Viable Paths Towards Graphene Circuits: Implementation Styles and Logic Synthesis Tools

Sandeep Miryala, Valerio Tenace, Andrea Calimera, Enrico Macii, Massimo Poncino

POLITECNICO
DI TORINO

DAUIN - Dipartimento di Automatica e Informatica

enrico.macii@polito.it

Graphene: A Viable Candidate for Future ICs

Single atom layer of Graphite with C atoms packed in a hexagonal lattice

Some other (electrical) properties

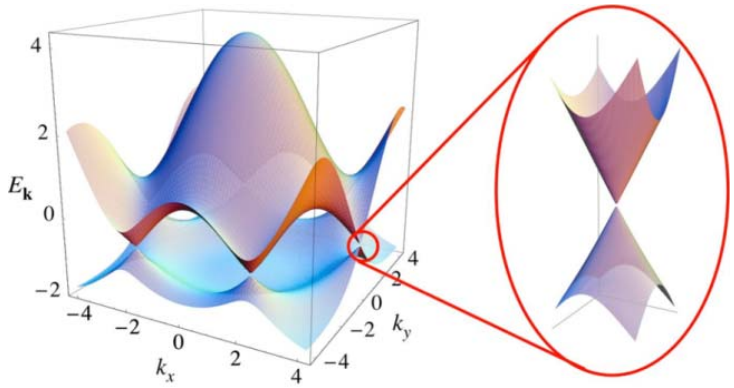
- Highest current density
- Longest mean free path
- Highest intrinsic mobility
- High thermal conductivity

*"It's the **thinnest** possible material you can imagine. It also has the **largest surface-to-weight ratio**: with one gram of graphene you can cover several football pitches... it's also the **strongest** material ever measured; it's the **stiffest** material we know; it's the most **stretchable** crystal. That's not the full list of superlatives, but it's pretty impressive."*

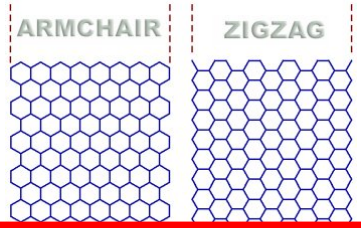
Prof. Andre Geim
Nobel Prize 2010

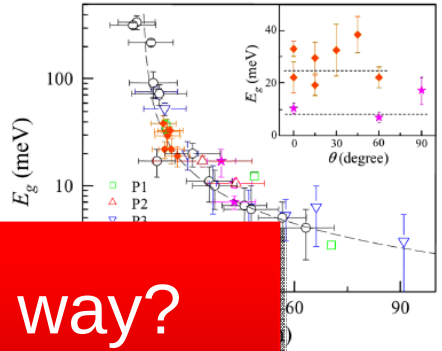
Graphene is a Wonder Material

- **But also Superheroes have limits**
  - Zero Band-Gap: low I_{on}/I_{off} ratio
  - Not suitable for digital applications
    - In the traditional sense!

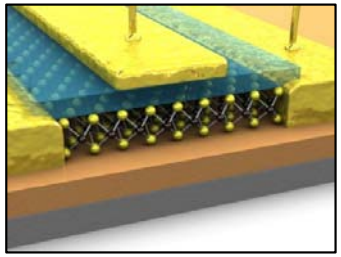


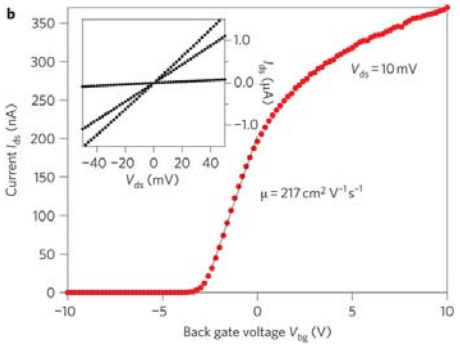
- **Standard (i.e., CMOS-like) digital circuits need a band-gap**
  - Trying to "artificially" create the band-gap
    (e.g., Nano-Ribbons or 2D Composities (TMDC):
    - May drastically affect the intrinsic properties of Graphene
    - May dramatically impact cost and stability





Is that the only way?





A Different View

- **We propose a simple concept**
  - Exploit the intrinsic properties of graphene rather than trying to modify them

- **Face the problem from a different angle**
  - Identify circuit implementation styles suitable to the properties of graphene, instead of playing with standard CMOS-like styles (which have been built and optimized for silicon)
  - Provide CAD tools to quantify the figures of merit of graphene circuits

P-N Junction on Pristine Graphene

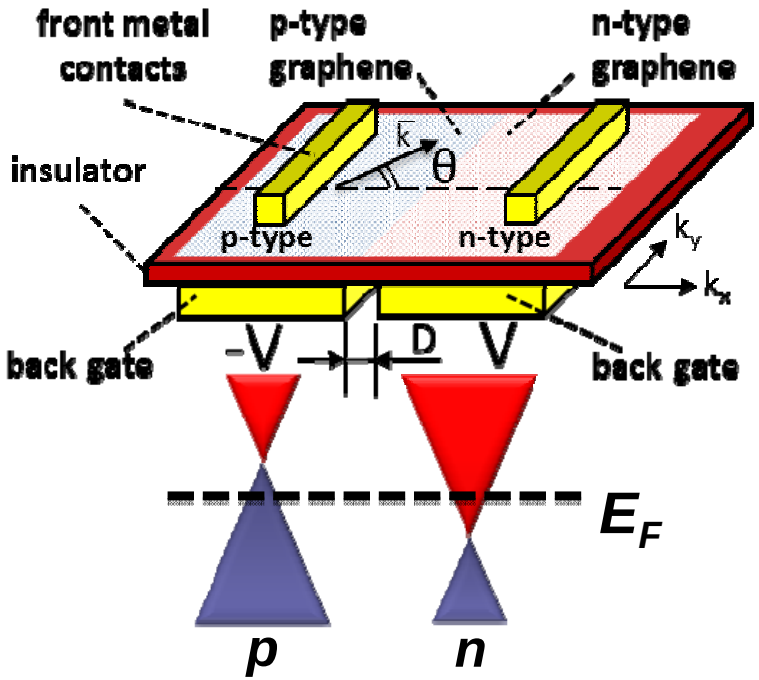
- **Electrostatic doping through split metal back-gates**

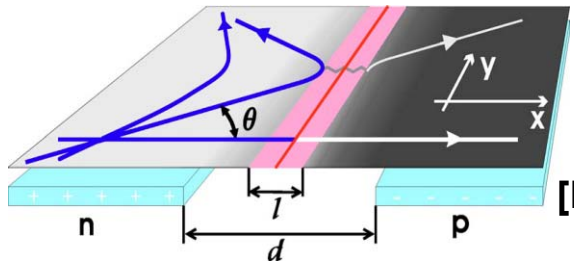
  $-V \rightarrow p\text{-}type$

  $+V \rightarrow n\text{-}type$

- **Transmission Probability**

$$T(\theta) = \begin{cases} 1 & when \quad pp/nn \\ \cos^2(\theta) e^{-\pi k D \sin^2(\theta)} & when \quad pn/np \end{cases}$$

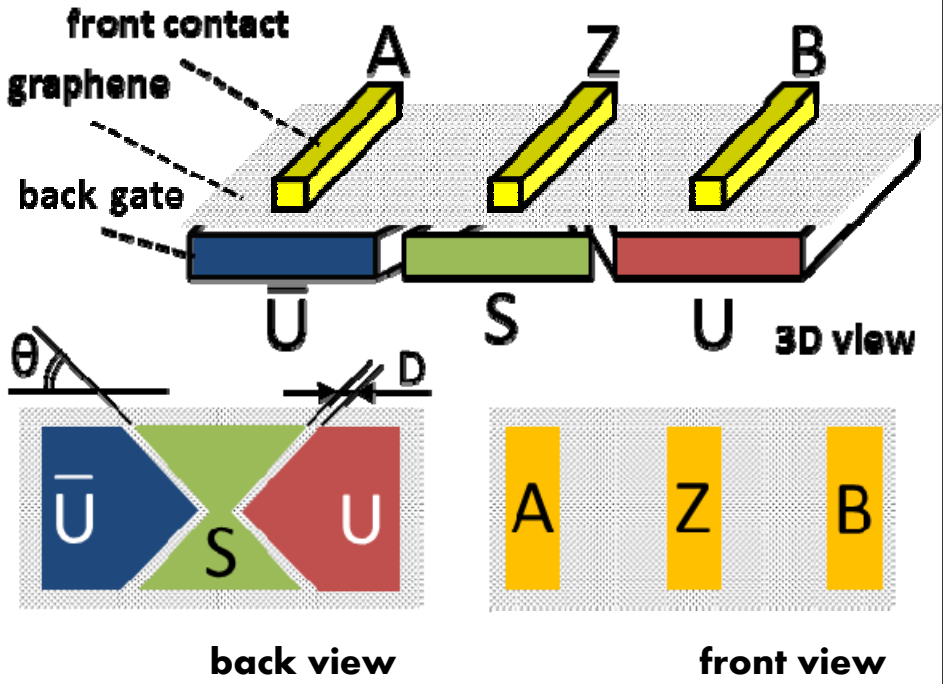


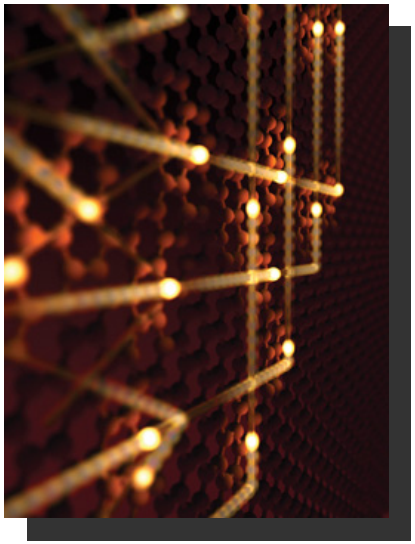


[Fal'ko06]

Reconfigurable (RG)-MUX: Structure

- **First proposed by IBM**
- **Two back faced PN-junctions**
  - 3 split back-gates which implement the electrostatic doping
  - Isolated from graphene by a thin layer of oxide
  - 3 front metal-to-graphene contacts which serve as input (A,B) and output (Z) pins

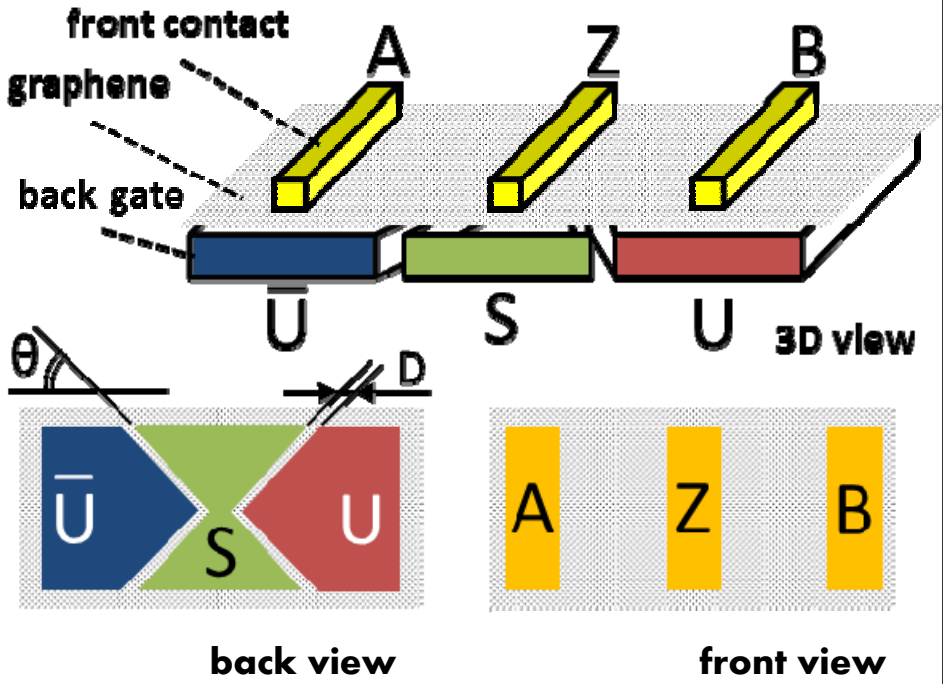




Graphene: The Ultimate Switch
IEEE Spectrum, Jan. 2012

RG-MUX: Transmission Probability

- **Back-gates control the doping profile of graphene**
  - - V → p-type graphene
  - +V → n-type graphene
- **Different doping profiles of adjacent graphene regions define the carriers transmission probabilities from inputs (A, B) to the output (Z)**



Transmission probability across the junctions

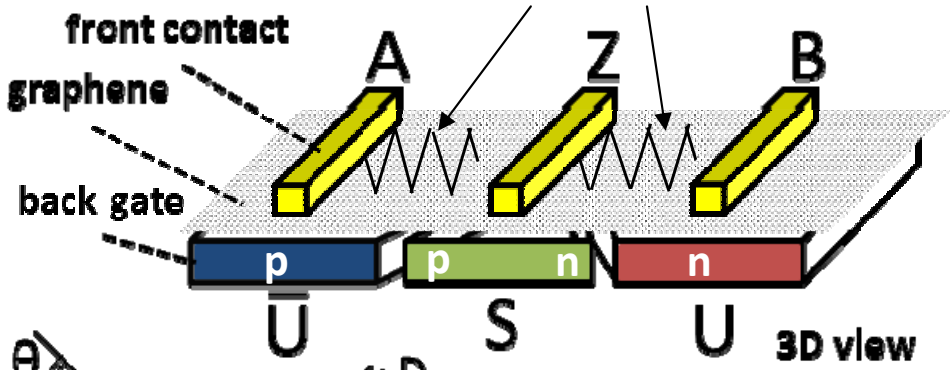
$$T_{AZ} = \begin{cases} 1 & V_S = V_{\bar{U}} \rightarrow pp/nn \\ \cos^2(\theta) e^{-\pi k_F D \sin^2(\theta)} & V_S \neq V_{\bar{U}} \rightarrow pn/np \end{cases}$$

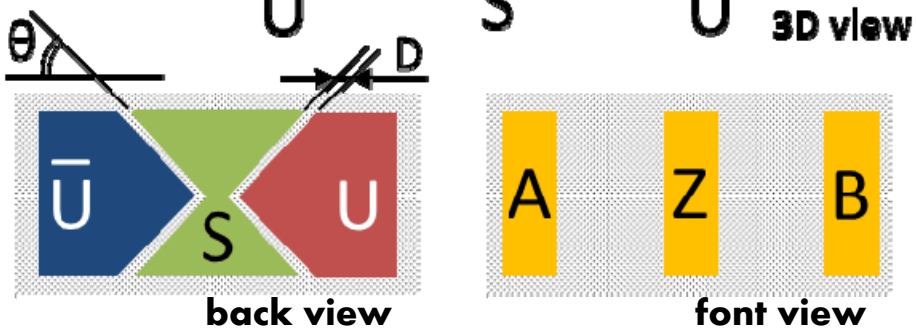
$$T_{BZ} = \begin{cases} 1 & V_S = V_U \rightarrow pp/nn \\ \cos^2(\theta) e^{-\pi k_F D \sin^2(\theta)} & V_S \neq V_U \rightarrow pn/np \end{cases}$$

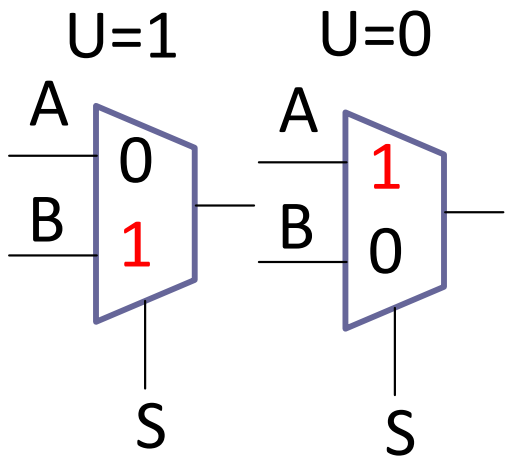
RG-MUX: Electrical/Logic Behavior

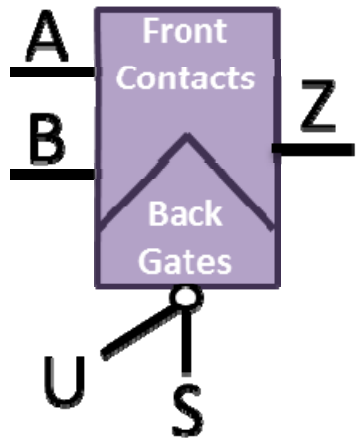
- **U and U' driven by complementary voltages**
  - U = '1' (Vdd/2)
  - U'= '0' (-Vdd/2)
- **if S=U' then Z←A**
  - p-p-n configuration
    - T_{AZ}=1; T_{BZ} ≈ 0.00003
    - R_{AZ}= R_0 /1 ≈ 300Ω
    - R_{BZ}= R_0 /T_{BZ} ≈ 10^7Ω
- **if S=U then Z←B**
  - p-n-n configuration
    - T_{AZ} ≈ 0.00003; T_{BZ} = 1
    - R_{AZ}= R_0/T_{AZ} ≈ 10^7Ω
    - R_{BZ}= R_0 /1 ≈ 300Ω

$$R_{in-out} = R_0 / T(V_S)$$







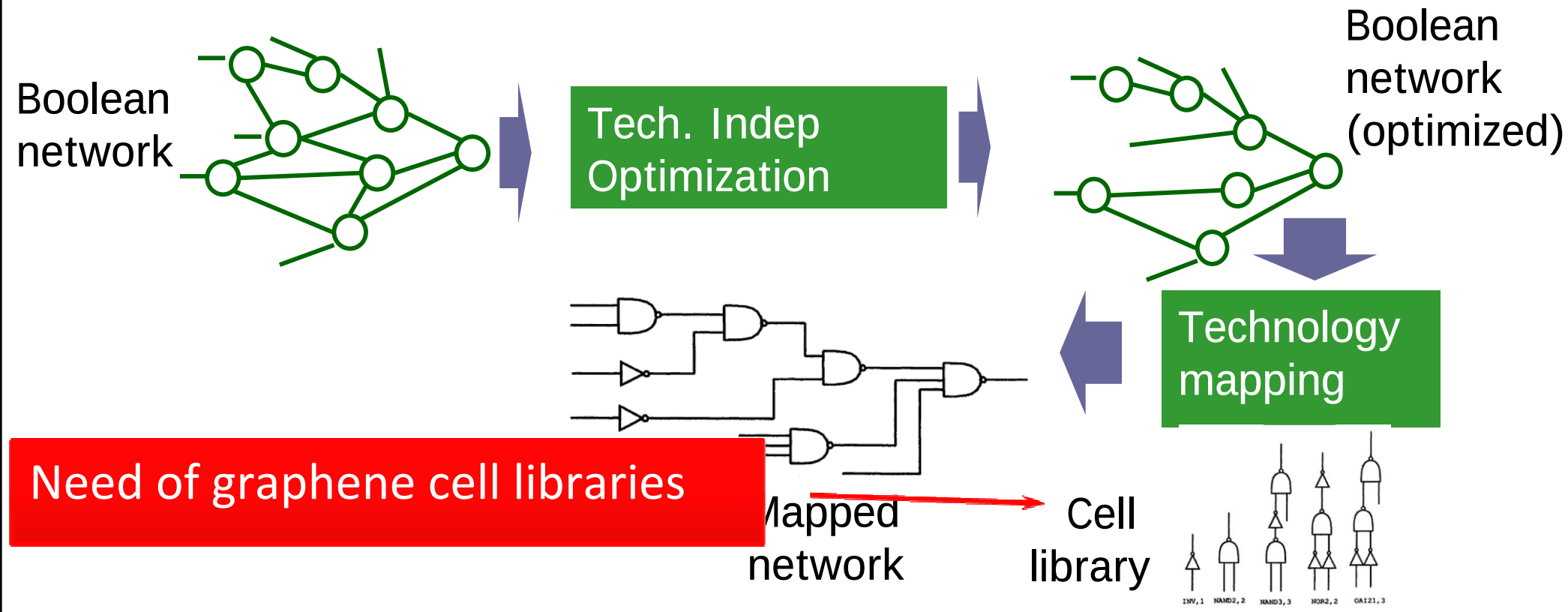


Possible Implementations and Logic Styles

- **Inspired by CMOS technologies, make use of MUXs/EXORs as logic primitives**
  - A. Standard Cell Style (STC)
    - Logic primitive: RG-MUX
    - Synthesis tool: Multi-level Logic Synthesis
  - B. Tree of MUX (TMUX)
    - Logic primitive: RG-MUX
    - Synthesis tool: BDD
  - C. FPGA/MUX
    - Logic primitive: MUX-based LUT
    - Synthesis tool: LUT decomposition

A) Standard Cell Design Style (STC)

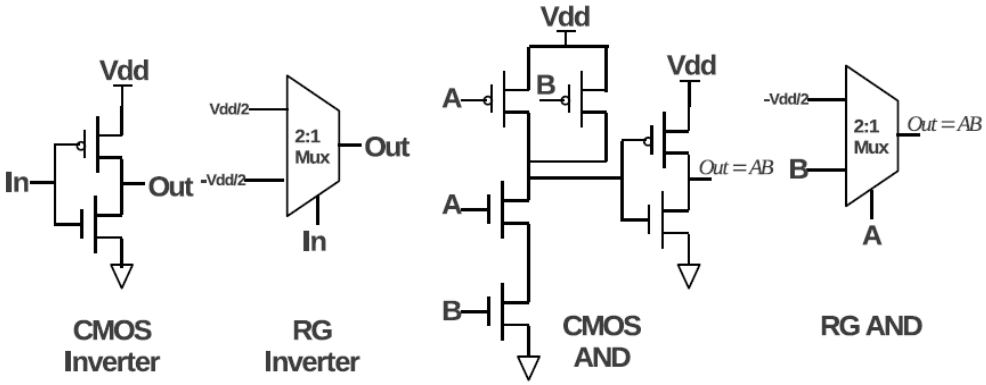
- **Adapting standard logic synthesis flow**
  1. Start from a generic Boolean Network
  2. Optimize it in terms of some cost-function (Area, Delay, Power)
  3. Map to a real technology using cell libraries



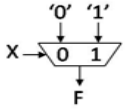
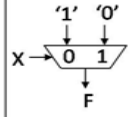
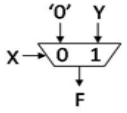
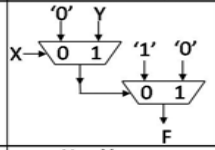
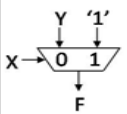
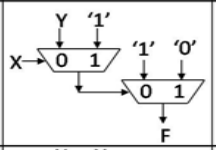
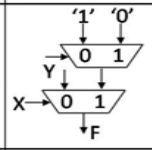
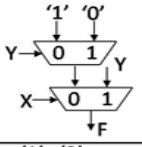
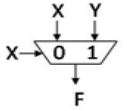
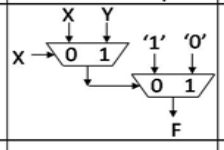
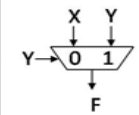
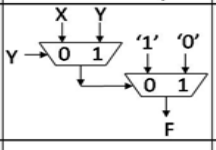
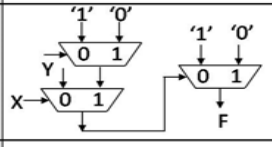
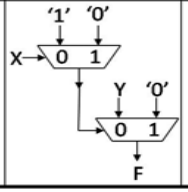
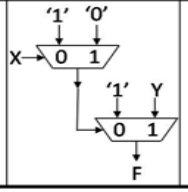
Need of graphene cell libraries

A) RG-MUX Logic-Gates

- By properly configuring one, or more, RG-MUXs it is possible to implement all basic logic functions



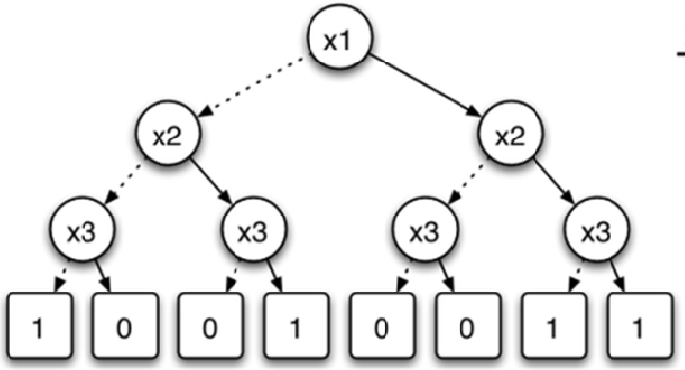
	CMOS	Graphene
Device Area	1	0.94
Switching Delay	1	0.18
Power	1	0.23

Logic / Arch	BUF (F = X)	INV (F = $\overline{X}$)	AND (F = XY)	NAND (F = $\overline{XY}$)	OR (F = X+Y)	NOR (F = $\overline{X+Y}$)	XOR (F = X⊕Y)	XNOR (F = $\overline{X⊕Y}$)
1								
2								
3								

Legend
Select = Pin A
0 = Pin B
1 = Pin C
Output = Pin F

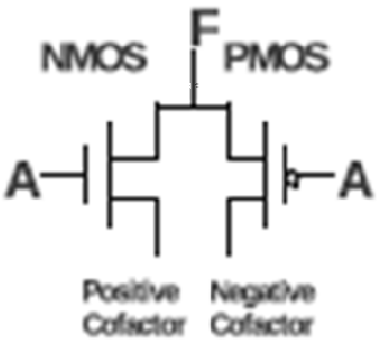
B) Tree of MUX (TMUX)

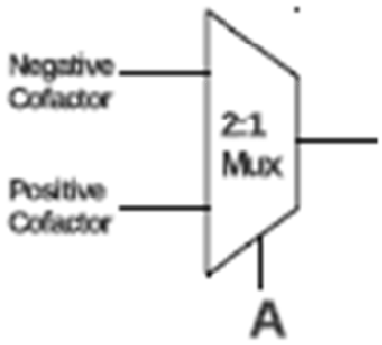
- **Input netlist is transformed to Binary Decision Diagrams (BDDs).**
  - Each node in BDD is a Multiplexer.
  - In CMOS technology, MUX gates are typically implemented with an ad-hoc structure based on transmission gates (PTL)
  - With graphene, RG-MUXes naturally implement multiplexers



x1	x2	x3	f
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

BDD Node Realization



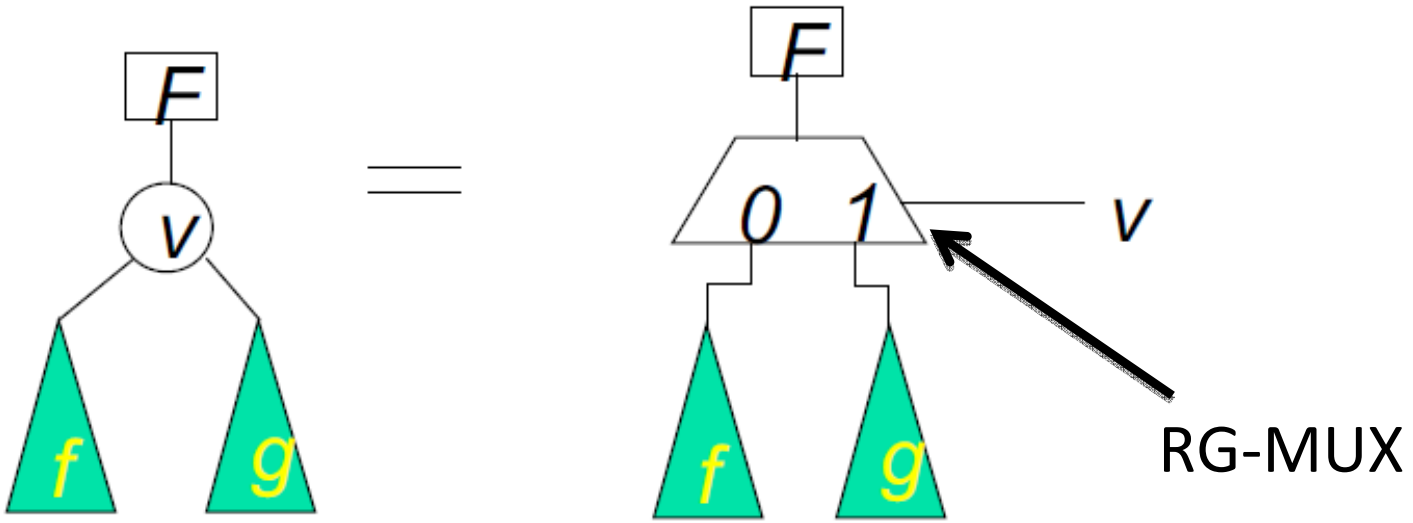


B) Tree of MUX (TMUX)

$$F(x_1 \dots, x_i, \dots, x_n) = x_i F(x_1 \dots, 1, \dots, x_n) + x_i' \; F(x_1 \dots, 0, \dots, x_n)$$

f g

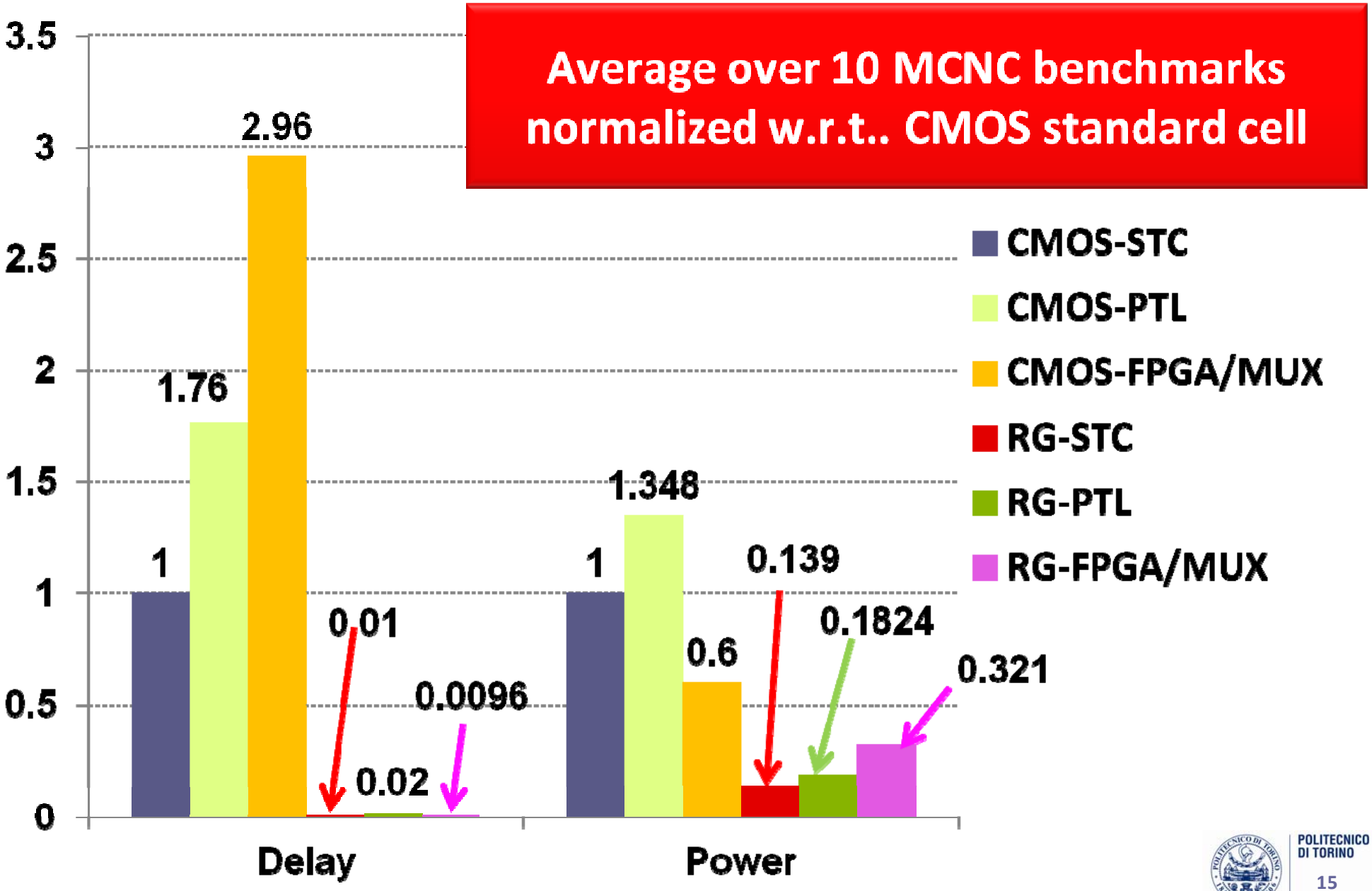
Recursive paradigm



C) FPGA/MUX

- **LUT-based FPGAs**
  - LUTs implemented as trees of multiplexers
- **CMOS implementation: PTL**
- **Graphene implementation: RG-MUX**

Preliminary Simulation Results

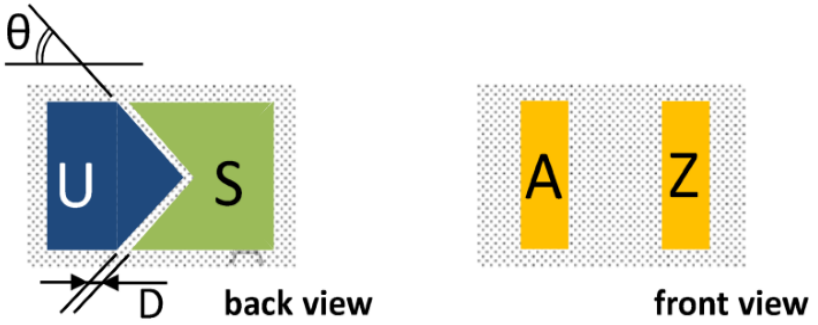


STC or PTL?

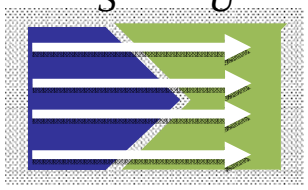
- **Graphene PN-junction looks a lot alike a pass gate**
  - Go for PTL
- **But, higher static power than STC**
  - Go for STC
- **Good things always stand in between…**
  - **Pass-XNOR Logic (PXL)**
    - Logic primitive: Pass-XNOR gate
    - Synthesis tool: Pass-Diagram + Gemini

Graphene PN-Junction as Logic Switch

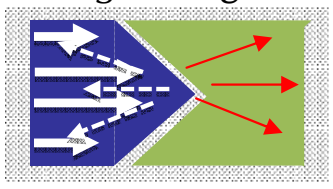
- **Voltages at the back-gates turn-ON/OFF the device**
  - U==S → R_{ON} → 1-logic
  - U!=S → R_{OFF} → 0-logic



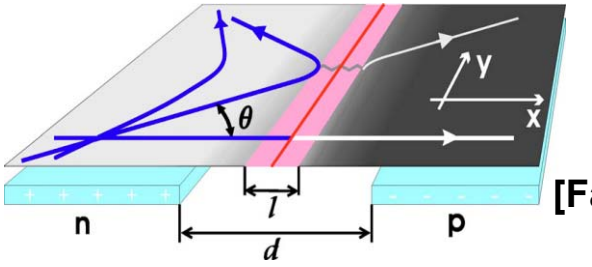
$V_S = V_U$



$V_S \neq V_U$



Leakage

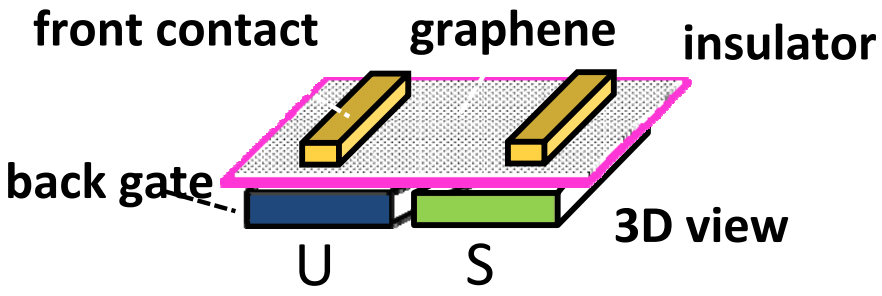


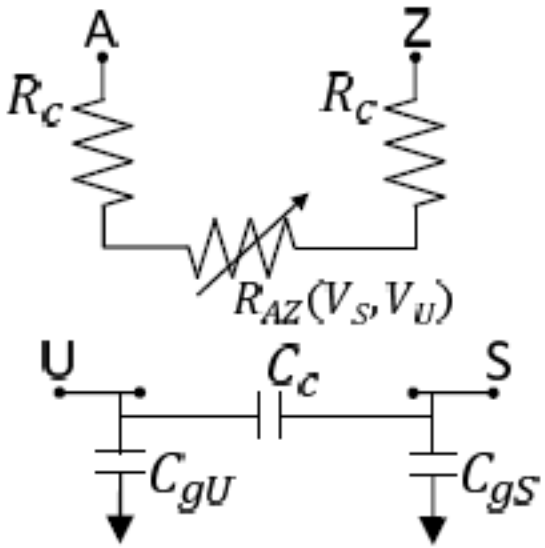
[Fal'ko06]

Electrical Model

A graphene PN-junction behaves as a voltage-controlled resistor whose resistance is inversely proportional to the junction transmission probability.

- R_C resistors represent the parasitic resistance of metal-to-graphene contacts;
- R_{AZ} models the resistive path across graphene between the input A and the output Z, function of V_S and V_U;
- C_C represents the coupling capacitance between the two metal split gates, and two lumped capacitances connected to the back-gates S and U, i.e., C_{gS} and C_{gU}, which consist of the series of the oxide capacitance and the quantum capacitance of the graphene sheet.

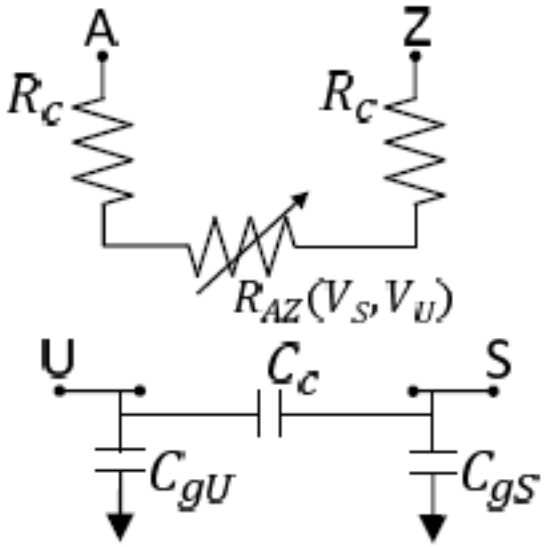




Verilog-A Model

Algorithm 1 Verilog-A code for the reconfigurable logic gate

```
1: `include ``disciplines.vams"
2: `include ``constants.vams"
3: module dev(A,B,C,F);
4: input A, B, C;
5: output F;
6: real Tox, d, Ef, pi, q, h, Vf, Y, E, EOT, Kf, Cox,
   Cq, T, Rnn, Rpn, Rc;
7: real Cc-in, Cc-out, Cg, Cin, A;
8: analog
9: begin
10:     Y = (4 * pi * q^2)/(h^2 * Vf^2);
11:     Ef = (sqrt(pow(E,2)+
12:         +(2 * Y * E * q * abs(V(a)) * EOT)) - E)/(Y * EOT);
13:     Kf = (2 * pi * abs(Ef))/(h * Vf);
14:     Rnn = (pi * h)/(4 * q^2 * w * Kf);
15:     Rpn = Rnn/T;
16:     Cox = E/Tox;
17:     Cq = Y * Ef;
18:     A = w^2 + 2 * w * d;
19:     Cg = A * (Cox * Cq)/(Cox + Cq);
20:     Cin = Cg + Cc-in;
21:     I(A) <+ Cin * ddt(V(A));
22:     if (V(A) ≥ 0)
23:       V(Y1) <+ V(C);
24:       V(Y2) <+ V(B);
25:     else
26:       V(Y1) <+ V(B);
27:       V(Y2) <+ V(C);
28:     end if
29:     V(Y3,Y1) <+ I(Y3,Y1) * (Rnn + Rc);
30:     V(Y3,Y2) <+ I(Y3,Y2) * (Rpn + Rc);
31:     V(Y3,F) <+ Rc * I(Y3,f);
32:     I(F) <+ Cc-out * ddt(V(F));
33: end
34: endmodule
```



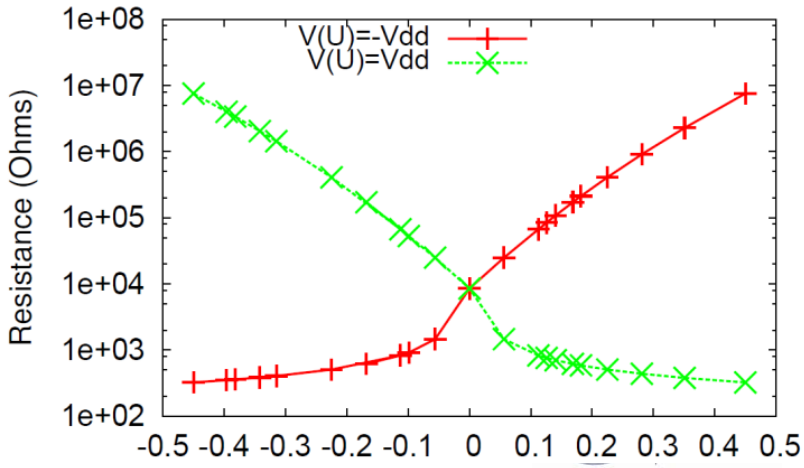
Parameters:

W = 194.5nm

T_{oxide}= 1.7nm

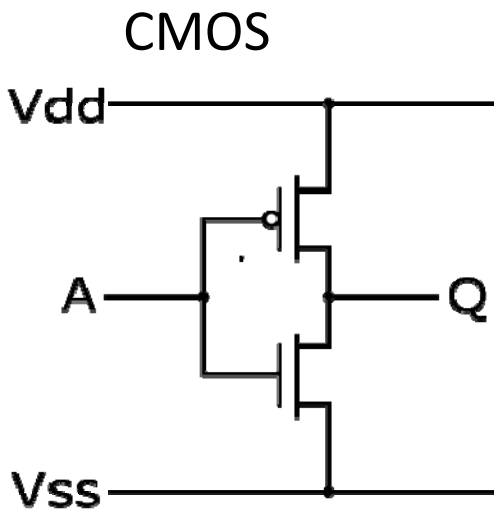
Area = 0.191µm^2

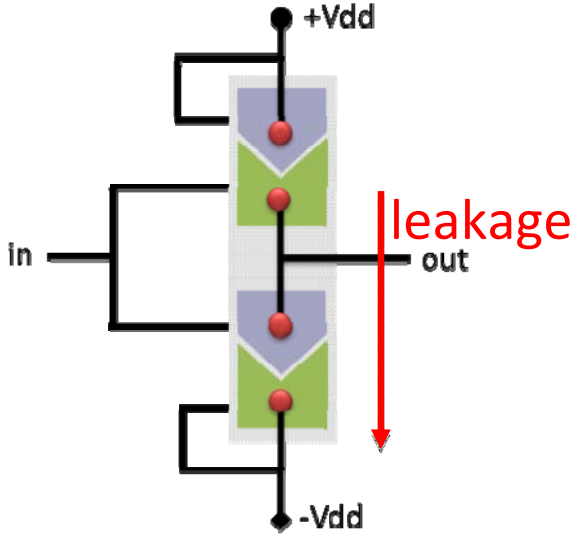
D = 18nmΘ = 45°



CMOS-like Static Implementation Style (CXL)

- **Using P-N junctions as transistors: not feasible**
  - High static power consumption due to leakage
  - Front-to-back connections



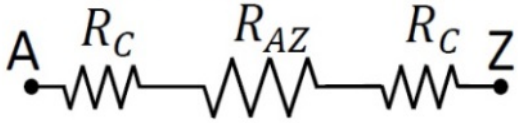


Exploit the concept of Dynamic Power Supply

Pass-XNOR Gate

- **Similar to MOS Transmission Gates**
  - Input logic signals drive the back-gates (electrostatic doping)
  - Front contacts propagate the «evaluation» signal
    - A function is evaluated as True if the input ramp passes through the gate and reach the output

DC electrical model



Pass EXclusive-NOR Gate

A	U	S	R_{AZ}	Z
Pulse V_{dd} 0V	0	0	R_{ON}	A
	0	1	R_{OFF}	Hi-Z
	1	0	R_{OFF}	Hi-Z
	1	1	R_{ON}	A

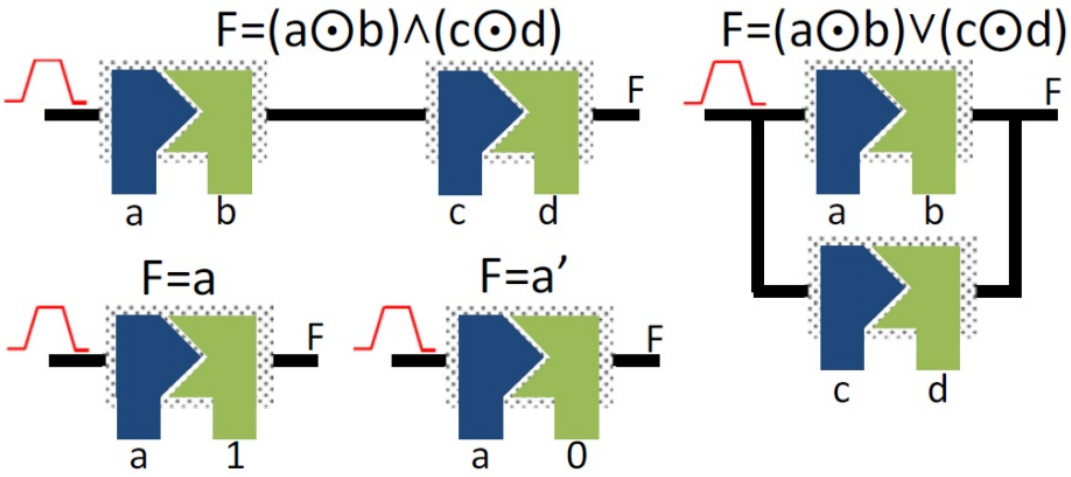
$$R_{Az} = \frac{R_0}{T_{AZ}(V_S, V_U, \vartheta = 45°)}$$

	CMOS XNOR	Graphene Pass-XNOR	Savings
[Width] μm	1.454	0.095	93.47 %
[Area] μm^2	2.116	0.191	90.97 %
[Delay] ns	0.37	0.08	78.37 %
[Static Power] μW	6.57	4	39.11 %
[Dynamic Power] μW	195.66	183.05	6.44 %
[Energy Delay Product] $fJ \cdot ns$	72.39	14.65	79.77 %

An energy-efficient new primitive with High Expressive-Power

Building Complex Functions with Pass-XNOR Gates

- **Pass-XNOR Logic (PXL)**
  - Implement product/sum of Exclusive-NORs
    - Product→Series
    - Sum →Parallel
    - Identity/Complement → one back-gate @ 1/0-logic



	# In	# Out	PXL	CXL
xnor9	9	1	9	35
xnor11	11	1	11	51
9symml	9	1	26	78
parity	16	1	18	91
misex1	8	7	38	55
Avg.			**20.4**	**62**

More area efficiency w.r.t. CMOS-like implementation styles

PXL vs. CXL

- **Experiments on a set of 46 logic functions (averaged results)**

	#devices	Area [µm^2]	Delay [ns]	P Leak [µA]	Dyn. Power [µW]
PXL	2.74	0.52	8.9	10.96	219.57
CXL	5.48	1.05	50.44	102.7	444.89

	Function		Function
F00	$\bar{a}$	F23	$a + (b \odot d) \cdot c$
F01	$a \odot b$	F24	$(a \odot d) + (b \odot d) \cdot c$
F02	$a + b$	F25	$a + (b \odot d) \cdot (c \odot d)$
F03	$a \cdot b$	F26	$(a \odot d) + ((b \odot d) \cdot (c \odot d))$
F04	$(a \odot b) + c$	F27	$(a \odot d) \cdot b \cdot c$
F05	$(a \odot b) \cdot c$	F28	$(a \odot d) \cdot (b \odot d) \cdot c$
F06	$(a \odot b) + (a \odot c)$	F29	$(a \odot d) \cdot (b \odot d) \cdot (c \odot d)$
F07	$(a \odot b) \cdot (a \odot c)$	F30	$(a \odot d) + (b \odot e) + c$
F08	$(a \odot b) + (c \odot d)$	F31	$(a \odot d) + (b \odot d) + (c \odot e)$
F09	$(a \odot b) \cdot (c \odot d)$	F32	$((a \odot d) + (b \odot e)) \cdot c$
F10	$a + b + c$	F33	$((a \odot d) + b) \cdot (c \odot e)$
F11	$(a + b) \cdot c$	F34	$((a \odot d) + (b \odot d)) \cdot (c \odot e)$
F12	$a + (b \cdot c)$	F35	$((a \odot d) + (b \odot e)) \cdot (c \odot d)$
F13	$a \cdot b \cdot c$	F36	$(a \odot d) + ((b \odot e) \cdot c)$
F14	$(a \odot d) + b + c$	F37	$a + ((b \odot d) \cdot (c \odot e))$
F15	$(a \odot d) + (b \odot d) + c$	F38	$(a \odot d) + ((b \odot e) \cdot (c \odot e))$
F16	$(a \odot d) + (b \odot d) + (c \odot d)$	F39	$(a \odot d) + ((b \odot e) \cdot (c \odot d))$
F17	$((a \odot d) + b) \cdot c$	F40	$(a \odot d) \cdot (b \odot e) \cdot c$
F18	$((a \odot d) + (b \odot d)) \cdot c$	F41	$(a \odot d) \cdot (b \odot d) \cdot (c \odot e)$
F19	$((a \odot d) + b) \cdot (c \odot d)$	F42	$(a \odot d) + (b \odot e) + (c \odot f)$
F20	$((a \odot d) + (b \odot d)) \cdot (c \odot d)$	F43	$((a \odot d) + (b \odot e)) \cdot (c \odot f)$
F21	$(a + b) \cdot (c \odot d)$	F44	$(a \odot d) + ((b \odot e) \cdot (c \odot f))$
F22	$(a \odot d) + (b \cdot c)$	F45	$(a \odot d) \cdot (b \odot e) \cdot (c \odot f)$

Higher leakage currents
Higher delays
Higher dynamic power

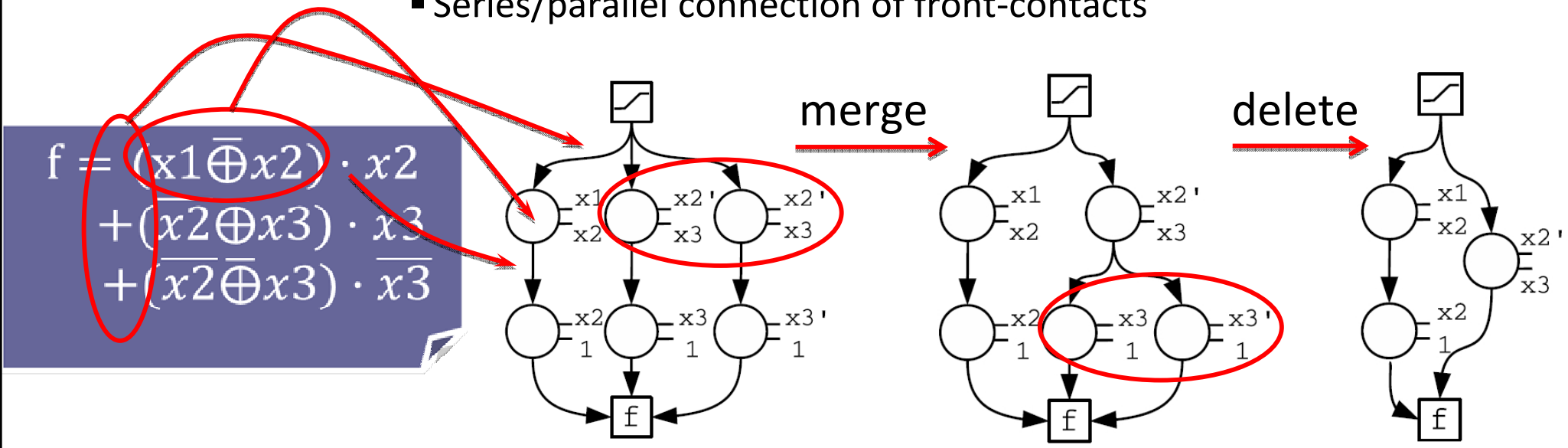
Lack of commercial logic-synthesis tools for the PXL!

Gemini: A PXL Synthesis Tool

- **One-Pass Synthesis**
  - Takes as input the implicant table of the logic function
  - Returns a minimum area/delay PXL circuit implementation
  - Main strengths
    1. PXL-oriented data structure for concurrent logic optimization and circuit mapping
       - Pass Diagrams (PDs), instead of BDDs, better match the final circuit implementation
       - Standard reduction rules used for BDDs still hold
    2. Table-based EXNOR-expansion using Local Variable Ordering
       - Global Variable Ordering drastically affects the cardinality of the data-structure

PDs & Minimization Rules

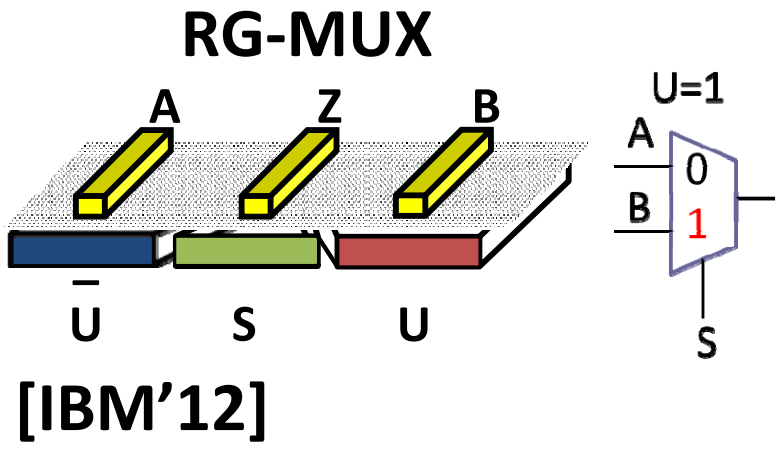
- **Pass Diagram PD = DAG(V, E)**
  - Polarized directed acyclic graphs
    - Root→source signal that evaluates the logic fucntion
    - Sink→output logic function
  - Internal nodes represent EXNOR of 2 input variables
    - 1-to-1 mapping between nodes and p-n junctions
    - Control variables connected to the p-n junctions' back-gates
  - Edges represent circuit topology
    - Series/parallel connection of front-contacts



Area/Delay Estimation

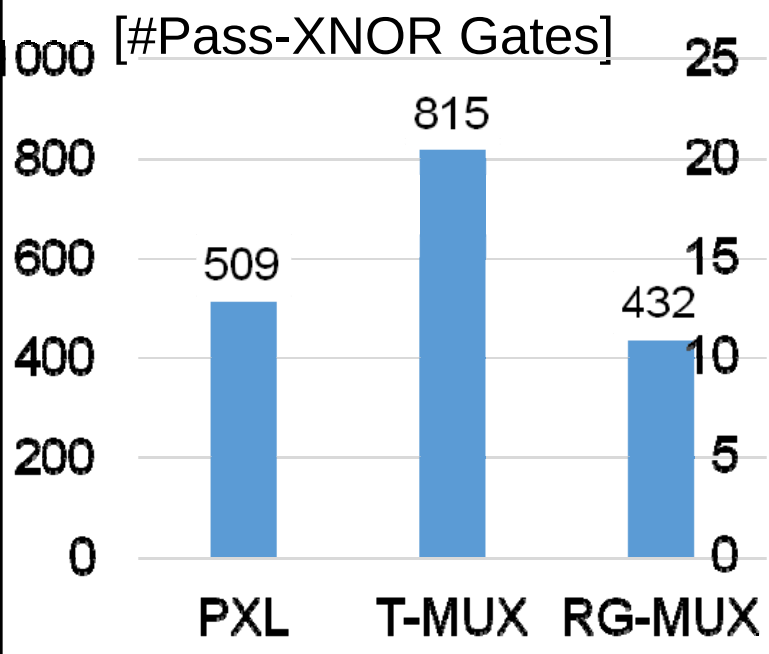
- **Combinational Benchmarks**
- **Experimental setup**

1. PXL circuits + Pass Diagrams
2. Tree-of-MUXes (T-MUX) + BDDs
3. RG-MUX + Multi-level Synthesis

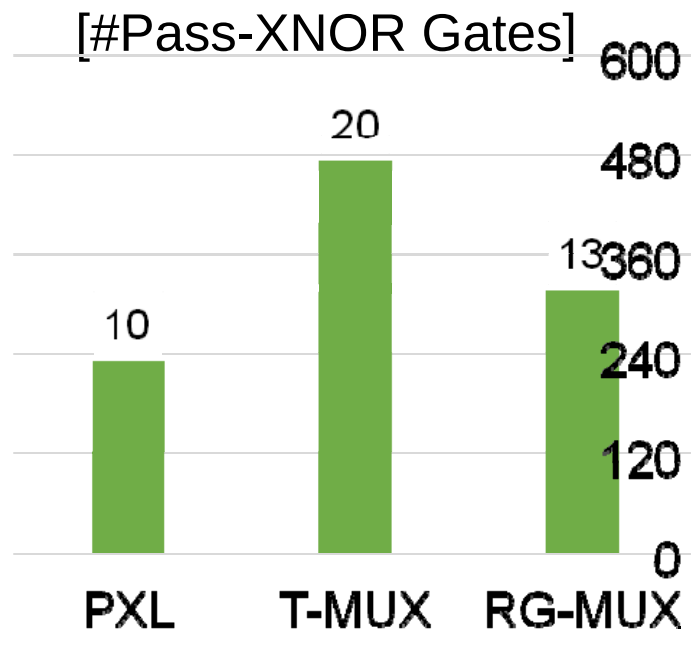


[IBM'12]

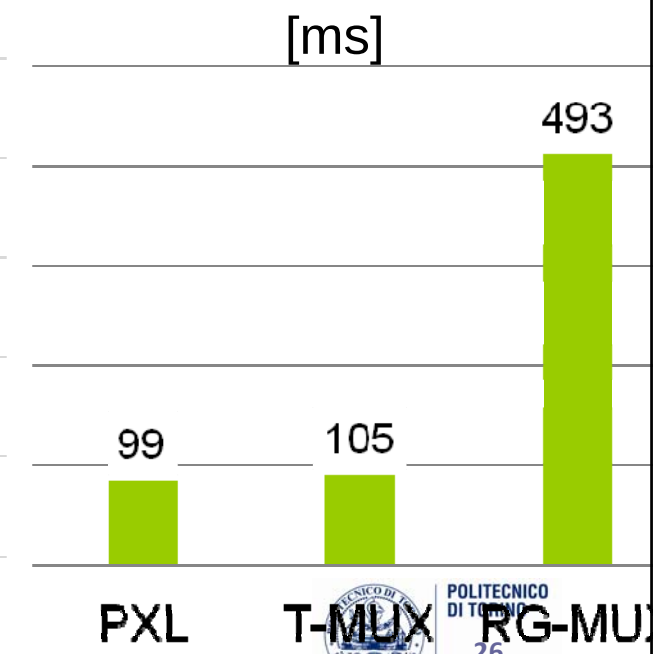
Average Area



Average Longest Path Depth



Average CPU time



Conclusions & Final Remarks

- **Graphene and 2D materials may become one of the technological vehicles for the next generation of ICs**
- **The lack of band-gap in graphene is a serious concern if one wants to implement digital circuits as we do with silicon**
- **Other strategies are possible**
  - Pass-XNOR Logic is a viable solution
    - High Expressive-Power (less devices, more function)
- **Possible application of PXL**
  - Flexible sensors with embedded computing features for data pre-processing, e.g., sensor fusion and context recognition
    - Reduce data transmission and data transfer to CPU
    - Less memory usage
    - Improve energy efficiency

Publications

1. S. Miryala, A. Calimera, E. Macii, M. Poncino, L. Bolzani, “Investigating the behavior of physical defects in pn-junction based reconfigurable graphene devices,” in IEEE *LATW’13*, 2013, pp. 1-6.

2. S. Miryala, A. Calimera, E. Macii, M. Poncino, “Delay Model for Reconfigurable Logic Gates Based on Graphene PN-junctions,” in *GLSVLSI’13*, 2013, pp. 227–232.

3. S. Miryala, A. Calimera, E. Macii, M. Poncino, “Power modeling and characterization of graphene-based logic gates,” in *PATMOS*, 2013, pp. 223–226.

4. S. Miryala , A. Calimera, E. Macii, M. Poncino, “Exploration of different implementation styles for graphene-based reconfigurable gates,” in *ICICDT’13*, 2013, pp. 21–24.

5. S. Miryala, M. Montazeri, A. Calimera, E. Macii, M. Poncino, “A Verilog-A Model for Reconfigurable Logic Gates Based on Graphene pn-Junctions,” in *DATE’13*, 2013, pp. 877–880.

6. V. Tenace, A. Calimera, E. Macii, M. Poncino, “Quantifying the figures of merit of graphene-based adiabatic Pass-XNOR Logic (PXL) circuits,” in *IEEE PRIME’14*, 2014, pp. 1–4.

7. V. Tenace, A. Calimera, E. Macii, M. Poncino, “Pass-XNOR Logic: A new Logic Style for PN-Junction based Graphene Circuits,” in *DATE’14*, 2014, pp. 1–4.

8. V. Tenace, A. Calimera, E. Macii, M. Poncino, “One-Pass Logic Synthesis for Graphene-Based Pass-XNOR Logic Circuits,” in *DAC’15,* pp.1-4.