- abstraction levels (which also correspond to centers of competence) are depicted in blue
- handoff points (between centers of competence) are depicted in yellow:
  - the compact model represents the interface between device developments and circuit design
  - the standard cell library (characterized) represents the interface between circuit design and functional block synthesis (this cannot be done by hand)
- physical design aspects, necessary for parasitic extraction (PEX) are depicted in green

The intent is for such a template to serve with varying levels of detail over the course of the project. Indeed, design can focus on parasitic-free functionality in a first approach, then incorporate physical design and then measurement data to refine understanding of real circuit performance. This would be first applied to the most mature (baseline) technology JL1, then branch out to other technologies as additional compact models become available. Finally, according to the system design requirements and maturity of the technology / refinement of compact models, additional detail can be added and analyzed (e.g. variability, temperature-dependent behavior).

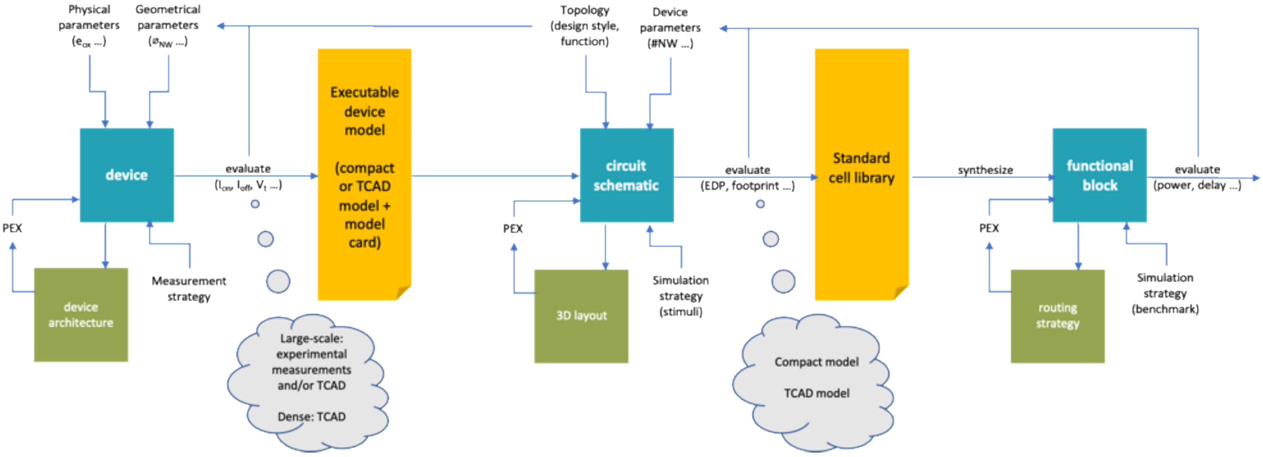

Physical parameters (e_{ox} ...)
Geometrical parameters ($\varnothing_{NW}$ ...)
device
PEX
device architecture
Measurement strategy
evaluate (I_{on}, I_{off}, V_t ...)
Executable device model (compact or TCAD model + model card)
Large-scale: experimental measurements and/or TCAD
Dense: TCAD
Topology (design style, function)
Device parameters (#NW ...)
circuit schematic
PEX
3D layout
Simulation strategy (stimuli)
evaluate (EDP, footprint ...)
Standard cell library
Compact model
TCAD model
synthesize
functional block
PEX
routing strategy
Simulation strategy (benchmark)
evaluate (power, delay ...)

Figure 3 : DTCO template

3. Logic design – rules and methodology

In this section, we describe the types of logic cells considered and the principles governing how they are designed. This includes the models and hypotheses, the physical design rules, as well as the metrics and measurement techniques.

I. LOGIC DESIGN STYLES

Several design styles can be considered for implementation of logic functions. Each design style has its own merits and shortcomings, and thus a proper choice has to be made by designers in order to provide the correct functionality. This is true in all technologies, and is given a further degree of importance with the 3D VNWFET technology proposed in FVLLMONTI due to (a) the possibility of vertical stacking, (b) the possibility of reorienting the channel direction by 90°, (c) fine-grain reconfigurability and (d) non-volatile behavior.

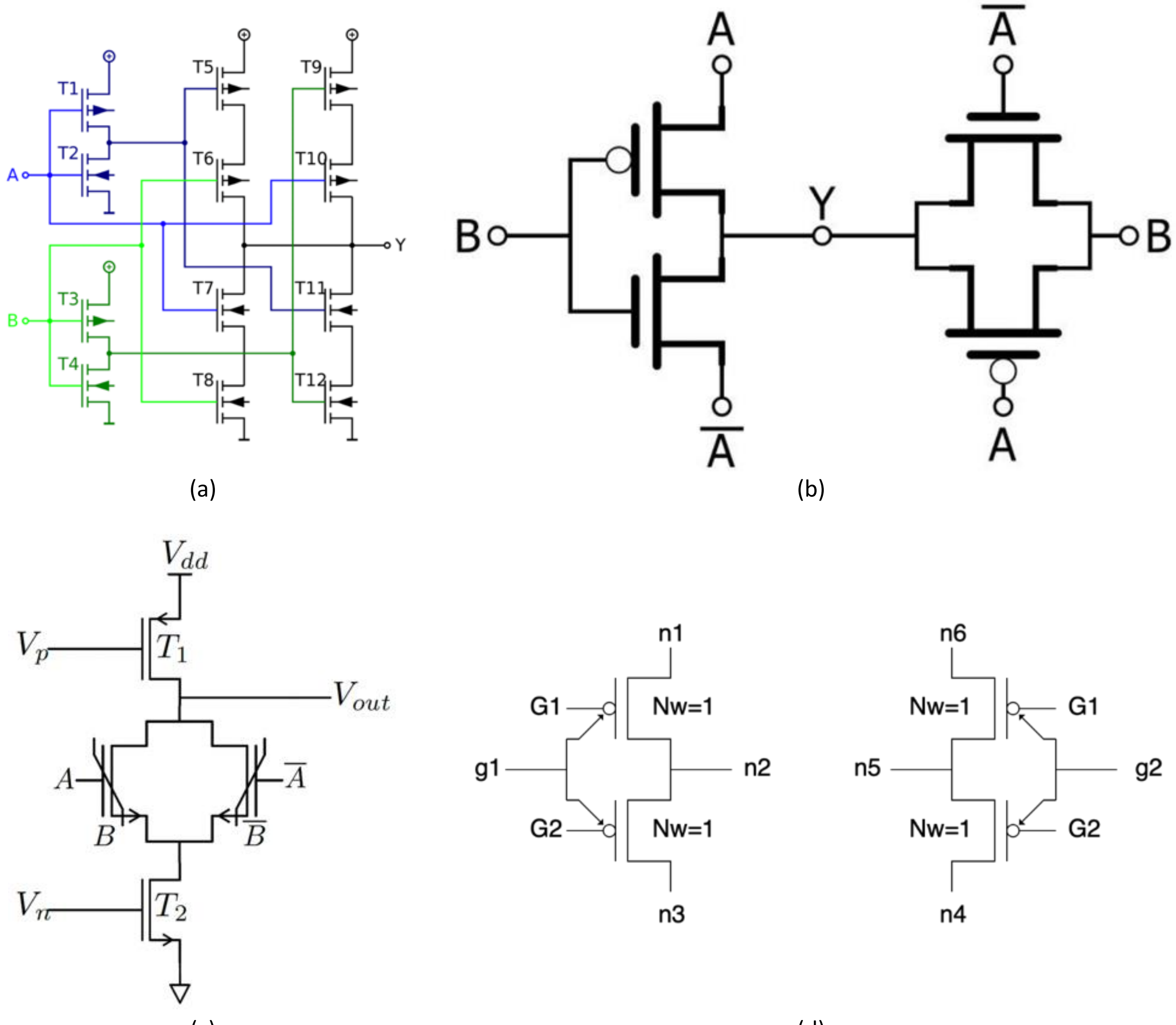
Candidate design styles considered of interest for the VNWFET technology are:

- Static CMOS-like: this is the baseline design style using pull-up and pull-down switching networks to enable propagation of the voltage of one of the two power rails (gnd='0', V_{DD}='1') based on the state of the inputs, which can only access transistors via the gate terminals.
- Pass Transistor Logic (PTL): this design style propagates data directly through transistor channels by allowing inputs to access the transistor either on the gate terminal or on one of the source/drain terminals. While this leads to more compact logic structures, the transistor channel resistance can lead to limited fanout and logic level degradation. However, as gate stacking is naturally suited to multiple transistors in series, the PTL approach presents an opportunity for exploration of such compact structures.
- Non-volatile (NV) logic: non-volatile ferroelectric transistor devices enable the storage of an operand data value within the device itself, followed by the arrival of a second operand data value. This approach is particularly well adapted to applications where one operand varies rarely (e.g. NN weight coefficient) while the other varies often.
- Ambipolar logic: electrostatic doping enables a single hardware device to achieve either n-type or p-type switching functionality, and therefore leads to attractive solutions for fine-grain reconfigurability, flexible hardware substrates and dense, regular architectures. The combination of ambipolarity and ferroelectric behavior also enables the non-volatile storage of a configuration value. A generic tile-based structure

presents opportunities for exploration. Particular points of concern will be leakage current and interconnect limitations.

Examples of these four types of logic design style are given in

Figure 4.



(a)

(b)

(c)

(d)

Figure 4 : Examples of logic design styles to be explored in view of N^2C^2 implementation. Complementary static CMOS design style – XOR2 gate (a). PTL design style – XOR2 gate (b). Non-volatile design style – dynamic XOR2 gate (c). Ambipolar design style – reconfigurable tile (d).

II. PERFORMANCE METRICS – STANDARD CELL LIBRARY CHARACTERIZATION

Specifications for the design space (i.e. performance metrics to be extracted from N^2C^2 hardware measurements and/or simulations) are detailed in this section. The design spaces are populated with data in the form of Pareto Fronts and originating from circuit-level simulations using technological variants described in the previous section in D04.05b.

Table 3: Summary of performance metrics

Metric	Detail	Units	Comments
V_{DD}_nom	Nominal operating supply voltage	V	1.0
V_{DD}_min	Min operating supply voltage	V	
T_{op}_nom	Nominal operating temperature	°K	300
T_{op}_max	Max operating temperature	°K	
N_{ctrl}	Number of control inputs		Only relevant for reconfigurable gates
N_{in}/N_{out}	Number of data inputs / outputs		
T_{prog}	Programming time per function	s	Measured under nominal operating conditions Only relevant for non-volatile gates
E_{prog}	Programming energy per function	J	Measured under nominal operating conditions Only relevant for non-volatile gates
L_{ex}	Execution latency per function	s	Measured under nominal operating conditions For single Boolean functions, this is equivalent to the propagation delay t_d
E_{ex}	Execution energy per function	J	Measured under nominal operating conditions
Thr_{ex}	Execution throughput per function	bits/s	Measured under nominal operating conditions For single Boolean functions, this is simply 1/L_{ex}
Vol	Volume	μm^3	
Err_count	Reliability	errors / operation	Measured under both nominal (V_{DD}_nom, T_{op}_nom) and worst-case operating conditions (V_{DD}_min, T_{op}_max)

1. *Logic circuit performance metrics*

The fundamental tool to be used for logic cell library characterization is transistor-level transient electrical simulation (e.g. Spectre). Analysis of the input and output voltage waveforms allows extraction of delay data, while analysis of the supply current waveform enables extraction of power and energy data.

All values are calculated for the combination of all the defined input transitions and C_{load} values, and are then stored in a library characterization file for future reference and use by logic synthesis tools.

Resolution is a key parameter and represents a tradeoff between accuracy and simulation/extraction time. Here we consider the following:

- 4 values for t_r/t_f and 4 values for C_{load}. This leads to 16 simulations per cell.
- 4 drive strengths per logic function. In conjunction with the previous line, this leads to 64 simulations per logic function

1) Transitions

In the case of a 2-input cell:
For each input: there are 2 stable states (0,1) and 2 transitions (↑, ↓)
This works out to 12 input combinations:

- 2 stable states on input 1 * 2 transitions on input 2 = 4
- 2 transitions on input 1 * (2 transitions + 2 stable states) on input 2 = 8

The following sequence is a general 2-input stimulus that can be used to extract all necessary information:
… 00 01 10 01 00 10 11 10 00 11 01 11 …

It should be noted that some combinations can be eliminated if the output is stable (i.e. transition couples 2 lines in truth table with identical output).

Table 4: Exhaustive sequence of combinations of stable states and transitions for standard cell library performance extraction for 2-input logic gates

A	B	XOR2	NAND2
0	↑	↑	- (1)
↑	↓	- (1)	- (1)
↓	↑	- (1)	- (1)
0	↓	↓	- (1)
↑	0	↑	- (1)
1	↑	↓	↓
1	↓	↑	↑
↓	0	↓	- (1)
↑	↑	- (0)	↓
↓	1	↑	↑
↑	1	↓	↓
↓	↓	- (0)	↑

2) Timing values

In the liberty file, a timing and power consumption description is detailed at the level of the output pins of each cell. These matrices reflect the values related to output transitions affected by a single input transition. It is considered here that only one input can trigger one transition at a time, thus leading to output transition.

For the correct description of these values, a time sensing parameter should be identified to set how the input transition affects the output transition. There are three possibilities:

- negative unate: falling (resp. rising) input leads to rising (resp. falling) output
- positive unate: falling (resp. rising) input leads to falling (resp. rising) output
- non unate: No generalized unateness relation between inputs and output

As propagation delay is a key metric for logic circuits, it is important to measure this property in individual logic gates accurately and with clearly defined methods and test conditions. In this subsection, we study the delay in the output transition for the aforementioned logic circuits as compared to the related input transition(s), where the timing is composed of the cell delay and rise/fall time of the output. For a single input-output pair, the delay is measured as:

$$t_{delay} = t|_{V_{out}=V_{DD}/2} - t|_{V_{in}=V_{DD}/2}$$

Since there are two types of transition (rise, fall) that can occur at the output, we measure both transition delays. In addition, we trigger the output rising (resp. falling) time which we consider as the time needed by the output voltage to rise (resp. fall) from 0.1V_{DD} (resp. 0.9V_{DD}) to 0.9V_{DD} (resp. 0.1V_{DD}):

$$t_{rise} = t|_{V_{out}=0.9V_{DD}} - t|_{V_{out}=0.1V_{DD}}$$

$$t_{fall} = t|_{V_{out}=0.1V_{DD}} - t|_{V_{out}=0.9V_{DD}}$$

3) Power and energy values

The power consumption of a logic cell is mainly a contribution of two parts: the static (or leakage) power and the dynamic power. The leakage power is due to the cell leakage current when there is no transition on any input i.e. the cell is in a static state. It is usually defined as:

$$P_{leak} = I_{leak}V_{DD}$$

The dynamic power consumption is itself composed of two contributions: (i) the switching power that is due to the charging of the load capacitors and (ii) the short circuit power (internal power) that corresponds to the transition time during which both pull-up and pull-down networks are conducting.

For this part, we defined the total energy consumption per transition to be:

$$E_t = V_{DD}\int_{t_i}^{t_f} |I_{DD}|\, dt$$

In our case we considered that t_i corresponds to the time at which V_{out}= $0.1V_{DD}$ and t_f corresponds to the time at which V_{out}= $0.9V_{DD}$. Indeed, the switching energy is usually dependent on C_{load} and can be defined as:

$$E_{switching} = C_{load}\int_{V_i}^{V_f} |V_{out}|\, dV$$

We consider that V_i corresponds to $0.1V_{DD}$ and V_f corresponds to $0.9V_{DD}$.

Finally, the intrinsic energy consumption (essentially due to the leakage and short-circuit currents, excluding energy consumption required to drive C_{load}) is defined as:

$$E_{int} = E_t - E_{switching}$$

2. *Physical design metrics*

Physical design metrics enable understanding of the cost of logic cells and their implementation, as well as giving valuable information concerning the actual resistive and parasitic loads of the cells, as seen by the circuits driving them. While these have not been extracted in the current version of the deliverable, work is underway to do so.

1) Compactness and connectivity

- Footprint: x,y,z data of each logic cell / functional block
- Pin positions: these must be fixed and represent information required by routing tools e.g. power rail routing as described in D3.3

2) Parasitic extraction

Resistive and capacitive parasitic elements depend on the physical design and layout of each logic cell and are extracted either predictively (before fabrication) through TCAD analysis of the cell, or after fabrication through experimental de-embedding and measurement techniques.

Information of interest covers:
- Parasitics to ground
- Parasitics between wires
- Parasitics between VNWFET nanowires

4. Logic cell library data management

The formulation of measured logic cell timing/energy data is necessary for two objectives:
- The first objective is to enable the use of data (within libfiles) for logic synthesis. Indeed, it is impossible to scale to a complex functional block manually - this requires automated tools and therefore data informing the tools about the characteristics of the logic cells. In this work, we focus on the Liberty format for the characterization of standard cell libraries.
- The second objective is to generate a dataset for use by scientific community. This is important for the legacy of the project legacy and to maximize its impact.

I. LIBERTY TECHFILE – STANDARD CELL LIBRARY CHARACTERIZATION

As logic synthesis is mandatory for the implementation of complex and optimized circuits in terms of performance and power consumption, it is important to have a standard cell library that characterizes the basic logic cells, thus enabling the synthesis process. In this section, we explain our approach to build such a library.

For this aim, we adopted the standard Liberty file format. This file format contains timing and physical information about the target standard cells needed for the library generation. In our approach, we also adopted the nonlinear delay model where the gate delay is affected by the input slew rate and load capacitance, and is not directly proportional to the input transition time. Thus, a lookup table (LUT) should be defined to store the values of timing and power consumption based on the input slew and the load capacitance. For example, defining X input transition values and Y C_{load} values will result in an X by Y LUT that contains all the possible combinations between the defined values.

The liberty file is built in a hierarchical way over three levels: the library level, the cell level and the pin level. Each level stores the required information to be used by the synthesis optimization.
- At the library level, all the information related to the delay model, unit attributes (for the timing, voltage, current, resistance, leakage current and capacitive loads), the operating conditions, the high and low slew thresholds for both the inputs and outputs and the lookup tables (LUT) templates must be specified.
- Each logic cell should have a dedicated description at the cell level. This description starts from the cell leakage power and area to the input/output pins.

- At the pin level, for each pin in the logic cell, it is necessary to define its type (input or output), the logic function (for output pins) and input capacitance (for input pins). Then the timing and internal power LUT data should be included based on the effect of each of the input pins as described in the subsection below.

II. ZENODO DATASET – STANDARD CELL LIBRARY LEGACY

As previously indicated, our objective is to generate a dataset for use by researchers in the scientific community. This is important for the legacy of the project legacy and to maximize its impact.

The target dataset describes, through multiple representations (schematics, sticks diagrams, vsticks (vertical sticks) diagrams, Pareto front data, sized structures: 3D-rendered + GDSII files), elementary (1-bit) logic circuits for typical boolean operations (INV1, NAND2, NOR2, XOR2, DFF) for CMOS-like and PTL design styles and for combinations of FVLLMONTI technological flavors (1-2-3 stacked gates, chemical-/electrostatic-doping, absence or presence of ferroelectric layer in gate stack).

This data is collected to validate VNWFET-based logic circuit designs with a clear identification of technological hypotheses and design style. Validation is intended both in terms of technological feasibility (DTCO - particularly using physical design related representations) and in terms of their use in N^2C^2 architectures to achieve target performance metrics (STCO - particularly the Pareto-front data).

The types of data collected are:

- reference circuit schematics using clear symbols to indicate the type and technology of VNWFET used.
- physical design related representations of logic cells such as sticks diagrams, vertical (vsticks) sticks diagrams, GDSII, 3D-representations.
- Pareto-front data extracted from actual simulations using compact models and parasitic extractions illustrating quantitative tradeoffs between delay, energy consumption, volume / footprint, reliability

The data is in the following format:

- Text files (.docx, .pdf) to document circuits.
- Multimedia files (images - .png) to illustrate sticks and vsticks diagrams, 3D-representations

🟦	M1 - metal1 (lower)
🟪	M2 - metal2 (upper)
🟥	G1 - gate 1
🟫	G2 - gate 2
🟩	n-type VNW
🟨	p-type VNW
▨	ambipolar VNW

Figure 5. Color code to be used in sticks diagrams and physical design data

Sticks diagrams also allow first estimations of cell area (footprint), expressed in units of F^2. F expresses the minimum lithographic feature of the densest process layer, taken in this document to be equal to the half-pitch dimension of the M1 layer.

- Spreadsheet files (.xlsx) to hold simulation results, Pareto-front data.

Data is organized according to the following convention:
[LogicFunction]_[Datawidth]_[DesignStyle]_[TechnologyVariant]_v[YYYYMMDD]
where:

- LogicFunction represents the Boolean operation (e.g. NAND, NOR, XOR, MAJ ...);
- DataWidth represents the number of bits in the operands (1bit, 2bit, 4bit ...);
- DesignStyle represents the design approach (e.g. CStatic, PPTL, CNVL ...);
- TechnologyVariant represents the technology options (e.g. JL1, JL2, JLFE1, AP1 ...)

5. P-type static logic (PStatic) cell library – JL1, JL2

In this section, we consider the generation of the main logic functions using only P-type VNWFETs. This constraint originates from the devices available with the technological resources at LAAS, for which our objective is to generate hardware for experimental measurement. This gives rise to the P-type static logic design style (PStatic), and is explored with one- and two-levels of gates on the VNWFET (JL1, JL2).

I. PSTATIC CELL DESIGN

1) Functional analysis

P-type FETs usually make up the "Pull-Up Network" (PUN) branch of static logic gates.

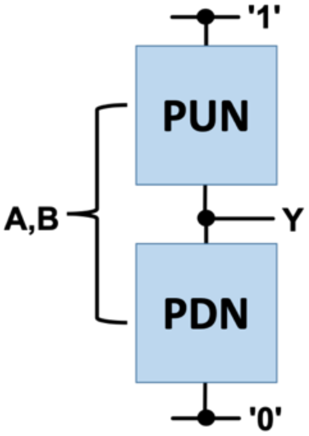


Figure 6. Conventional structure for logic gates including two dual networks to connect the output to a logic rail: a Pull-Up Network (PUN) for logic '1' and a Pull-Down Network (PDN) for logic '0'

In Figure 6, '1' and '0' represent the supply rail voltages used to represent logic '1' and logic '0'. The voltage used to represent logic '1' is always greater than that used to represent logic '0'. Usually, this is positive logic i.e. '1' = V_{DD} = a non-zero, positive value (e.g. 1 V) and '0' = ground = 0 V. However, it is also possible to define negative logic i.e. '1' = ground = 0 V and '0' = -V_{DD} = a non-zero, negative value (e.g. -1 V).

While lateral FETs are symmetrical and channel contacts can be used interchangeably as source or drain, this is not the case with the vertical structure since the source contact is pad connected to the bottom silicide. For a p-type structure, the voltage applied to the source is greater than or equal to that applied to the drain, and this usually implies that V_s = '1' (or, using the previous definitions 1 V for positive logic, 0 V for negative logic).

If we use the p-type FET structures as the PUN and an external resistance as the PDN, then we implement a p-type static (PStatic) logic gate.

NOR2_1_PStatic_JL2

The schematic shown in Figure 7 represents a 2-input NOR gate using the PStatic logic design style.

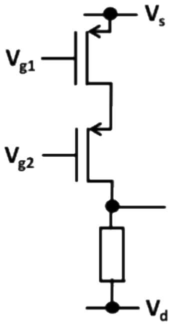


Figure 7. NOR2_1_PStatic_JL2 cell schematic

By manipulating inputs V_{g1} and V_{g2}, this circuit can also be shown to implement several functions:

- if V_{g1} = data input A and V_{g2} = data input B, then Y = !(A+B) - NOR2 operation
- if V_{g1} = !A and V_{g2} = !B then by De Morgan's theorem Y = A.B - AND2 operation
- if V_{g1} = A (resp. 0) and V_{g2} = 0 (resp. B) then Y = !A (resp. !B) - INV operation

In the following tables, we assume that V_{g1} and V_{g2} are binary data inputs using the same logic representation as the supply voltages for logic '1' and logic '0'.

Table 5: NOR2 truth table and corresponding gate drive voltage levels for positive- and negative-logic conventions

Logic truth table – NOR2			Positive-logic '0' = 0 V / '1' = 1 V			Negative-logic '0' = -1 V / '1' = 0 V		
A	B	Y	V_{g1}	V_{g2}	V_d	V_{g1}	V_{g2}	V_d
0	0	1	0 V	0 V	1 V	-1 V	-1 V	0 V
0	1	0	0 V	1 V	0 V	-1 V	0 V	-1 V
1	0	0	1 V	0 V	0 V	0 V	-1 V	-1 V
1	1	0	1 V	1 V	0 V	0 V	0 V	-1 V

XOR2_1_PStatic_JL2

With two such structures in parallel (source and drain contacts connected) we can also implement the PUN branch of an XOR2 logic gate, as shown in Figure 8. Note that this structure excludes the inverters required to generate !A and !B (useful for test purposes).

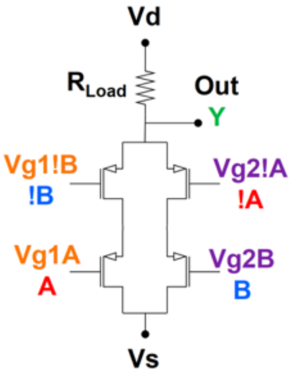


Figure 8. XOR2_1_PStatic_JL2 cell schematic, without inverters (test-observable version)

In this structure:

- if V_{g1A}=A and $V_{g1!B}$=!B, and $V_{g2!A}$=!A and V_{g2B}=B, then Y=A⨁B – XOR2 operation
- if V_{g1A}=A and $V_{g1!B}$=B, and $V_{g2!A}$=!A and V_{g2B}=!B, then Y=!(A⨁B) – XNOR2 operation

Table 6: XOR2 truth table and corresponding gate drive voltage levels for positive- and negative-logic conventions

Logic truth table – XOR2			Positive-logic '0' = 0 V / '1' = 1 V					Negative-logic '0' = -1 V / '1' = 0 V				
A	B	Y	V_{g1A}	$V_{g1!B}$	V_{g2B}	$V_{g1!A}$	V_d	V_{g1A}	$V_{g1!B}$	V_{g2B}	$V_{g1!A}$	V_d
0	0	0	0 V	1 V	0 V	1 V	0 V	-1 V	0 V	-1 V	0 V	-1 V
0	1	1	0 V	0 V	1 V	1 V	1 V	-1 V	-1 V	0 V	0 V	0 V
1	0	1	1 V	1 V	0 V	0 V	1 V	0 V	0 V	-1 V	-1 V	0 V
1	1	0	1 V	0 V	1 V	0 V	0 V	0 V	-1 V	0 V	-1 V	-1 V

The complete version of the XOR2 gate, with additional inverters to generate the complemented versions of data inputs A and B, has also been designed and is shown in Figure 9.

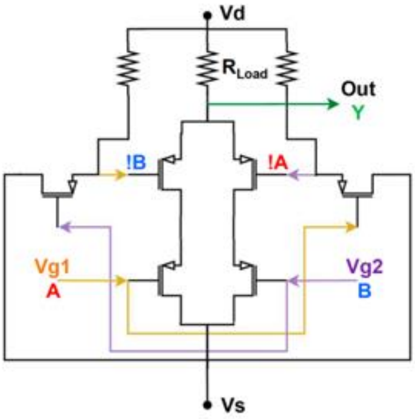


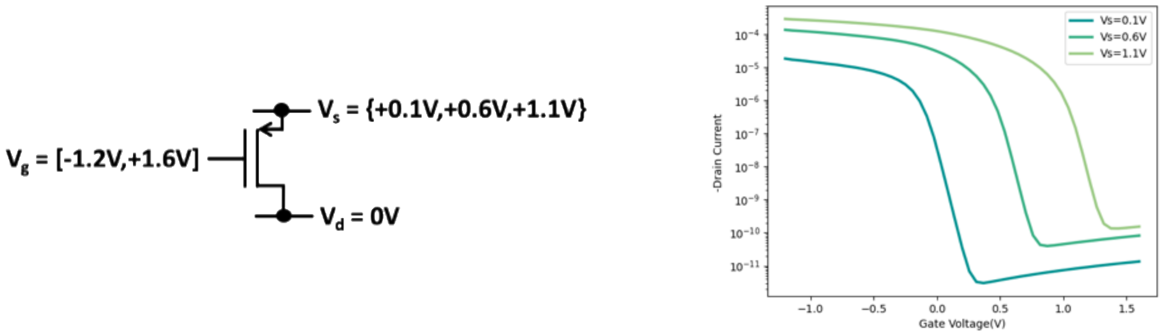
Figure 9. XOR2_1_PStatic_JL2 cell schematic, with inverters (complete version)

2) Electrical analysis – two series-connected p-type VNWFETs

In order to compare positive and negative logic, INL ran simulations on a single JL1 device based on the IMS compact model under several sets of operating conditions.

To evaluate **positive logic** performance, as shown in Figure 10:

- p-type drain terminal V_d is tied to 0V as the lowest potential in the circuit
- p-type source terminal V_s is tied to either +0.1V, +0.6V or +1.1V (positive biasing)
- p-type gate terminal V_g is swept from -1.2V (device should be on) to +1.6V (device should be off)



(a) (b)

Figure 10. p-type VNWFET I_{sd}/V_g measurement under positive logic bias conditions. (a) Circuit setup. (b) Simulation results.

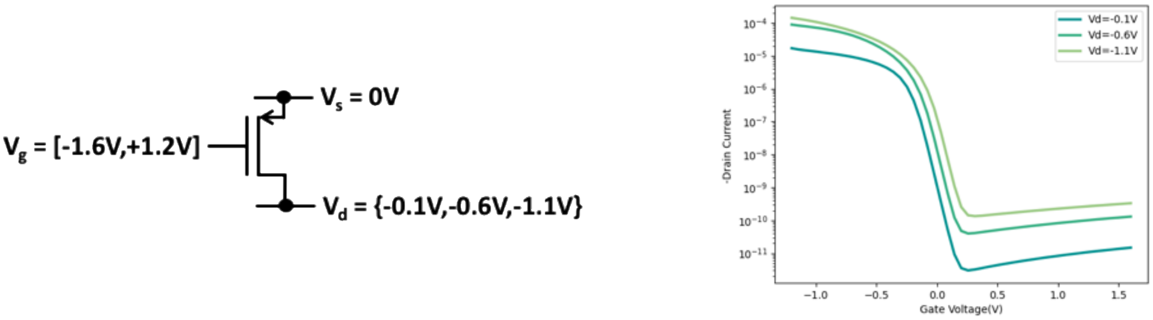
We analyze the case where $V_s = 1.1V$ and $V_d = 0V$, i.e. $V_{sd} = V_s - V_d = 1.1V$

- For point ON+:
  - $V_s = 1.1V\ ,V_g = 0V \Rightarrow V_{sg} = 1.1V > \ |V_{th}|$
  - $I_{ON+} \approx 4.97\ uA$
- For point OFF+:
  - $V_s = 1.1V\ ,V_g = 1.1V \Rightarrow V_{sg} = 0V < \ |V_{th}|$
  - $I_{OFF+} \approx 6.17\ nA$

From these points we can conclude that for positively biased devices, $\frac{I_{ON}}{I_{OFF}} = 8.06 \times 10^2$.

To evaluate **negative logic** performance, as shown in Figure 11:

- p-type source terminal is tied to 0V as the highest potential in the circuit
- p-type drain terminal is tied to either -0.1V, -0.6V or -1.1V (negative biasing)
- gate terminal is swept from +1.2V (device should be off) to -1.6V (device should be on)



(a) (b)

Figure 11. p-type VNWFET I_{sd}/V_g measurement under negative logic bias conditions. (a) Circuit setup. (b) Simulation results.

We analyze the case where $V_s = 0V$ and $V_d = -1.1V$, i.e. $V_{sd} = V_s - V_d = 1.1V$

- For point ON-:
  - $V_s = 0V\ ,V_g = -1.1V \Rightarrow V_{sg} = 1.1V > \ |V_{th}|$
  - $I_{ON+} \approx 5.12\ uA$
- For point OFF-:
  - $V_s = 0V\ ,V_g = 0V \Rightarrow V_{sg} = 0V < \ |V_{th}|$
  - $I_{OFF-} \approx 2.03\ nA$

From these points we can conclude that for negatively biased devices, $\frac{I_{ON}}{I_{OFF}} = 2.53 \times 10^3$.

From this analysis, we can see that the functionality of the device is similar, irrespective of the biasing approach (positive of negative). It also appears that while the absolute values of on-current are comparable for both positive and negative logic biasing conditions, the I_{on}/I_{off} ratio is significantly better under negative logic biasing conditions.

3) Electrical analysis – negative-logic NOR2_1_PStatic_JL2

Following individual device analysis, we also evaluated through simulation the PStatic NOR2 logic gate (Figure 7) using negative logic conditions. Hence the supply voltage values were set as Vs=0V, Vd=-1V; data inputs A,B varied between two values {-1V, 0V} to represent logic levels {0, 1} respectively, with pulse lengths of 5ns and with a PDN resistance of 1MΩ. The simulation results are shown in Figure 12.

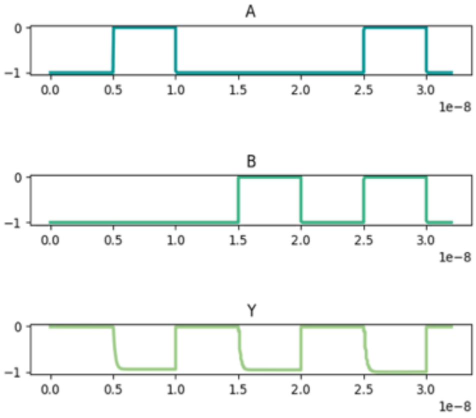


Figure 12. Simulation of NOR2_1_PStatic_JL2 under negative-logic conditions

6. Complementary static logic (CStatic) cell library – JL1, JL2, JL3

Efficient logic circuit design requires optimal design of the basic building blocks, i.e. standard logic cells. In this section, we demonstrate the operating functionality of the following basic logic cells using the complementary static logic design style while taking advantage of the vertical nanowire technology and exploring the impact of the variation of critical parameters. Figure 13 depicts the transistor level schematics of the INV1, NAND2, NOR2 and XOR2 Boolean gates, where the formalism OPn indicates the Boolean operation OP and the number of inputs n. In this section, the logic design style is complementary static (CStatic) and the underlying technology is JL1, JL2 or JL3.

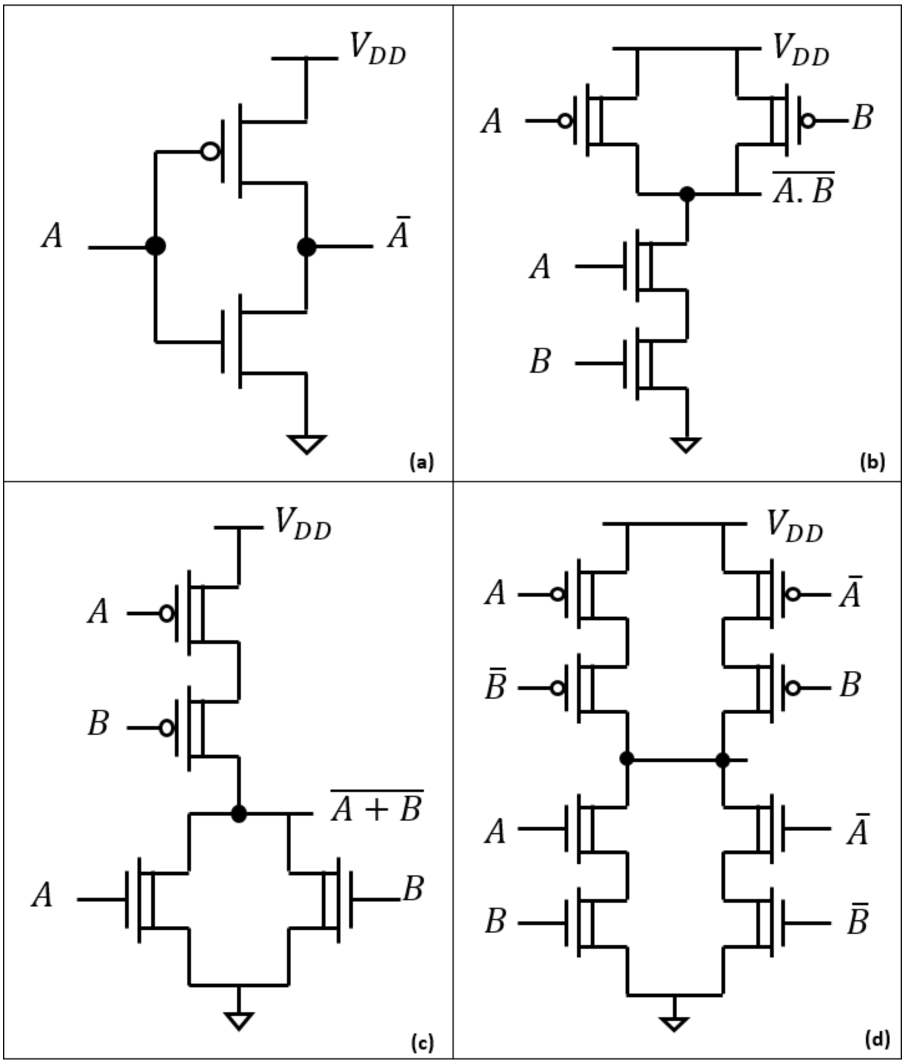


Figure 13. Schematic of logic cells studied: (a) INV1, (b) NAND2, (c) NOR2, (d) XOR2

I. SIMULATION FLOW

In our transistor-level electrical simulations we used the compact model supplied by IMS, implemented as a Verilog-A executable model. The gate physical length L_g and the nanowire (NW) diameter d_{nw} are both parameters which are determined by the fabrication process. Throughout this work, and based on experimental devices, the values of these parameters are set to L_g = 18nm and d_{nw} = 22nm respectively. Based on this, as well as the definition of other fixed model parameters, we investigated the number of nanowires for p-type and n-type transistors to be used, as the only design parameter remaining to enable the optimization of device and circuit performance. The first essential step in this work is to verify the functionality of the n-type and p-type VNWFET devices themselves through DC-sweep simulation in order to ensure that they have the expected I_{DS}/V_{GS} characteristic behavior, as illustrated in Figure 14.

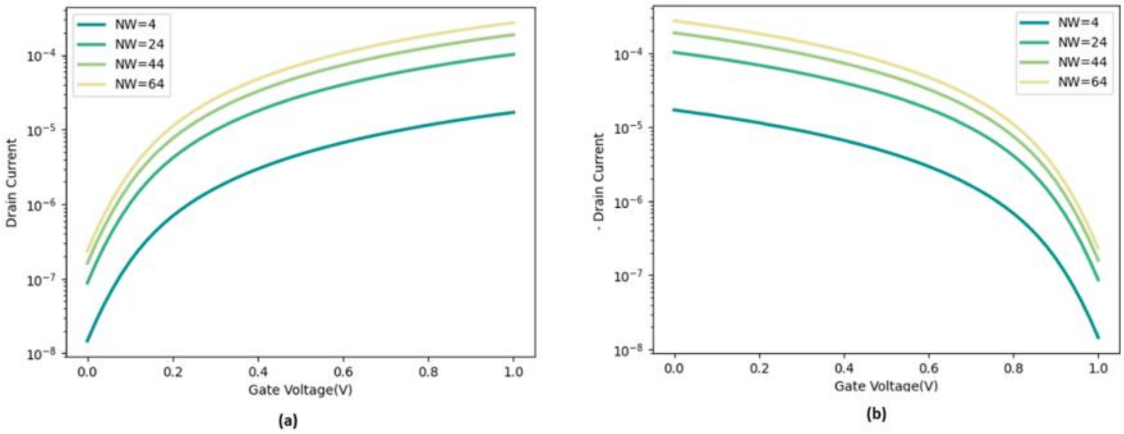


Figure 14. Variation of drain-source current I_{DS} with gate-source voltage V_{GS} and number of nanowires per VNWFET NW of (a): n-type VNWFET and (b): p-type VNWFET

Subsequent to this verification, we simulate an elementary inverter gate shown in Figure 13(a) in order to determine the correct ratio between n-type and p-type NW (number of nanowires per VNWFET) values to obtain an optimal midpoint voltage. In this work, we also chose a range of values of NW for n-type transistors that allows us to study the behavior of the device under different drive strengths. For each value, we run DC-sweep simulations while varying the gate voltage from 0 V to V_{DD} and assessing the output behavior for varying numbers of nanowires for the p-type device. We find that for all the defined NW values, a **ratio of 1** between the number of NWs of p-type and n-type will give us an optimal midpoint voltage at half V_{DD}, which leads to balanced noise margins and well-matched rise and fall times of the cell as shown in Figure 15.

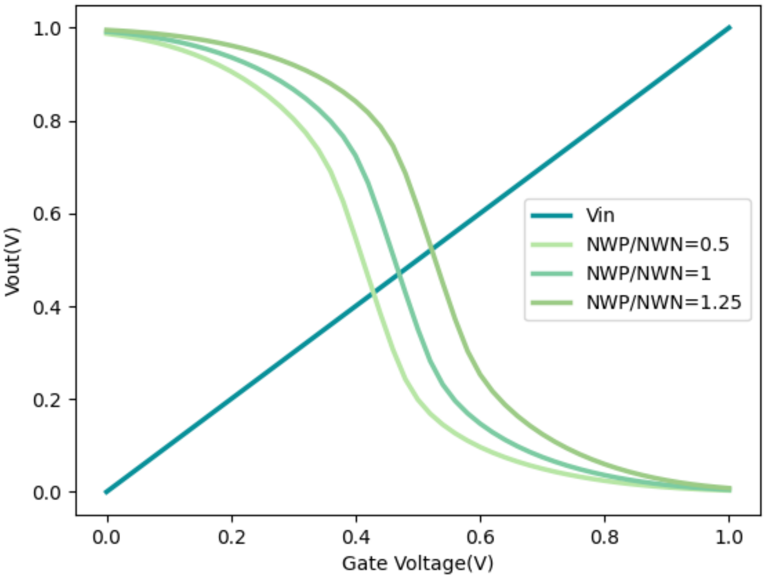


Figure 15. Variation of inverter output voltage as function of the input voltage variation between 0 V and V_{DD} (1 V) for different ratios between the number of nanowires (NW) used for p-type and n-type transistors. It can be clearly seen that a ratio of 1 between NW of p-type and n-type devices achieves optimal midpoint voltage at half V_{DD}.

It is important of course to note that this ratio depends on the set of parameters used in the simulation. For this study, the chosen values for the number of NWs of the n-type device of the inverter were 4, 24, 44 and 64. The corresponding number of NWs for the p-type device is chosen according to the above described methodology. For the other cells, and in order to achieve drive strengths equivalent to that of the inverter, we redefine the number of nanowires as shown in Table 7 (doubling the number of NWs for two in-series transistors).

Table 7: Number of nanowires used for simulations

Logic Cell	n-type NW values	p-type NW values
INV1_1_CStatic_JL1	4, 24, 44, 64	4, 24, 44, 64
NAND2_1_CStatic_JL1	8, 48, 88, 128	4, 24, 44, 64
NOR_1_CStatic_JL1	4, 24, 44, 64	8, 48, 88, 128
XOR_1_CStatic_JL1	8, 48, 88, 128	8, 48, 88, 128

After defining the sets of NW parameters, we identify the simulation limitations of the executable model and adjust our simulations accordingly. Then a detailed study of the static and dynamic behavior of all cells was carried out, as detailed in the following subsections.

1. *Verification of the logic functionality*

In order to verify the dynamic behavior of each cell and quantify the performance metrics, we run a transient simulation to extract the targeted matrices.

The first step is to define all the possible input transition(s) and the corresponding output transitions for each cell (Table 8). This step allows us to verify that the targeted cells have the required logic behavior in all possible cases. It also allows us to find the delay and power values for all these defined cases which are required for the liberty file as explained in Section 3.

Table 8: All possible output transitions (of the logic cells under study) as affected by the input transitions

Logic Cell	Input Transition	Output Transition	Timing sense
INV1	↑	↓	Negative Unate
	↓	↑	
NAND2	↓1	↑	Negative Unate
	1↓		
	↓↓		
	↑1	↓	
	1↑		
	↑↑		
NOR2	↓0	↑	Negative Unate
	0↓		
	↓↓		
	↑0	↓	
	0↑		
	↑↑		
XOR2	↓1	↑	Non-unate
	↑0		
	1↓		
	0↑		
	↑1	↓	
	↓0		
	1↑		
	0↓		

II. CSTATIC_JL1 CELL BENCHMARKING

After defining the general methodology on which our characterization is based, we perform the study for each logic gate while considering some of the model limitations we faced. The simulations for each cell were carried out at two levels: the static and the dynamic performance.

1. *Static Behavior*

Leakage current per nanowire is about 3.64 nA (i.e. I_{off}). We found also that the I_{on} of the devices increases in a linear way with the increase of the number of nanowires used thus having a value of 4.26 µA per nanowire. These two values give us a I_{on} / I_{off} ratio of 1.17×10^3.

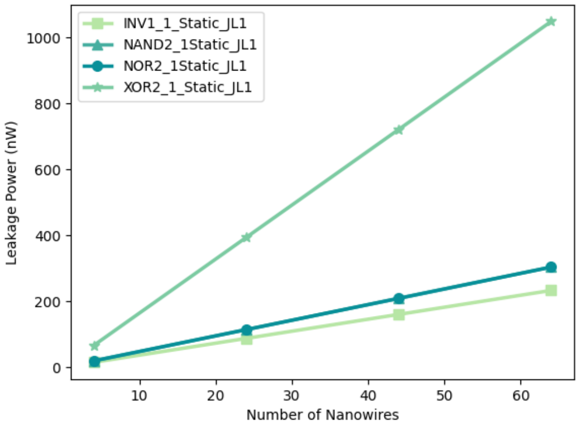
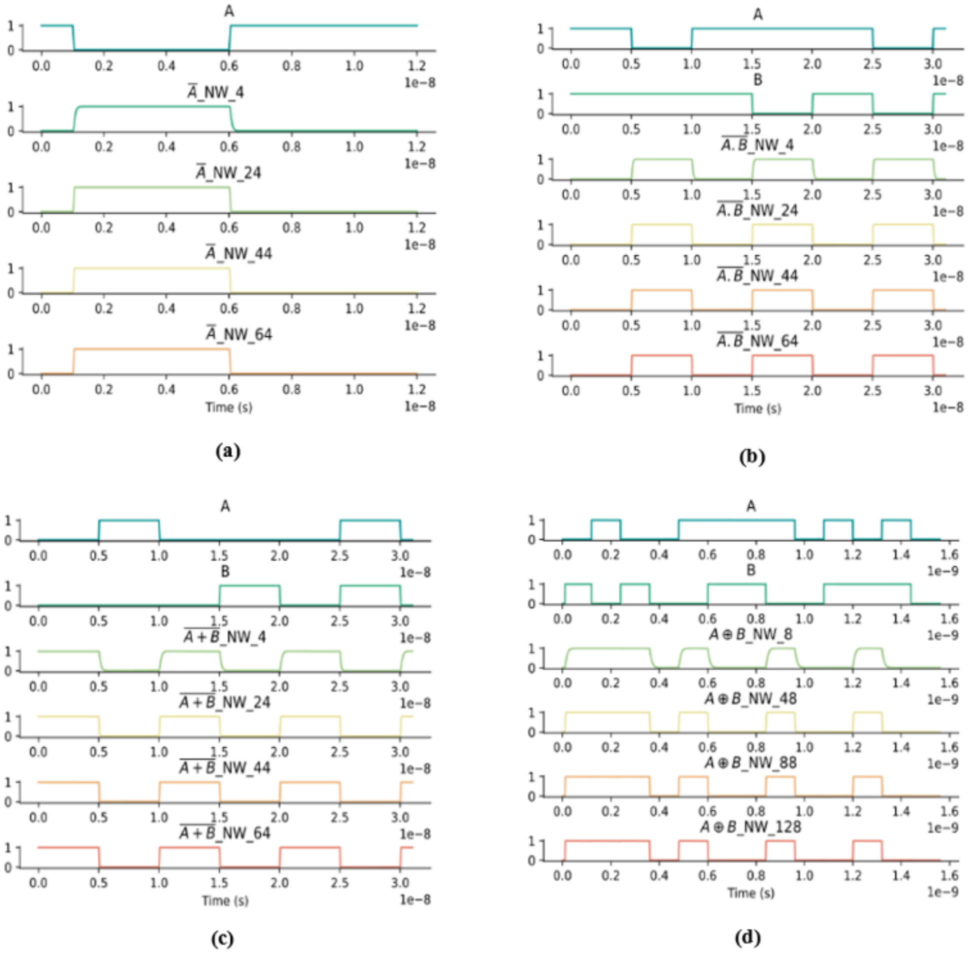


Figure 16. Leakage power in different logic cells, varying the number of nanowires per VNWFET

2. *Dynamic Behavior*

For these simulations, we used V_{DD} = 1V and an input rising-falling time of 50 ps. The C_{load} values are 160 aF, 1440 aF, 2640 aF and 900 aF for INV1, NAND2, NOR2 and XOR2 cells respectively. These values are chosen according to the model characteristics.

Figure 17 shows simulation results using all possible input transitions leading to an output transition, for all considered logic cells. All cells have the expected logic behavior. As expected, increasing the number of nanowires used per transistor leads to higher speed output transitions.



(a)

(b)

(c)

(d)

Figure 17. All possible input transitions applied to (a) INV1 (b) NAND2 (c) NOR2 and (d) XOR2 for varying NW and corresponding outputs.

1) Delay and rise and fall times

After verifying for correct output behavior, we study the detailed delay in the output transitions with respect to that of the input causing it. For example, Figure 18 clearly shows that an increase in the number of NWs used per VNWFET will lead to a decrease in the propagation delay. This behavior was similar for all the logic cells under study. Figure 19 shows a detailed view of the possible input transitions leading to either output rising or falling transition and the delay (as defined previously) in each case for NAND2 cells. It illustrates that the delay is dependent on the input(s) leading to this transition. It also verifies again that a decrease in the delay will be observed with the increase in the number of NWs. This exact detailed study was done for all other cells and showed the same behavior. The average values of these transitions are presented in Figure 18 for simplicity.

2) Energy Consumption

Another important metric for cell characterization, the dynamic energy consumption per transition was triggered. An example of the XOR2 cell is shown in Table 9. We can clearly see that with the increase of the number of nanowires an increase in the energy consumption is observed. The difference between the rising and falling energies is due to the charging of the load capacitance. This behavior was observed with the other cells as well.

Table 9. Dynamic energy consumption of XOR2 cells with varying numbers of nanowires and per transition as affected by each input

		Nb of nanowires of n-type			
Output transition	**Inputs transitions**	**8**	**48**	**88**	**128**
↑	↓1	742.15	1288.80	1934.20	2533.20
	↑0	735.99	1015.30	1241.40	1386.40
	1↓	760.66	1498.00	2367.60	2484.60
	0↑	749.75	1401.90	1234.90	1369.80
↓	↑1	27.48	712.37	1781.00	1800.90
	↓0	66.64	270.73	495.41	667.48
	1↑	19.54	545.67	1168.80	1785.20
	0↓	17.66	244.50	484.49	671.71

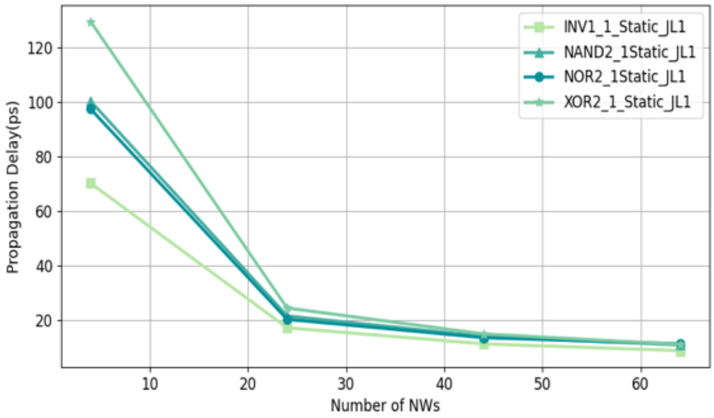


Figure 18. Average propagation delay of the four logic cells under study as function of the number of nanowires used per VNWFET device.

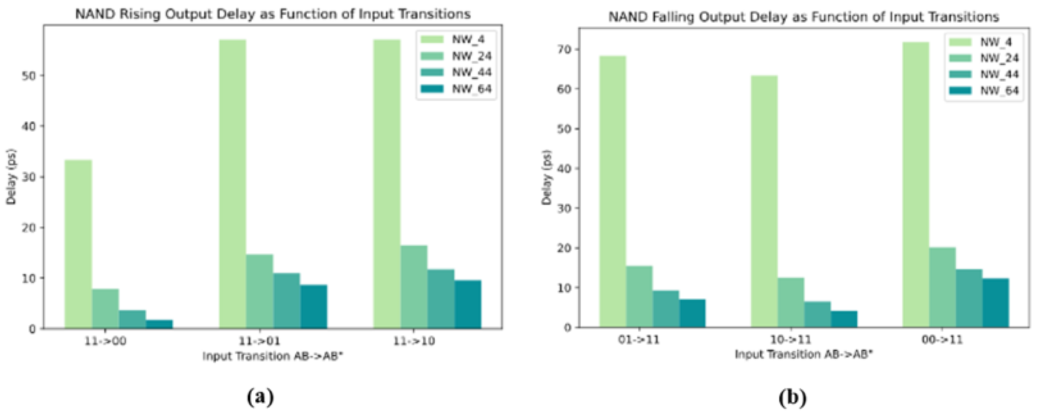


Figure 19. Delay in the output (a): rising, (b): falling transitions of a NAND logic cell with respect to the corresponding input(s) transition(s)

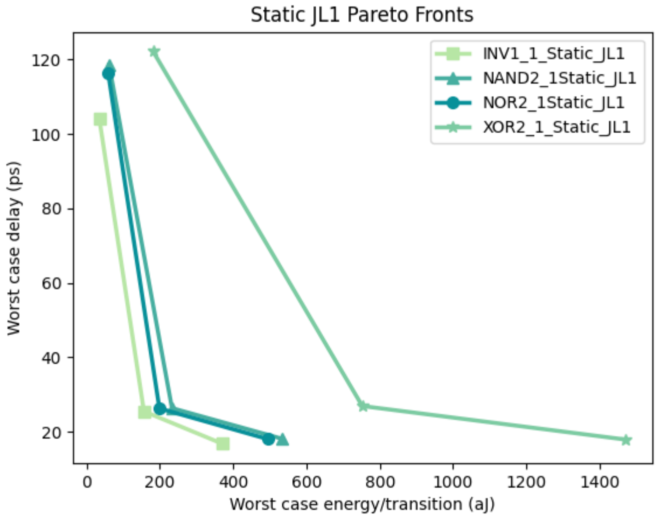


Figure 20. Pareto fronts for all standard logic cells in JL1 (wc delay vs wc energy/transition)

7. P-type pass-transistor logic (PPTL) cell library – JL1, JL2

I. SERIES-CONNECTED P-TYPE TRANSISTOR PAIR

It is also of interest to test multiplexing (MUX) capabilities of structures composed of two transistors in series. This can also be seen as generalized cases of pass-transistor logic (PTL) gates. Schematically, this is represented as shown in Figure 21.

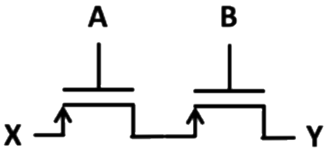


Figure 21. Series-connected P-type transistor pair schematic

Here, Y=X if A=0 and B=0 (notwithstanding probable logic degradation if X=0).

II. 2-INPUT MULTIPLEXER (MUX2)

A MUX2 gate, as shown in Figure 22, implements four such structures in parallel with a common drain contact, and 4 separate source contacts for inputs X_{00}, X_{01}, X_{10}, X_{11}; only one of which will be selected by a specific combination of A and B.

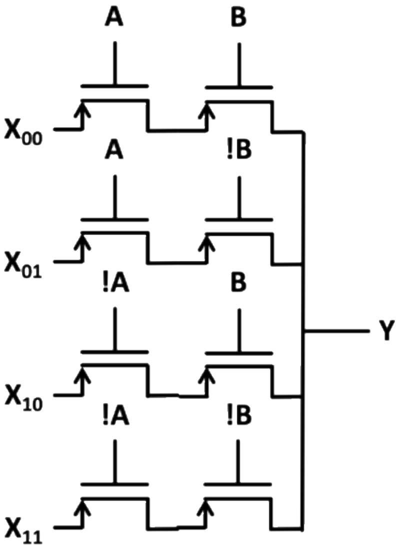


Figure 22. MUX2 schematic based on series-connected P-type transistor pair structures

8. Crossbar (Xbar) cell library – JL1, JL2

Crossbar architectures are often used in the design of hardware accelerators for machine learning algorithms (e.g. convolutional and deep neural networks such as CNNs and DNNs), which typically involve a large number of multiply-accumulate (or dot-product) operations. A crossbar is thus intended to perform arithmetic operations, usually using analog representations of data inputs on rows, weight inputs on conductance values at each row-column intersection, and output sums of products on columns.

I. XBAR OPERATING PRINCIPLE

The principle of operation of a crossbar unit can be demonstrated through a 2x2 example as shown in Figure 23.

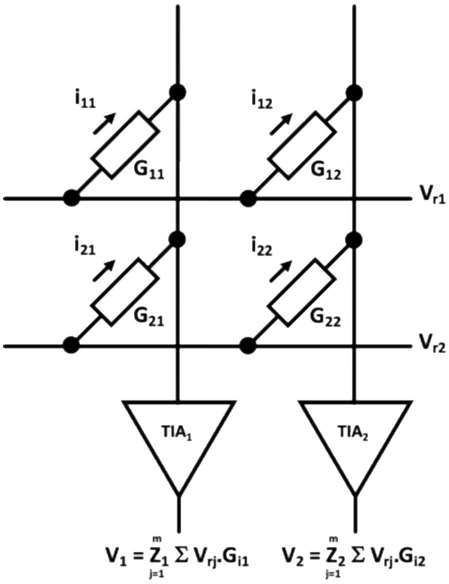


Figure 23. 2x2 example case of crossbar structure

Assuming no line access resistance in row/column lines, and virtual ground at the transimpedance amplifier input, we can write:

$$\begin{aligned} i_{11} &= V_{r1} \cdot G_{11} \\ i_{12} &= V_{r1} \cdot G_{12} \\ i_{21} &= V_{r2} \cdot G_{21} \\ i_{22} &= V_{r2} \cdot G_{22} \end{aligned}$$

As currents are summed over the column line, we obtain:

$$\begin{aligned} V_{c1} &= Z_1(i_{11} + i_{21}) = Z_1(V_{r1} \cdot G_{11} + V_{r2} \cdot G_{12}) \\ V_{c2} &= Z_2(i_{21} + i_{22}) = Z_2(V_{r1} \cdot G_{12} + V_{r2} \cdot G_{22}) \end{aligned}$$

Where Z_n represents the transimpedance gain of each column TIA (usually considered identical).

We have thus implemented, on each column, a sum of products (dot product) operation on 2x2 inputs:

- Data inputs [V_{r1}, V_{r2}] and weights [G_{11}, G_{12}] for column 1
- Data inputs [V_{r1}, V_{r2}] and weights [G_{21}, G_{22}] for column 2

This principle can be extended to any number of lines and columns. However, line resistance will increase with scale, and will degrade the linearity of each operation.

II. JL1 XBAR

In the case of the JL1 technology, each conductance G_{xx} is implemented by a single transistor controlled by an independent gate voltage (Figure 24). As no NV memory is involved, this would be adapted to direct matrix-vector multiplication operations.

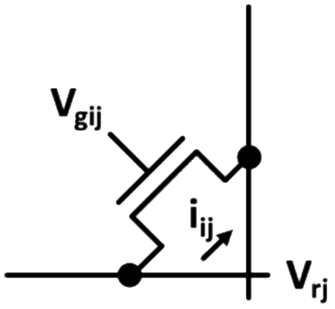


Figure 24. JL1 implementation of conductance G_{xx}

The device should be in triode mode to obtain a drain-source current I_{DS} (=i_{ij}) expression varying with device data input V_{DS} (=V_{rj}) and weight input V_{GS} (=V_{gij}). The small-signal model for a field effect transistor is shown in Figure 25.

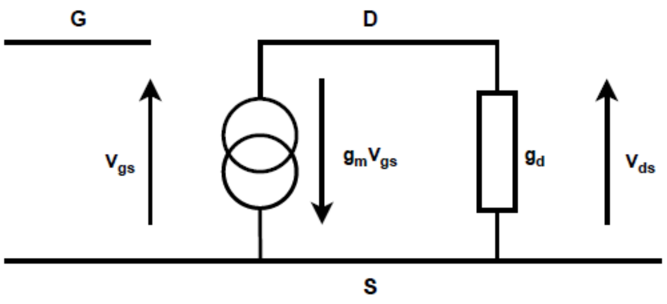


Figure 25. First order small-signal model for field effect transistor

In the triode mode, the expression for the drain-source current is given by:

$$I_{DS} = \beta\left[V_{GS} - V_T - \frac{V_{DS}}{2}\right]V_{DS} = \underbrace{\beta(V_{GS} - V_T)V_{DS}}_{product\ term} \underbrace{-\frac{{V_{DS}}^2}{2}}_{error\ term}$$

where β represents the transconductance parameter = μ Cox W / L. This expression for drain-source current leads to small-signal quantity expressions for:

- Transconductance: $g_m = \frac{dI_{DS}}{dV_{GS}} = \beta V_{DS}$
- Output conductance: $g_d = \frac{dI_{DS}}{dV_{DS}} = \beta[V_{GS} - V_T - V_{DS}]$

such that the overall small-signal current generated is given by:

$$i_{ds} = \underbrace{\beta V_{DS} v_{gs}}_{product\ term} \underbrace{+\beta[V_{GS} - V_T - V_{DS}]v_{ds}}_{error\ term}$$

In both large-signal and small-signal representations, we can observe a product term between drain and gate voltages as well as a nonlinear error term linked to the square of the drain voltage. In the triode operating region, the drain voltage is significantly lower than the gate voltage and can be minimized to a certain extent to minimize nonlinearity – however, a limit exists due to signal to noise ratio. A tradeoff therefore exists between the relative values of the product result with respect to the nonlinearity (i.e. relative impact of the error term) and noise.

III. JL2 XBAR

For the JL2 technology, each conductance G_{xx} is implemented by two transistors in series, each controlled by independent gate voltages (Figure 26).

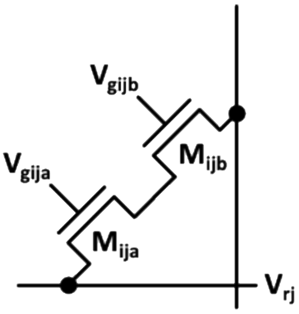


Figure 26. JL2 implementation of conductance G_{xx}

This configuration can be used in two ways:

- As an analog cascode structure, where M_{ija} requires a constant reference gate voltage and acts as a cascode transistor, while M_{ijb} employs the weight input gate voltage. V_{rj} is still used to represent the data input. This structure is intended to improve linearity.
- As a stochastic multiplier, where both gate voltages hold a digital stochastic time-varying representation of the weight input on V_{gija} and the data input on V_{gijb} (these assignments are interchangeable), while the row voltage is set to logic '1'. This approach effectively implements an AND operation using the two series transistor structure, which is the fundamental logic operation required to multiply two inputs in the stochastic data representation domain. The output is also a stochastic stream of current pulses representing the sum of products over the whole column, which should be integrated in the analog domain and normalized to the duration of the stochastic time window.

9. Complementary non-volatile logic (CNVL) cell library – JLFE1, JLFE2

The majority of ferroelectric circuits have previously focused on memory bitcells. Ferroelectric non-volatile logic circuits such as [11] are of significant interest for weight-stationary hardware such as CNNs. Prior art is based on logic using n-type FETs in the pull-down network with the ferroelectric oxide integrated within the gate stack, with either a resistive load in the pull-up network or dynamic logic approaches [12]. Recent advances in the implementation of p-type devices, including VNWFETs and ambipolar devices, now make it possible to explore ferroelectric circuits based on a complementary ferroelectric device technology.

We first describe the fundamentals governing the operation of FeFET devices. We then explore performance quantification, including programming protocols and methodology for extraction of programming latency and energy. NV-Xbar will be discussed in section 10.

I. FUNDAMENTALS

The operating principle behind most FeFET circuits is V_T shifting: depending on the polarization state of the ferroelectric layer, the transistor threshold voltage can be increased or decreased.

For example, N-channel FETs are considered non-conducting when the applied gate voltage is lower than their threshold voltage, which depends on oxide thickness and gate length, as well as technological parameters. Re-polarizing the ferroelectric layer releases charges in the transistor gate, which is a floating node. These charges in turn help to establish conduction in the channel, requiring a lower input voltage to reach the threshold voltage on the floating transistor gate if the previously applied polarization pulse was positive, or a higher one if it was negative.

V_T modulation can affect the behavior of transistors in a non-volatile manner. To create FeFET-based circuits, it is necessary to choose both the starting threshold voltage, and the voltage shift amount.

1. *n-type FeFET analysis*

Let us examine the switching conditions for the n-type FeFET:

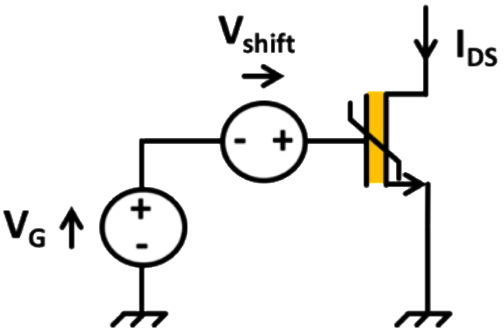


Figure 27. Bias circuit for analysis of on-conditions of n-type FeFET

1) Expression of on-conditions

The on-condition for the n-type transistor is:

$$V_{GS} = V_G + V_{shift} > V_T$$

Which can also be expressed as

$$V_G > V_T - V_{shift}$$

V_{shift} is defined by the programming pulse:

If V_{prog+} then $V_{shift} > 0$ V
If V_{prog-} then $V_{shift} < 0$ V

As shown in Figure 28, a positive (resp. negative) programming voltage will thus result in a lowering (resp. raising) of the threshold voltage. Note that we consider that during programming, the channel potential should be set to 0 V, i.e. $V_d = V_s = 0$ V.

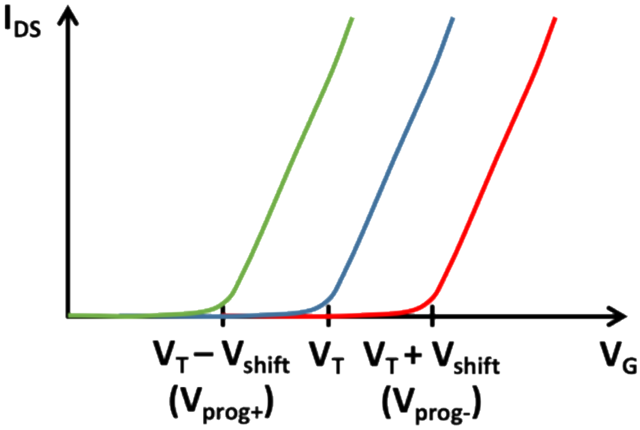


Figure 28. Impact of programming voltage on n-type FeFET threshold voltage

2) Analysis of dual-state V_T shift control

When the polarization can only be controlled using two programming voltages corresponding to two ferroelectric states (high and low), the desired positions of the new threshold voltages need to be chosen. As illustrated in Figure 30, the possibilities for the shifted threshold voltage V_T' are the following, with respect to the voltage levels representing logic states '0' and '1':

- V_T' < '0' < V_T < '1' – ALWAYS ON
- '0' < V_T' < V_T < '1' – Reduced V_T (normal switching operation)
- '0' < V_T < V_T' < '1' – Increased V_T (normal switching operation)
- '0' < V_T < '1' < V_T' – ALWAYS OFF

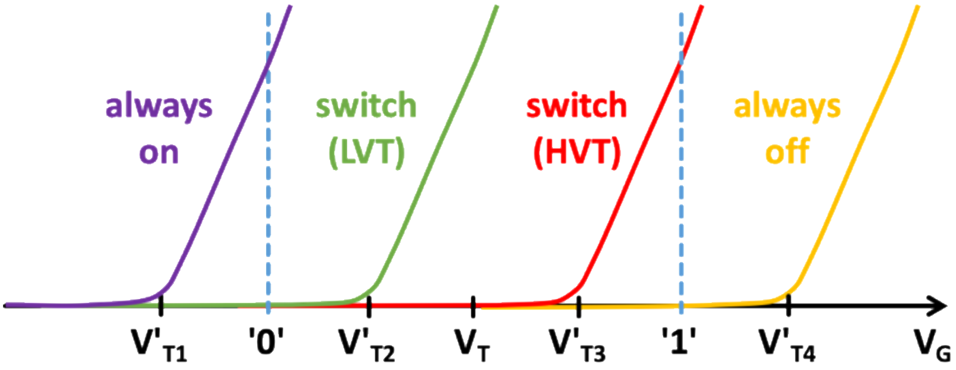
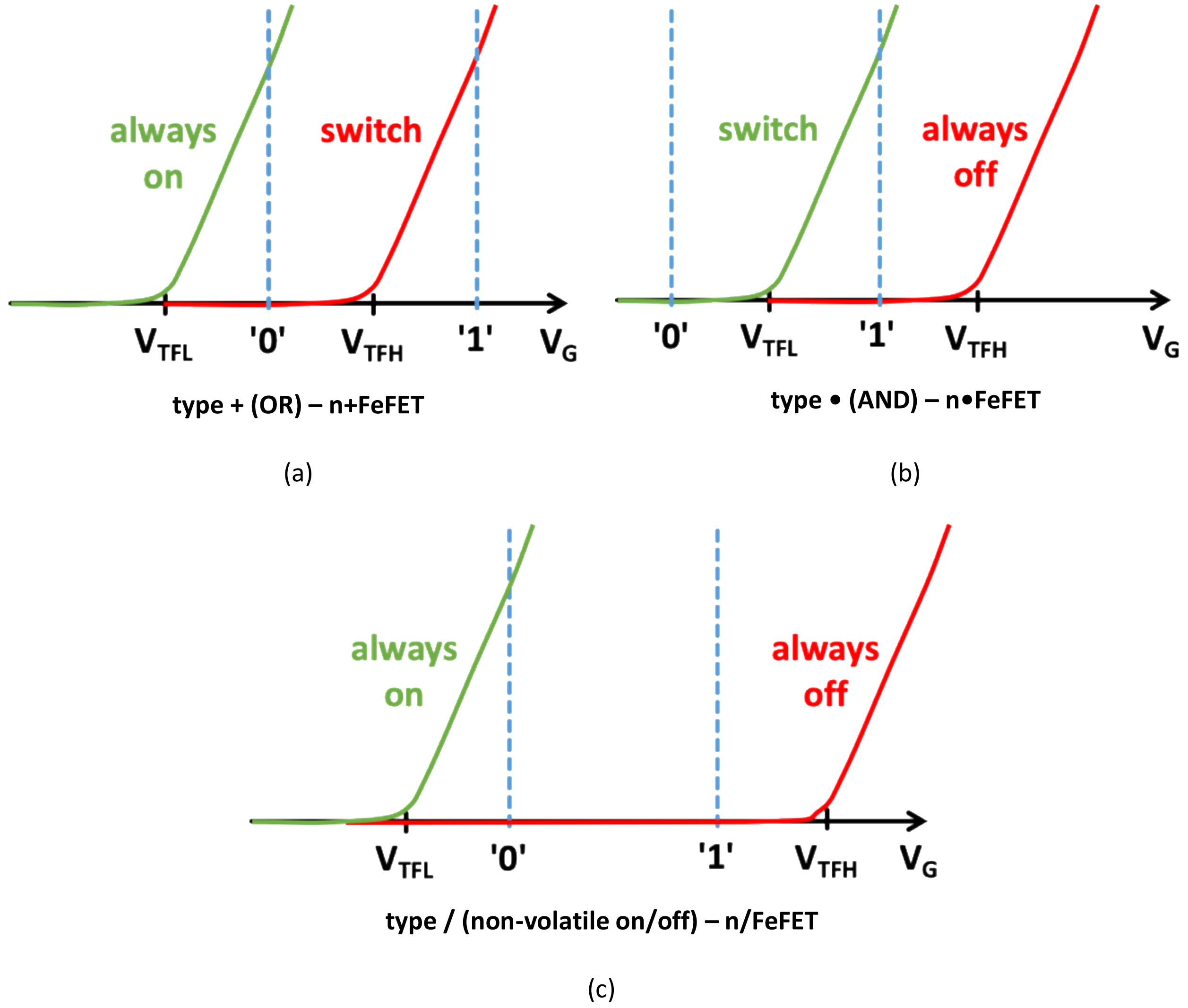


Figure 29. Illustration of n-type FeFET states as related to logic levels

In the list above, '1' and '0' represent for logic high and logic low voltage values, respectively. V_T represents the threshold voltage of a regular transistor. In an ideal case, the FeFET’s nonshifted threshold voltage (V_{TF}) should be chosen independently of it, and set to the midpoint of both desired shifted threshold voltages V_{TFL} < V_{TFH} . The actual threshold voltage shift depends on the coercive voltage used for programming, and can be used in an analog, or multi-level way; though this use-case will not be detailed.

There are therefore three interesting cases to consider:

- ALWAYS ON / Regular switch: V_{TFL} < '0' and '0' < V_{TFH} < '1' – type + (OR)
- Regular switch / ALWAYS OFF: '0' < V_{TFL} < '1' and '1' < V_{TFH} – type • (AND)
- ALWAYS ON / ALWAYS OFF: V_{TFL} < '0' and '1' < V_{TFH}, which corresponds to the largest V_T shift – type / (non-volatile on/off)



type + (OR) – n+FeFET

(a)

type • (AND) – n•FeFET

(b)

type / (non-volatile on/off) – n/FeFET

(c)

Figure 30. 3 main configurations of n-type FeFET

2. *p-type FeFET analysis*

Now let us examine the switching conditions for the p-type FeFET:

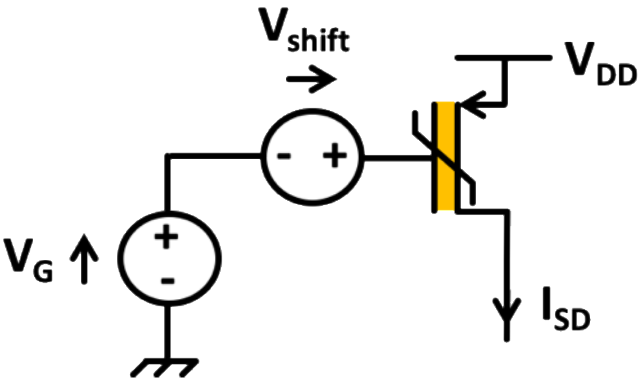


Figure 31. Bias circuit for analysis of on-conditions of p-type FeFET

1) Expression of on-conditions

The on-condition for the transistor is:

$$V_{DD} - (V_{shift} + V_G) > |V_T|$$

Which can also be expressed as

$$V_G < V_{DD} - V_{shift} - |V_T|$$

V_{shift} is defined by the programming pulse:

If V_{prog+} then $V_{shift} > 0$ V
If V_{prog-} then $V_{shift} < 0$ V

As shown in Figure 32, a positive (resp. negative) programming voltage will thus result in a lowering (resp. raising) of the threshold voltage. Note, again, that we consider that during programming, the channel potential should be set to 0 V, i.e. $V_d = V_s = 0$ V.

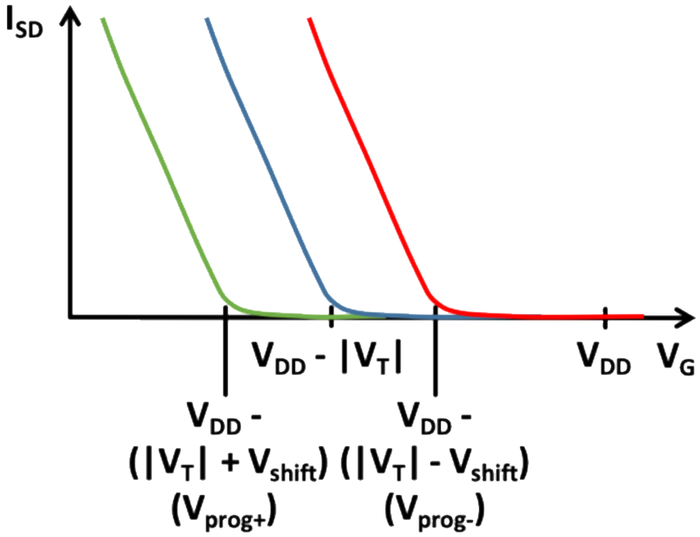


Figure 32. Impact of programming voltage on p-type FeFET threshold voltage

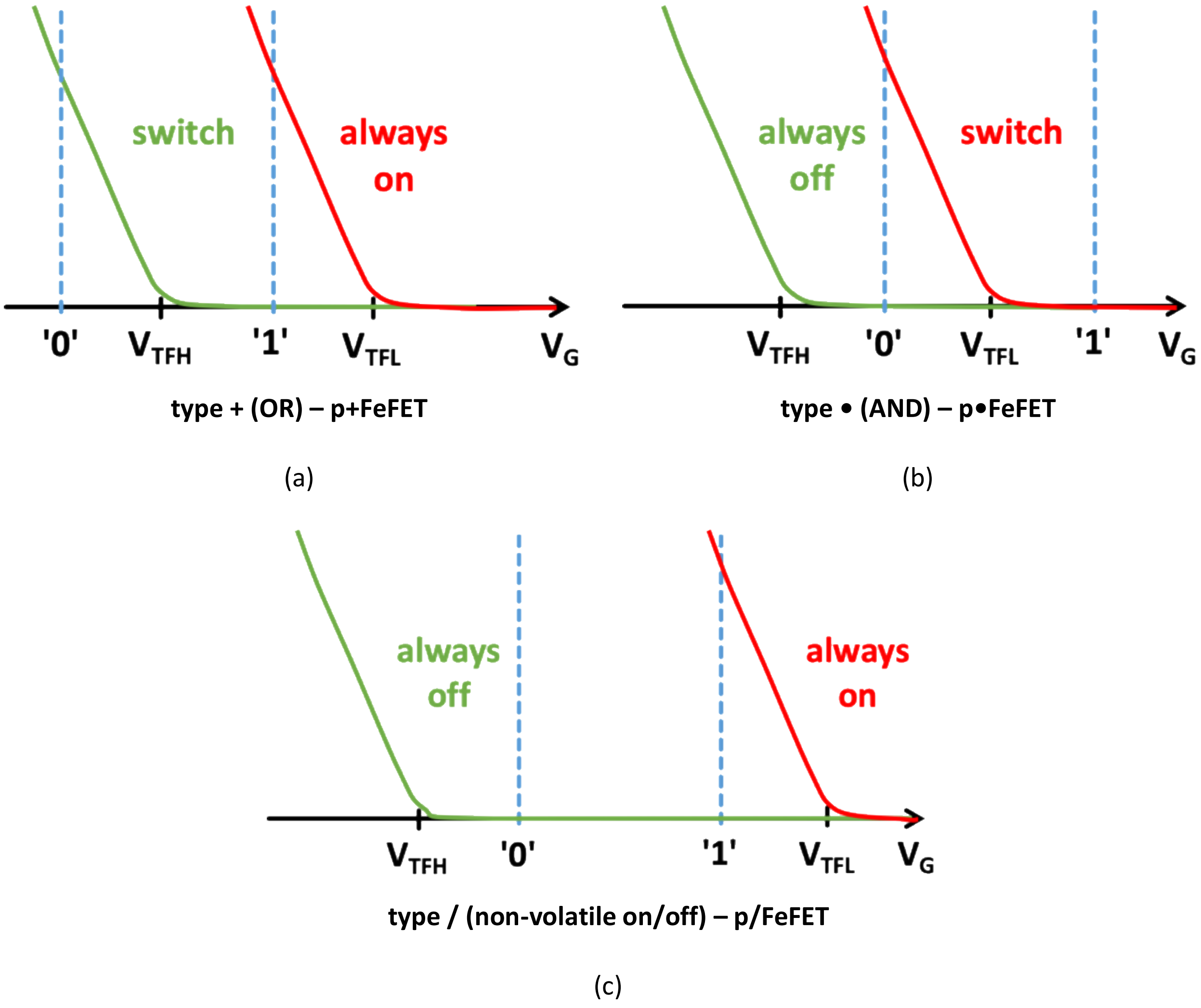
2) Analysis of dual-state V_T shift control

In this case, considering again the shifted threshold voltage V_T' ($V_T \pm V_{shift}$):

- V_{DD}-V_T' < '0' < V_{DD} - V_T < '1' – ALWAYS OFF
- '0' < V_{DD} - V_T' < V_{DD} - V_T < '1' – Increased V_T (normal switching operation)
- '0' < V_{DD} - V_T < V_{DD} - V_T' < '1' – Reduced V_T (normal switching operation)
- '0' < V_{DD} - V_T < '1' < V_{DD} - V_T' – ALWAYS ON

And so:

- Always on / Regular switch: '1' < V_{TFL} and '0' < V_{TFH} < '1' – type + (OR)
- Regular switch / ALWAYS OFF: '0' < V_{TFL} < '1' and '0' > V_{TFH} – type • (AND)
- ALWAYS ON / ALWAYS OFF: V_{TFH} < '0' and '1' < V_{TFL}, which corresponds to the largest V_T shift – type / (non-volatile on/off)



type + (OR) – p+FeFET

(a)

type • (AND) – p•FeFET

(b)

type / (non-volatile on/off) – p/FeFET

(c)

Figure 33. 3 main configurations of p-type FeFET

3. *Discussion – physical mechanisms for V_T tuning*

Several physical mechanisms are in theory available to achieve fine V_T tuning for each case:

- the operating and programming pulse voltages ('0', '1' and V_{prog+}, V_{prog-})
- the nominal device threshold voltage (V_T) – LVT, HVT devices

- the ferroelectric layer characteristics (V_{shift} and other parameters)

Among the three mechanisms, we consider that the ferroelectric layer characteristics will be fixed in this context and in a first approach by LAAS and NLB partners (this will however be a clear focus of DTCO in the next stages of the project.

We also exclude the direct use of LVT/HVT devices, since it is also important to consider the capacitance ratio. Failure to do so can lead to malfunctioning devices [13].

We will therefore work with the first mechanism i.e. operating and programming pulse characteristics to target varying V_T shift characteristics.

4. *Simulation / measurement methodology*

In addition to energy / latency measurements in cell compute mode (output varying with volatile input A), it is also necessary to measure energy / latency during programming of the non-volatile input B.

This also enables quantification (and optimization) of the conditions to achieve the various values of threshold voltage.

In this context, we typically carry out the transient simulation / measurement shown in Figure 34.

[temporary image]

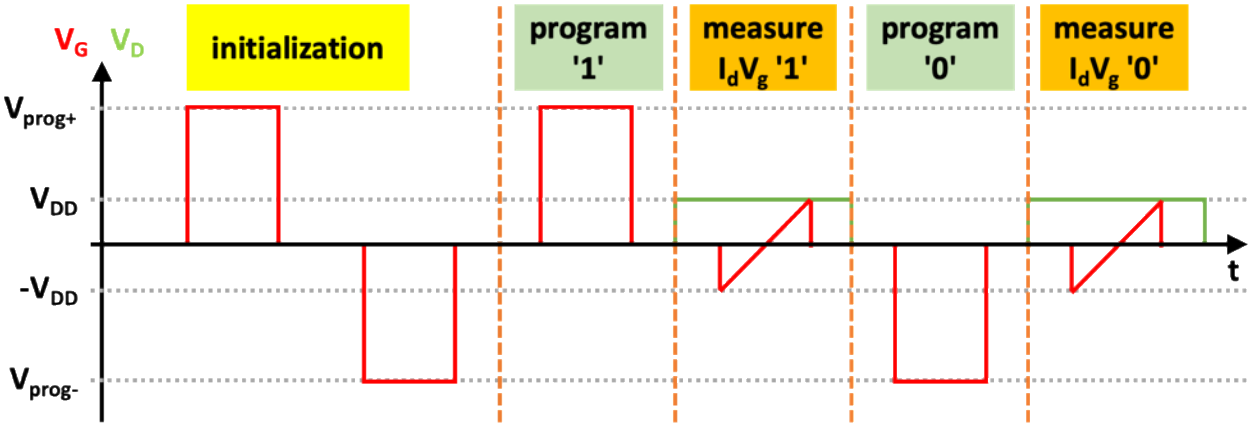


Figure 34. Simulation / measurement protocol to measure threshold voltage variation, programming energy and latency

II. CNVL_JLFE1 LIBRARY

In NV logic, single gate devices represent two sequential inputs – one non-volatile destined to remain stored for a considerable length of time, one volatile destined to change at every compute cycle. In this analysis, we will denote the non-volatile input with an underscore, and for a 2-input function will arbitrarily consider non-volatile input B and volatile input A.

1. *NAND2_1_CNVL_JLFE1 design*

For the non-volatile NAND2 gate with input operands A and B such that the output Y = !(A.B), the standard NAND2 truth table is shown below:

line	A	B	Y	
1	0	0	1	**PUN**
2	0	1	1	
3	1	0	1	
4	1	1	0	**PDN**

If we begin the analysis by looking at the PDN, we realize that we need an n•FeFET because when B=1 (V_{prog+}, V_{TFL}) it is necessary to switch ON (as with a regular switch) when A=1 (line 4 in the truth table) but also to switch OFF when A=0. When B=0 (V_{prog-}, V_{TFH}), the device is ALWAYS OFF.

For the PUN, we identify that if B=0 (V_{prog-}, V_{TFL}), then Y=1 whatever the value of A, i.e. ALWAYS ON (lines 1, 3); when B=1 (V_{prog+}, V_{TFH}), the device should switch ON for A=0 (line 2) and OFF for A=1. This is a p+FeFET.

In theory, this should cover all lines in the truth table. In addition, we see that the representation of B is identical for both PUN and PDN, implying that it may also be possible to share the ferroelectric layer between both n•type and p+type devices.

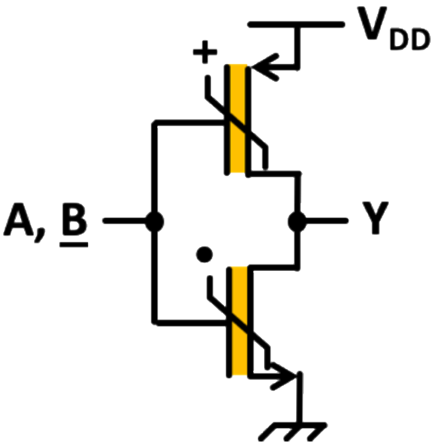


Figure 35. NAND2_1_CVNL_JLFE1 schematic

If it proves challenging to match the all the variables to meet the various supply, threshold and shift voltage requirements simultaneously, a more robust design is also possible. If the PUN is in fact a p/FeFET (ALWAYS ON, ALWAYS OFF) (lines 1,3), then it is also necessary to add a regular p-type MOSFET in parallel in the PUN to ensure switching based on A (line 2).

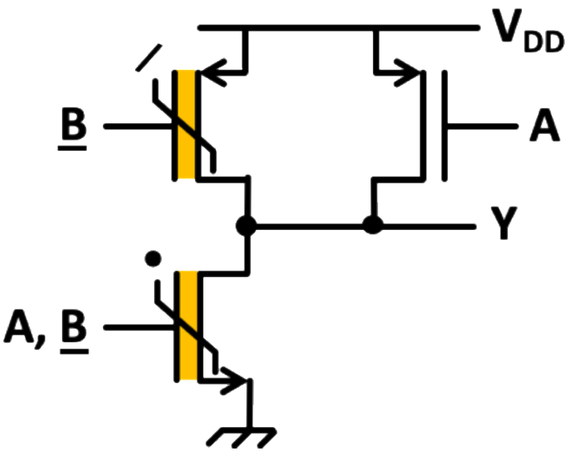


Figure 36. NAND2_1_CVNL_JLFE1 alternative schematic

2. *NOR2_1_CNVL_JLFE1 design*

We will now look at the NOR2 gate with input operands A and $\underline{B}$, output Y = !(A + $\underline{B}$).

The standard NOR2 truth table is shown below:

line	A	$\underline{B}$	Y	
1	0	0	1	**PUN**
2	0	1	0	**PDN**
3	1	0	0	
4	1	1	0	

From the PUN line (1), we need a p•FeFET because when $\underline{B}$=0 (V_{prog-}, V_{TFH}) it is necessary to switch ON (as with a regular switch) when A=0 (line 4 in the truth table) but also to switch OFF when A=1. When $\underline{B}$=1 (V_{prog+}, V_{TFL}), the device is ALWAYS OFF.

For the PDN, we identify that if $\underline{B}$=1 (V_{prog+}, V_{TFH}), then Y=0 whatever the value of A, i.e. ALWAYS ON (lines 2, 4); when $\underline{B}$=0 (V_{prog-}, V_{TFL}), the device should switch ON for A=1 (line 3) and OFF for A=0. This is an n+FeFET.

In theory, this should cover all lines in the truth table. In addition, we see that the representation of $\underline{B}$ is identical for both PUN and PDN, implying that it may also be possible to share the ferroelectric layer between both n+type and p•type devices.

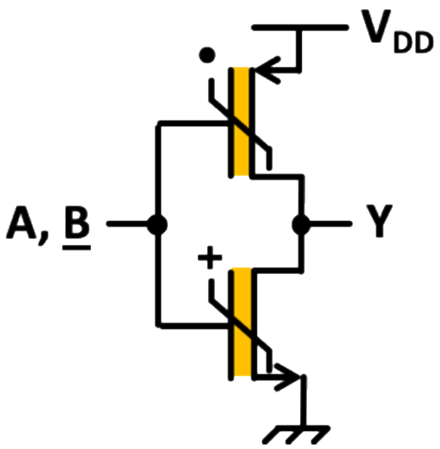


Figure 37. NOR2_1_CVNL_JLFE1 schematic

Again, if it proves challenging to match the all the variables to meet the various supply, threshold and shift voltage requirements simultaneously, then an alternative approach is possible. If the PDN is in fact an n/FeFET (ALWAYS ON, ALWAYS OFF) (lines 2,4), then it is also necessary to add a regular n-type switch in parallel in the PDN to ensure switching based on A (line 3).

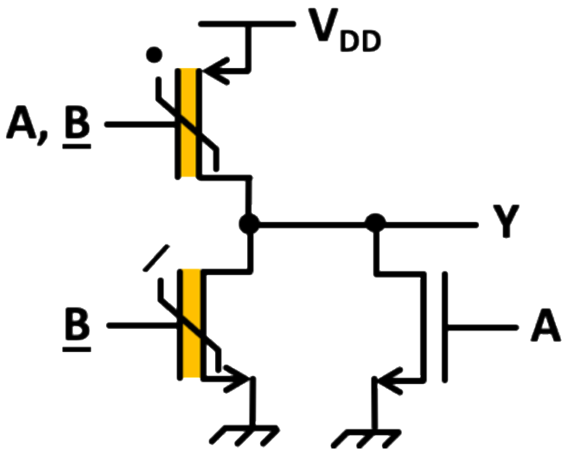


Figure 38. NOR2_1_CVNL_JLFE1 alternative schematic

3. *XOR2_1_CNVL_JLFE1 design*

We will now look at the XOR2 gate with input operands A and $\underline{B}$, output Y = A⊕$\underline{B}$.

The standard XOR2 truth table is shown below:

line	A	$\underline{B}$	Y	
1	0	0	0	PDN
2	0	1	1	PUN
3	1	0	1	
4	1	1	0	PDN

For the PDN, there are two distinct lines (1 and 4):

- line 1: we identify that if B=0 (V_{prog-}, V_{TFL}), then the device should switch ON for A=1 and OFF for A=0. If B=1, the device should be ALWAYS OFF. This is an n•FeFET driven by A, B.
- line 4: we identify that if B=1, and !B=0 (V_{prog-}, V_{TFL}) – an inverter is therefore required to generate !B – then the device should switch ON for A=0 (!A=1) and OFF for A=1 (!A=0) – another inverter is therefore required to generate !A. If B=0, and !B=1, the device should be ALWAYS OFF. This is an n•FeFET driven by !A, !B.

For the PUN, there are also two distinct lines (2 and 3):

- line 2: we identify that if B=1, and !B=0 (V_{prog-}, V_{TFL}) – an inverter is therefore required to generate !B – then the device should switch ON for A=0 and OFF for A=1. If B=0, the device should be ALWAYS OFF. This is a p•FeFET, driven by A, !B.
- line 3: we identify that if B=0 (V_{prog-}, V_{TFL}), then the device should switch ON for A=1 (!A=0) and OFF for A=0 (!A=1) – an inverter is therefore required to generate !A. If B=1, the device should be ALWAYS OFF. This is a p•FeFET, driven by !A, !B.

In this circuit, no ferroelectric layer sharing appears possible since volatile and non-volatile data inputs are crossed.

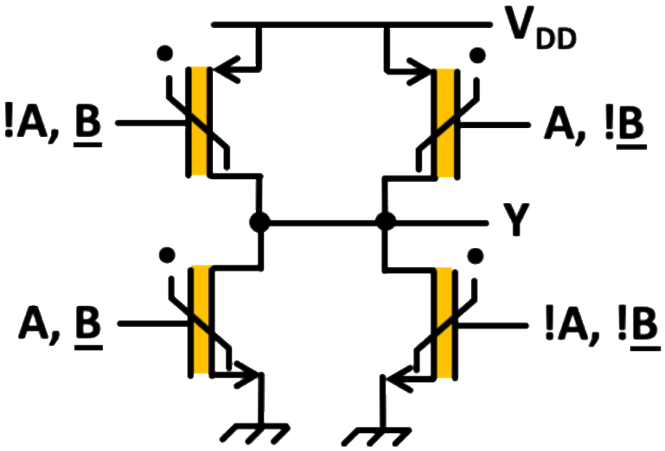


Figure 39. XOR2_1_CVNL_JLFE1 schematic

4. *Discussion*

In this analysis, we can observe that the two series-connected complementary FeFET structure is a fundamental building block. We also observe that negative unate functions can be achieved with a single structure – which is also run-time reconfigurable.

Combining individual structures with a single always on/always off switch could be a run-time configurable mechanism to target non-unate, and potentially positive unate functions.

III. CNVL_JLFE2 LIBRARY

JLFE2 could allow up to 4-input gates (2 volatile, 2 non-volatile). Even though the ferroelectric oxide layer is common to both gates around the nanowire channel, localized polarization enables independent configuration of each device in the equivalent two series-connected transistors.

Using the JLFE1 section as a basis for analysis, we now establish the on-conditions for a single JLFE2 device with independent gates.

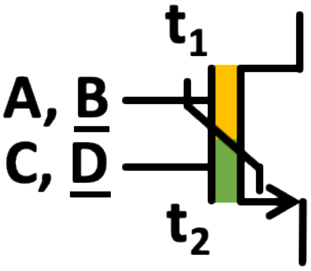


Figure 40. Formalism used to describe and analyze JLFE2 n-type device

For each individual gate, that part of the device can be considered to be on if:

- t_n = / AND $\underline{\text{NV}}$ = true; OR
- t_n = + AND (V + $\underline{\text{NV}}$ = true); OR
- t_n = • AND (V . $\underline{\text{NV}}$ = true)

t_1	A	$\underline{B}$	$state_1$	t_2	C	$\underline{D}$	$state_2$	nJLFE2 state
/	X	0	off	X	X	X	X	off
/	X	1	on	/	X	X	on	on
				+	0	0	off	off
					0	1	on	on
					1	0	on	on
					1	1	on	on
				•	0	0	off	off
					0	1	off	off
					1	0	off	off
					1	1	on	on
+	0	0	off	X	X	X	X	off
	0	1	on	/	X	X	on	on
				+	0	0	off	off
					0	1	on	on
					1	0	on	on
					1	1	on	on
				•	0	0	off	off
					0	1	off	off
					1	0	off	off
					1	1	on	on
	1	0	on	/	X	X	on	on
				+	0	0	off	off
					0	1	on	on
					1	0	on	on
					1	1	on	on
				•	0	0	off	off
					0	1	off	off
					1	0	off	off

					1	1	on	on
	1	1	on	/	X	X	on	on
				+	0	0	off	off
					0	1	on	on
					1	0	on	on
					1	1	on	on
				•	0	0	off	off
					0	1	off	off
					1	0	off	off
					1	1	on	on
•	X	0	off	X	X	X	X	off
	0	1	off	X	X	X	X	off
	1	1	on	/	X	X	on	on
				+	0	0	off	off
					0	1	on	on
					1	0	on	on
					1	1	on	on
				•	0	0	off	off
					0	1	off	off
					1	0	off	off
					1	1	on	on

10. Non-volatile crossbar (NVXbar) cell library – JLFE1, JLFE2

I. JLFE1 NVXBAR

With the JLFE1 technology, each conductance G_{xx} in the crossbar matrix (Figure 23) is implemented by a single ferroelectric transistor controlled by an independent gate voltage (Figure 41). In this case, the weight inputs are defined during programming by setting the FeFET gate voltage, and are thus stored in a non-volatile way.

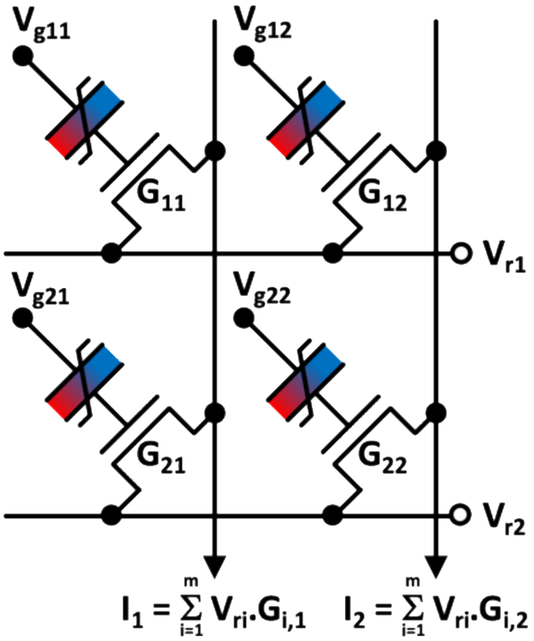


Figure 41. JLFE1 implementation of 2x2 conductance matrix

The main impact of FeFET programming is to modulate the threshold voltage V_T of the transistor in an analog and non-volatile way. This is of significant interest since it implies that it is an efficient means of representing any time-invariant data, which is the case for weights in CNNs. This is in contrast to the JL1 implementation of the crossbar, in which the threshold voltage is constant and the gate voltage represents the weight input.

According to the previously derived product term in the overall drain current equation (reproduced below), the weight inputs in the JLFE1 implementation are therefore encoded in the (non-volatile) threshold voltage value, while the gate voltage is set to be constant and common to all devices (for example, and for simplicity, to logic '1'). Data inputs are encoded in row voltage values, as in the JL1 implementation.

$$I_{DS} = \underbrace{\beta(V_{GS} - V_T)V_{DS}}_{product\ term} \; - \underbrace{\frac{{V_{DS}}^2}{2}}_{error\ term}$$

Depending on the number of discrete (distinguishable) levels, L, that can be programmed in the FeFET, the matrix devices can be used to represent data in digital ($\log_2 L = 1$) or quasi-analog ($\log_2 L \geq 2$) form. The latter solution requires in general larger devices since multiple ferroelectric domains are necessary to enable partial and progressive switching of the ferroelectric layer polarization. Distinct programming Methodologies can also be identified to program devices to reach the target threshold voltage range: (i) single pulses of varying duration and/or amplitudes, and (ii) a stream of fixed duration and amplitude pulses [14].

Another application for a matrix of JLFE1 devices is that of energy-efficient associative memory, targeting the implementation of few-shot learning paradigms in edge AI hardware platforms [15]. Few-shot learning aims to identify, given a set of support images with few image(s) per class, which support image is most similar to a query image. It can be effectively implemented using neural networks augmented with attentional memory. In this application, each support image is encoded to a feature vector, stored in the attentional memory. The attentional memory subsequently employs associative searches which apply distance metric calculations to determine which entries are most relevant to a given query image.

With the JLFE1 matrix, the distance function can be implemented by combining complementary FeFETs centered around a single search voltage such that $V_{Tn} = V_{DD}$-V_{Tp} (Figure 42). In this approach, one column pair would be composed of a column of n-type devices and a column of p-type devices. During programming, a query voltage V_{search} would be used for one pair of n- and p-devices such that after programming, $V_{Tn}+\Delta_n = V_{DD}-(V_{Tp}-\Delta_p) = V_{search}$. In this way, each n-/p-type device pair is programmed to respond with a minimum (ideally zero) combined output current during search operations if the gate voltages $V_{g11} = V_{g12}$ are queried with a corresponding voltage V_{search}; and a non-zero combined output current otherwise. In the context of a device matrix, columns would be used to associate multiple data to refine the associative search according to the programming resolution enabled by each ferroelectric device.

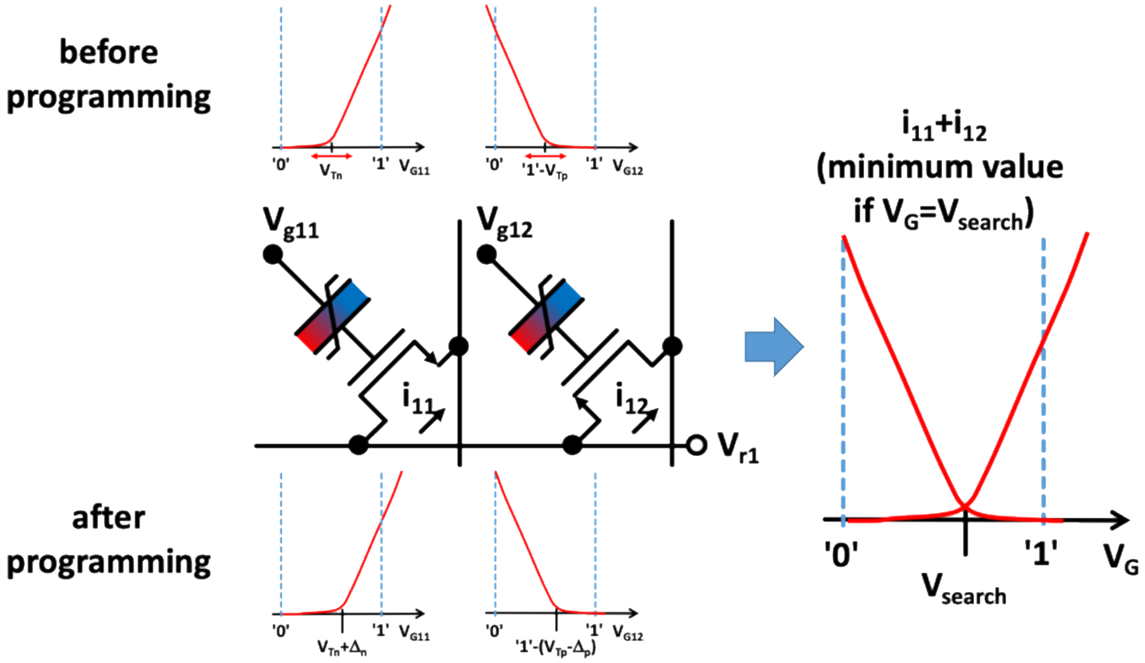


Figure 42. Implementation of analog CAM using complementary FeFETs

11. Ambipolar logic cell library – 1AP, 2AP

There is growing interest in logic circuits based on polarity-controllable devices, as evidenced by previous work developing polarity controllable silicon NWFETs [16] and 2D material devices. 32-bit adder benchmarks leveraging design possibilities at the circuit level on the 2D material devices demonstrate a ~7× reduction in EDP when compared with CMOS. Runtime reconfigurable combinational [19] and dynamic circuits have also been devised, such as 6-T NAND/NOR/MIN [17] and AOI/XOR/XNOR cells. Multi-independent gate (MIG)-RFETs allow for compact multi-bit adders, buffer-free multiplexers and arithmetic logic units (ALU) [18].

In this work, we target ultra-fine grain reconfigurable architectures and cells based on polarity-controllable FETs, leading also to applications in interconnect topologies and architectures based on reconfigurable cells.

The AP technologies define reconfigurable transistors via electrostatic doping using g_x (polarity gate). Here we will assume:

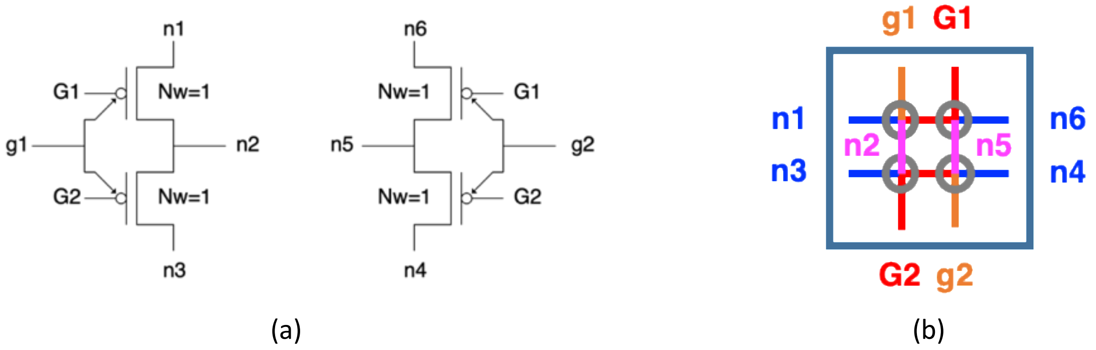
- g_x=1: n-type
- g_x=0: p-type

AP1 supposes a "U"-shaped transistor (i.e. two single gate VNWFET devices connected in series, where one acts as the polarity-control transistor.

AP2 supposes two gates stacked in series on a single nanowire device, and where one of the gates acts as the polarity gate.

I. RECONFIGURABLE (UNIVERSAL) LOGIC GATE

Efficient implementations of several logic functions are possible using hardware structures based on groups of two ambipolar transistors in series [20]. Figure 43 shows a tile implementing two groups of such structures and capable of realizing multiple logic functions according to how tile terminals are connected to power rails, data inputs/outputs, polarity-control (configuration) inputs.



(a)

(b)

Logic	n1	n2	n3	n4	n5	n6	G1	G2	g1	g2
XOR2	Gnd	Out	Vdd	Gnd	Out	Vdd	A	A'	B'	B
XNOR2	Gnd	Out	Vdd	Gnd	Out	Vdd	A	A'	B	B'
NAND2	Out	Vdd	Out	Out	-	Gnd	A	B	Gnd	Vdd
NOR2	Vdd	-	Out	Out	Gnd	Out	A	B	Gnd	Vdd
INV	Vdd	Out	Vdd	Gnd	Out	Gnd	A	A	Gnd	Vdd
BUF	01	Vdd	Out	Out	Gnd	01	A	01	Gnd	Vdd

(c)

Figure 43. Multi-function reconfigurable logic tile using groups of ambipolar transistors in series. (a) Tile schematic. (b) Sticks representation of VNWFET implementation. (c) Function table.

II. LOGIC GATES AND ROUTING

RFET devices are also promising solutions for the flexible routing of interconnected devices. A cross-shape design (CS-RFET) from NLB [21] employing multiple sources paves the way to establish novel RFET based logic circuits. Two designs are explored:

- a balance-tunable inverter design as shown in Figure 44(a)
- a 2-way multiplexer as shown in Figure 44(b)

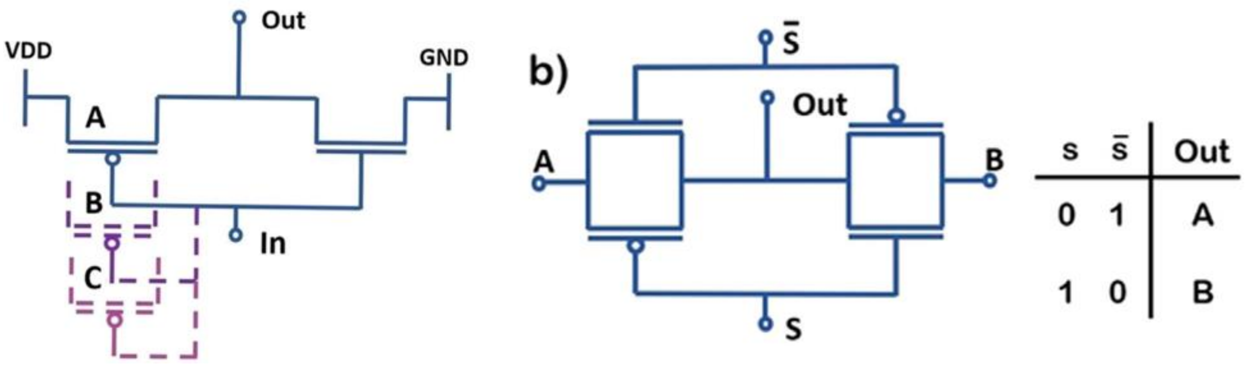
In both designs, adjacent devices with symmetric transistor characteristics are configured such that one transistor is always programmed to be p-type configuration ($V_{PG} < 0V$) while the other is programmed as n-type ($V_{PG} > 0V$).

For the inverter, based on a single CS-RFET structure, it is possible to steer from 1 to 3 pull-up branches to balance the inverter according to energy and delay constraints. In particular, the rise time is maximal with a single branch and minimal with all branches operating together, while the fall time remains the same (since the pull-down branch always operates with a single branch). Beyond logic, such a circuit could be of interest for synaptic functions, if combined with non-volatile storage feature at either the inputs or the outputs.

For routing, two CS-RFETs can be combined as a 2:1 multiplexer (MUX) in a transmission gate configuration. A typical MUX is a building block for the arithmetic circuits to route inputs to output in the data path. In this specific example, one of the interconnected wires and the corresponding control gate is left open for both adjacent transistors. Two input signals (A and B) are injected into two branches of both adjacent transistors. The circuit then operates as follows:

- select voltage s is '0' and !s is '1': the A branch of both devices is enabled to conduct to the output, while the other branches of the CS-RFETs are successfully blocked. Consequently, the output of the logic gate will follow signal A.
- select voltage s is '1' and !s is '0': the circuit enables the branches with signal B to conduct the output, and the gate output follows signal B.

Hence, we show that a single CS-RFET device can efficiently implement multiple transistors which share a common drain terminal and carrier type.



s	$\bar{s}$	Out
0	1	A
1	0	B

(a) (b)

Figure 44. RFET-based logic circuits. (a) Balance-tunable inverter. (b) MUX2.

12. Non-volatile ambipolar logic cell library – 1APFE, 2APFE

We now consider how the augmented functionality of ambipolar devices, combined with ferroelectric properties, can be used in a logic circuit context.

I. OPERATING PRINCIPLES

Let us suppose that the ambipolar ferroelectric field effect transistor (reconfigurable FeFET, or RFeFET) is a two-gate device that can be symbolized as below:

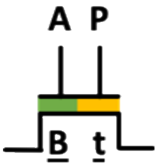


Figure 45. Reconfigurable FeFET (RFeFET) symbol. A and B represent volatile and non-volatile data inputs, respectively. P and t represent the polarity-control gate and the programmed transistor type, respectively. Non-volatile properties are denoted with an underline.

Gates are for data (volatile data A and non-volatile data B), and for polarity definition (non-volatile device property t). It is worth noting that P could also be used to enter volatile data (although this characteristic is not considered in this work). For the non-volatile property t, we consider that

- the device can be configured to n-type or p-type
- it can also be configured to an always on / switch (OR, + device) or to an always off / switch (AND, • device) using the notation introduced in section 0.I

The means by which configuration b) can be achieved is either static (design-time definition) via doping (underlying HVT or LVT device – for the VNWFET, this also depends on nanowire diameter) or dynamic (run-time definition) via the programming voltage applied to P.

This gives the following table:

type	on-condition	A	B	state
n+	A + B = 1	0	0	off
		0	1	on
		1	0	on
		1	1	on
n•	A . B = 1	0	0	off
		0	1	off
		1	0	off
		1	1	on
p+	!A + !B = 1	0	0	on
		0	1	on
		1	0	on
		1	1	off
p•	!A . !B = 1	0	0	on
		0	1	off
		1	0	off
		1	1	off

Let us now explore the logic capabilities of a two RFeFET structure, as illustrated in Figure 46.

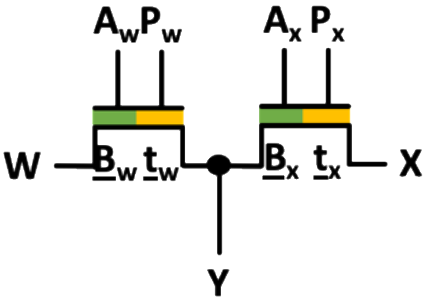


Figure 46. Two-RFeFET structure for configurable logic functions.

Preliminary remarks:

- data inputs are considered to be W, X; A_w, B_w; A_x, B_x. Each can be defined as time-varying data, or as static 1 (conventionally considered to be V_{DD}) or static 0 (conventionally considered to be GND)
- data output is considered to be Y

- device type inputs P_w and P_x are considered to define non-volatile properties t_w and t_x respectively as one of the four possible configurations (n+, n•, p+, p•) They are drawn in the symbol as facing the output connection Y as in most configurations this would be the drain terminal (which defines the ambipolar property for two-gate devices)

We will begin the analysis by exploring which logic functions can be defined in a static logic configuration (such that W, X are static 1, 0 or static 0, 1 respectively). Only negative unate logic functions (type of boolean function which has monotonic properties - e.g. NAND, NOR) will be possible in this approach and with this structure. For non-unate functions (e.g. XOR, XNOR), it will be necessary to investigate if the combination of two 2-device structures with common W, X, Y can be of use.

II. STATIC LOGIC CONFIGURATIONS

In this section, we consider that W and X must be connected to one or other of the supply rails, consistent with the static logic design style.

- NAND2: Y = !(A . B); W = 0; X = 1; t_w = n•; t_x = p+; $A_w = A_x$ = A; $B_w = B_x$ = B
- NOR2: Y = !(A + B); W = 0; X = 1; t_w = n+; t_x = p•; $A_w = A_x$ = A; $B_w = B_x$ = B

It should be noted that for both cases, W and X (and t_w and t_x), can be switched in case of routing parasitic optimization.

An inverter can be implemented simply by setting B=1 for the NAND2 or B=0 for the NOR2.
It is also possible to implement a TRUE (1) gate by setting B=0 for the NAND2; and a FALSE (0) gate by setting B=1 for the NOR2.

For the XOR2 operation, this requires two 2-RFeFET structures with connected W, X, Y:

- structure 1: W = 0; X = 1; t_w = n•; t_x = p•; A_w = !A; B_w = !B; A_x = A; B_w = !B
- structure 2: W = 0; X = 1; t_w = n•; t_x = p•; A_w = A; B_w = B; A_x = !A; B_w = B

III. PASS TRANSISTOR LOGIC (PTL) CONFIGURATIONS

In PTL, W and X can be disconnected from the supply rails and propagate data. In this case we can identify the following function configurations:

- Propagation: Y = A; W = X = A; t_w = n+; t_x = p+; $A_w = B_w$ = 1; $A_x = B_x$ = 0
- Pass transistor: Y = A.S; W = X = A; t_w = n+; t_x = p+; A_w = S; B_w = 1; A_x = !S; B_x = 0
- MUX: Y = A.S1 + B.S2; W = A; X = B; t_w = n•; t_x = n•; $A_w = S_1$; B_w = 1; $A_x = S_2$; B_x = 1

The XOR2 function requires two 2-RFeFET structures with connected A_w, A_x, Y:

- structure 1: W = !A; X = A; t_w = n•; t_x = p•; A_w = 1; B_w = B; A_x = 0; B_w = B
- structure 2: W = A; X = !A; t_w = n•; t_x = p•; A_w = 1; B_w = !B; A_x = 0; B_w = !B

It should be noted that there are other permutations of terminal assignments which can also work.

In a continued "reconfigurable" spirit, considering that such a structure should be connected to input/output pins and supply rails in different ways (including the building of composite 2-device structures for non-unate static logic functions), we will look at how to configure switches to always ON (short-circuit, connection) and to always OFF (open-circuit, disconnection) to implement reconfigurable routing resources in D4.3.

13. Conclusion

In this deliverable we presented the hypotheses and fundamentals of work related to logic and arithmetic circuit design, covering the scope of the DTCO (Design-Technology Co-Optimization) approach, the definition of design styles and metrics used in logic design and the data management and nomenclature used throughout the project.

We studied and characterized logic cells based on the static logic approach (P-type to incorporate technology fabrication limitations) as well as the complementary logic approach for a more conventional comparison. We focused on evaluating INV1, NAND2, NOR2 and XOR2 logic functions based on vertical nanowire transistors (with single gate JL1 or two-gate stack JL2) in terms of delay and power consumption, and proving that this new technology can be used as the basic building block for logic gates.

We also proposed novel logic and arithmetic circuits based on more advanced VNWFET technologies incorporating either ferroelectric layers for non-volatile operation, ambipolar contacts for reconfigurability, or both. Here, the goal is rather to illustrate the potential of the technology, rather than to pursue detailed characterization at this stage. The intent is to demonstrate building blocks for unconventional computing approaches and open paths for circuit-level hardware demonstrators in the latter stages of the project.

As technology development and logic cell design progresses, this version of D4.01 is intended to serve as a reference document. This information will be used mainly in WP4 (to work towards a scaled down version of N^2C^2 in D4.4 as well as a second version of the virtual scalable N^2C^2 in D4.5b). It will also be used in WP1 for hardware design, and in WP5 to enable architectural exploration.

As the FVLLMONTI project progresses, the content of this deliverable will also be refined and updated to reflect opportunities and limitations that appear according to the state of technology and logic circuit development.

14. References

[1] L. Kong, J. Tan, J. Huang, G. Chen, S. Wang, X. Jin, P. Zeng, M. Khan, and S. K. Das, "Edge-computing-driven internet of things: A survey," ACM Comput. Surv., vol. 55, dec 2022.

[2] G. E. Moore et al., "Cramming more components onto integrated circuits," 1965.

[3] B. Hoefflinger, ed., Chips 2020. Springer Berlin Heidelberg, 2012.

[4] A. Bosio, I. O'Connor, G. S. Rodrigues, F. K. Lima, E. I. Vatajelu, G. Di Natale, L. Anghel, S. Nagarajan, M. C. R. Fieback, and S. Hamdioui, "Rebooting computing: The challenges for test and reliability," in 2019 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), pp. 8138–8143, 2019.

[5] S. Mukesh and J. Zhang, "A review of the gate-all-around nanosheet FET process opportunities," Electronics, vol. 11, p. 3589, Nov. 2022.

[6] G. Larrieu and X.-L. Han, "Vertical nanowire array-based field effect transistors for ultimate scaling," Nanoscale, vol. 5, no. 6, pp. 2437–2441, 2013.

[7] A. Hamzah, R. Ismail, N. E. Alias, M. L. P. Tan, and A. Poorasl, "Explicit continuous models of drain current, terminal charges and intrinsic capacitance for a long-channel junctionless nanowire transistor," Physica Scripta, vol. 94, no. 10, p. 105813, 2019.

[8] C. Mukherjee, M. Deng, F. Marc, C. Maneux, A. Poittevin, I. O'Connor, S. Le Beux, C. Marchand, A. Kumar, A. Lecestre, et al., "3d logic cells design and results based on vertical nwfet technology including tied compact model," in 2020 IFIP/IEEE 28th International Conference on Very Large Scale Integration (VLSI-SOC), pp. 76–81, IEEE, 2020.

[9] C. Mukherjee, A. Poittevin, I. O'Connor, G. Larrieu, and C. Maneux, "Compact modeling of 3D vertical junctionless gate-all-around silicon nanowire transistors towards 3d logic design," Solid-State Electronics, vol. 183, p. 108125, 2021.

[10] R. Van Langevelde and F. M. Klaassen, "Effect of gate-field dependent mobility degradation on distortion analysis in mosfets," IEEE Trans. Electron Devices, vol. 44, no. 11, pp. 2044–2052, 1997.

[11] E. T. Breyer, H. Mulaosmanovic et al., "Demonstration of versatile nonvolatile logic gates in 28nm HKMG FeFET technology," IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–5, 2018.

[12] C. Marchand, I. O'Connor, M. Cantan, E.T. Breyer, S. Slesazeck, T. Mikolajick, "FeFET based Logic-in-Memory: an overview," 16th International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), Apulia, Italy, 28-30 June 2021, (DOI: 10.1109/DTIS53253.2021.9505078)

[13] D. Deleruyelle, Personal communication, July 2023

[14] M. M. Sharifi, et al., "Application-driven design exploration for dense ferroelectric embedded non-volatile memories," International Symposium on Low Power Electronics and Design, 2021

[15] M. Niemier, X.S. Hu, L. Liu, M. Sharifi, I. O'Connor, D. Atienza, G. Ansaloni, C. Li, A. Khan, D. Ralph, "Cross Layer Design for the Predictive Assessment of Technology-Enabled Architectures," Design, Automation and Test in Europe (DATE), Antwerp, Belgium, 17-19 April 2023

[16] M. D. Marchi et al., "Configurable Logic Gates Using Polarity-Controlled Silicon Nanowire Gate-All-Around FETs". DOI: 10.1109/LED.2014.2329919

[17] J. Trommer et al., "Elementary Aspects for Circuit Implementation of Reconfigurable Nanowire Transistors". DOI: 10.1109/LED.2013.2290555

[18] S. Rai et al., "Designing Efficient Circuits Based on Runtime-Reconfigurable Field-Effect Transistors". DOI: 10.1109/TVLSI.2018.2884646

[19] J. Trommer et al., "Functionality-Enhanced Logic Gate Design Enabled by Symmetrical Reconfigurable Silicon Nanowire Transistors". DOI: 10.1109/TNANO.2015.2429893

[20] S. Bobba et al., " Physical synthesis onto a Sea-of-Tiles with double-gate silicon nanowire transistors," DAC 2012, DOI: 10.1145/2228360.2228369

[21] C. Cakirlar, M. Simon, G. Galderisi, I. O'Connor, T. Mikolajick, J. Trommer, "Cross-Shape Reconfigurable Field Effect Transistor for Flexible Signal Routing," Materials Today Electronics, vol. 4, June 2023, 100040 (DOI: 10.1016/j.mtelec.2023.100040)