Near-*Vt* RRAM-based FPGA

Opportunity for Low-Power Computing

Xifan Tang

Pierre-Emmanuel Gaillardon and Giovanni De Micheli

Dec. 12th, 2014

International **C**onference on **F**ield **P**rogrammable **T**echnology (**ICFPT**) , Shanghai

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

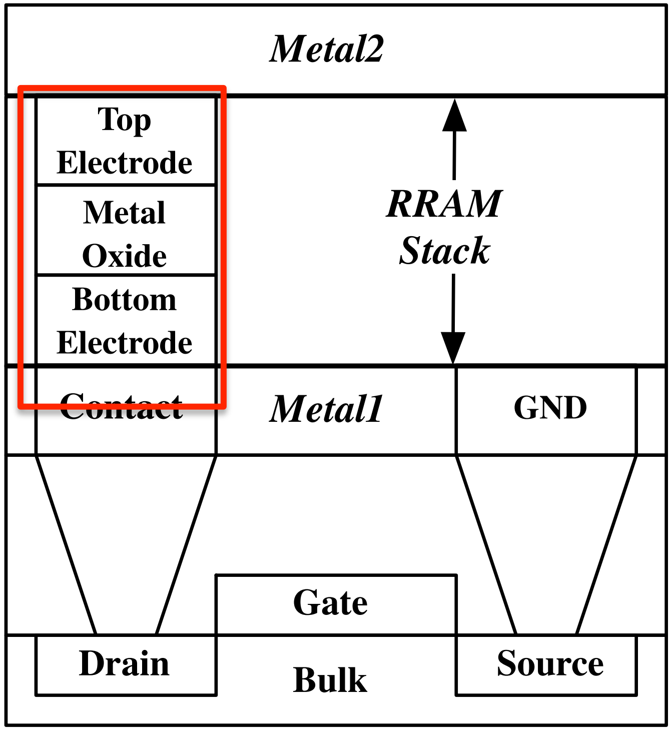
LSI Integrated Systems Laboratory

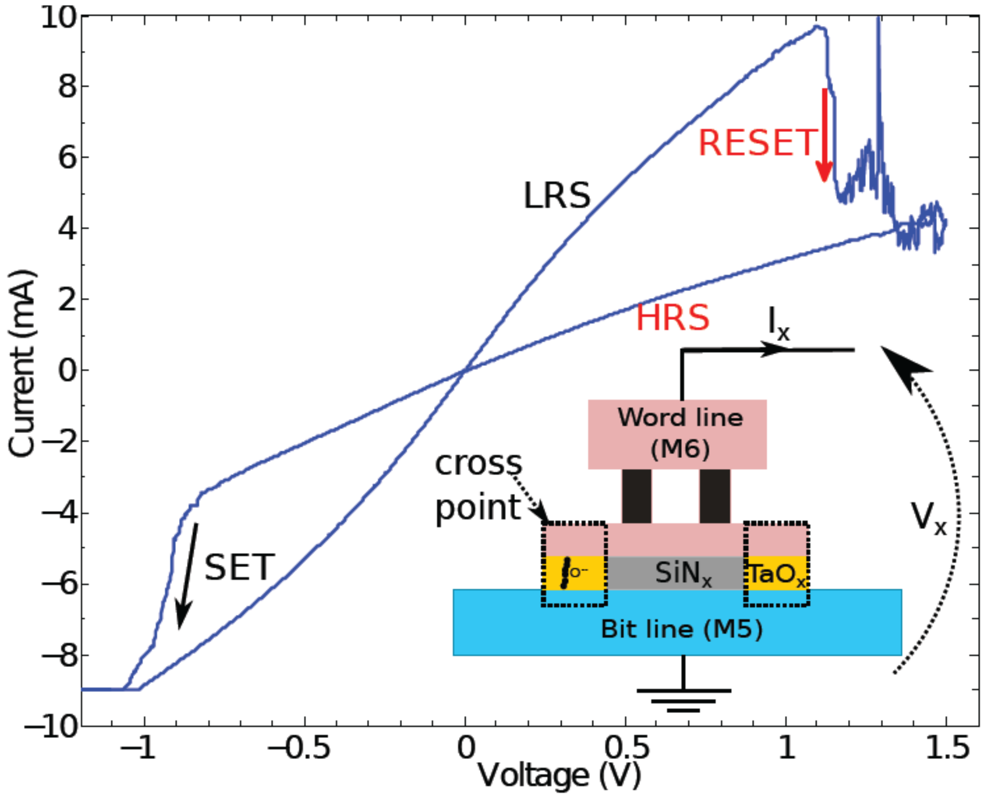
Motivation

- Challenges
  - Tight power budget
  - Heavy routing architecture
  - Volatility
- Key Contributions
  - Study Near-Vt RRAM-based FPGA
  - Size the transistors in RRAM-based FPGA

Resistive Random Access Memory (RRAM)

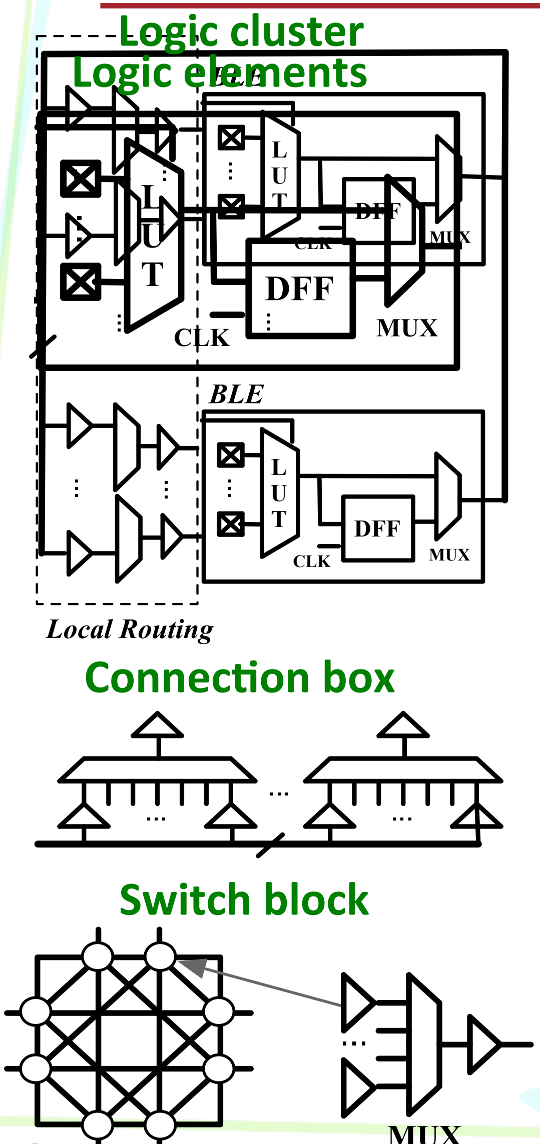
- Non-volatile
- Low/High Resistance States(LRS/HRS)



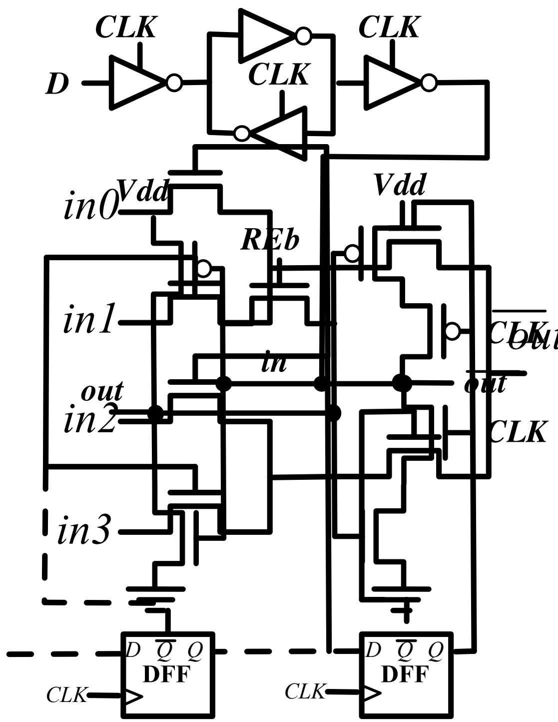


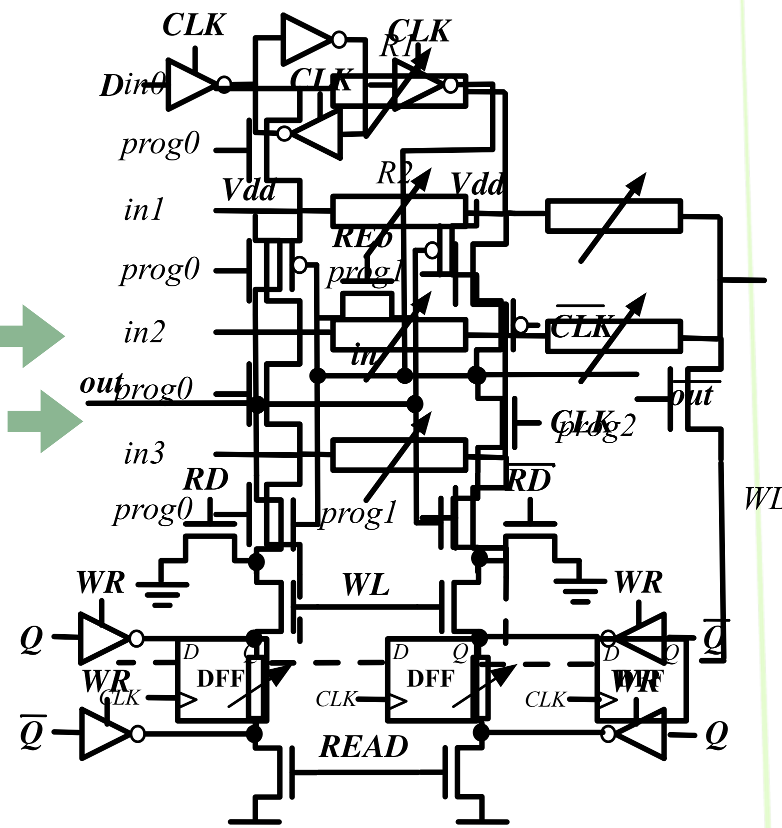
[1] J. Sandrini *et al.*, *Heterogenenous Integration of ReRAM Crossbars in a CMOS Foundry Chip*, Published in 40th International Micro and Nano Engineering Conference (MNE).

Part I: Near-Vt RRAM-based FPGA



- SRAM → NV-SRAM
- MUX → NV-MUX



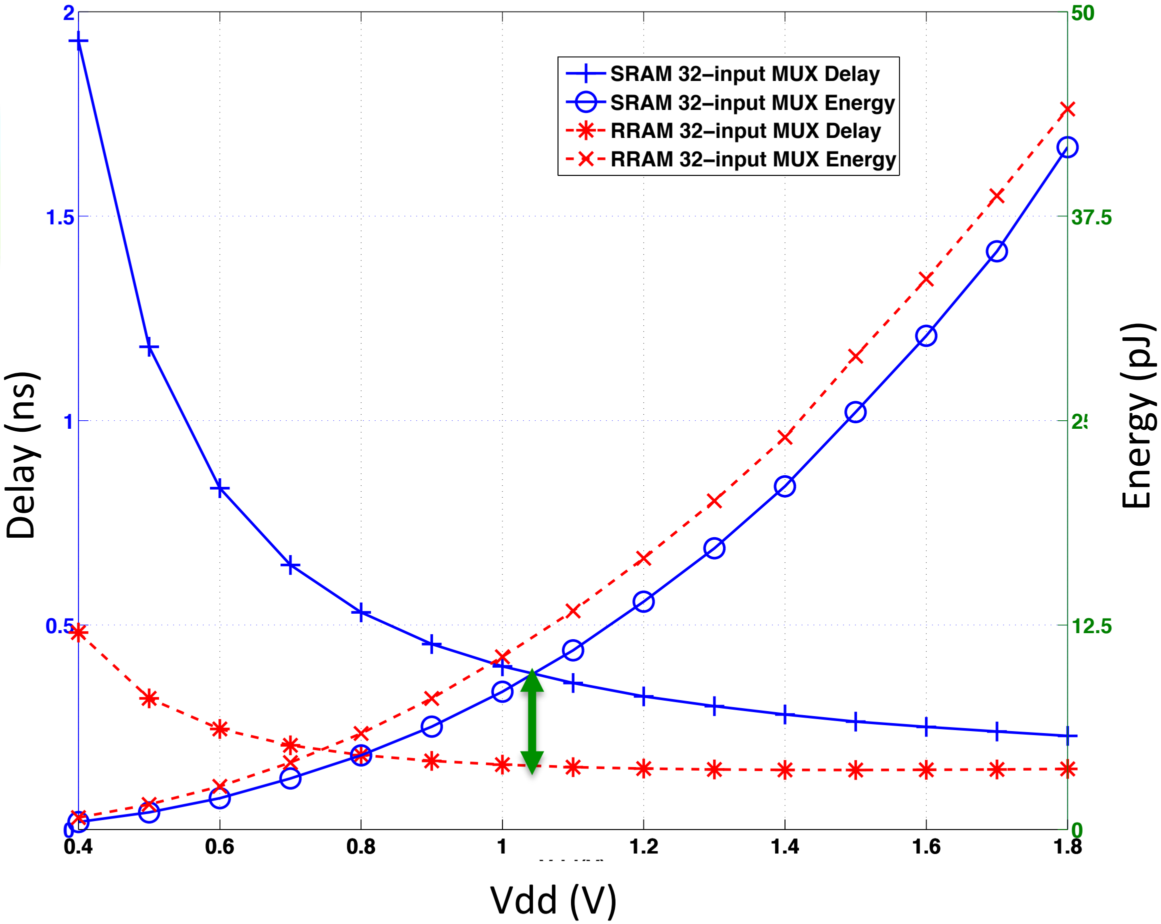


[1] I. Kazi *et al.*, *Energy/Reliability Trade-Offs in Low-Voltage ReRAM-Based Non-Volatile Flip-Flop Design*, IEEE TCAS-I, vol. 61, No. 11, pp. 3155-3164.

[2] P.-E. Gaillardon *et al.*, *GMS: Generic Memristive Structure for Non-Volatile FPGAs*, IEEE/IFIP *Int. Conf.* on VLSI-SoC, 2012, pp. 94-98.

Impact of Vdd on RRAM Routing Elements

Multiplexer: RRAM vs. SRAM



RRAM parameters: [1]
$R_{on} = 1\ k\text{Ohm}$,
$R_{off} = 1\ M\text{Ohm}$.

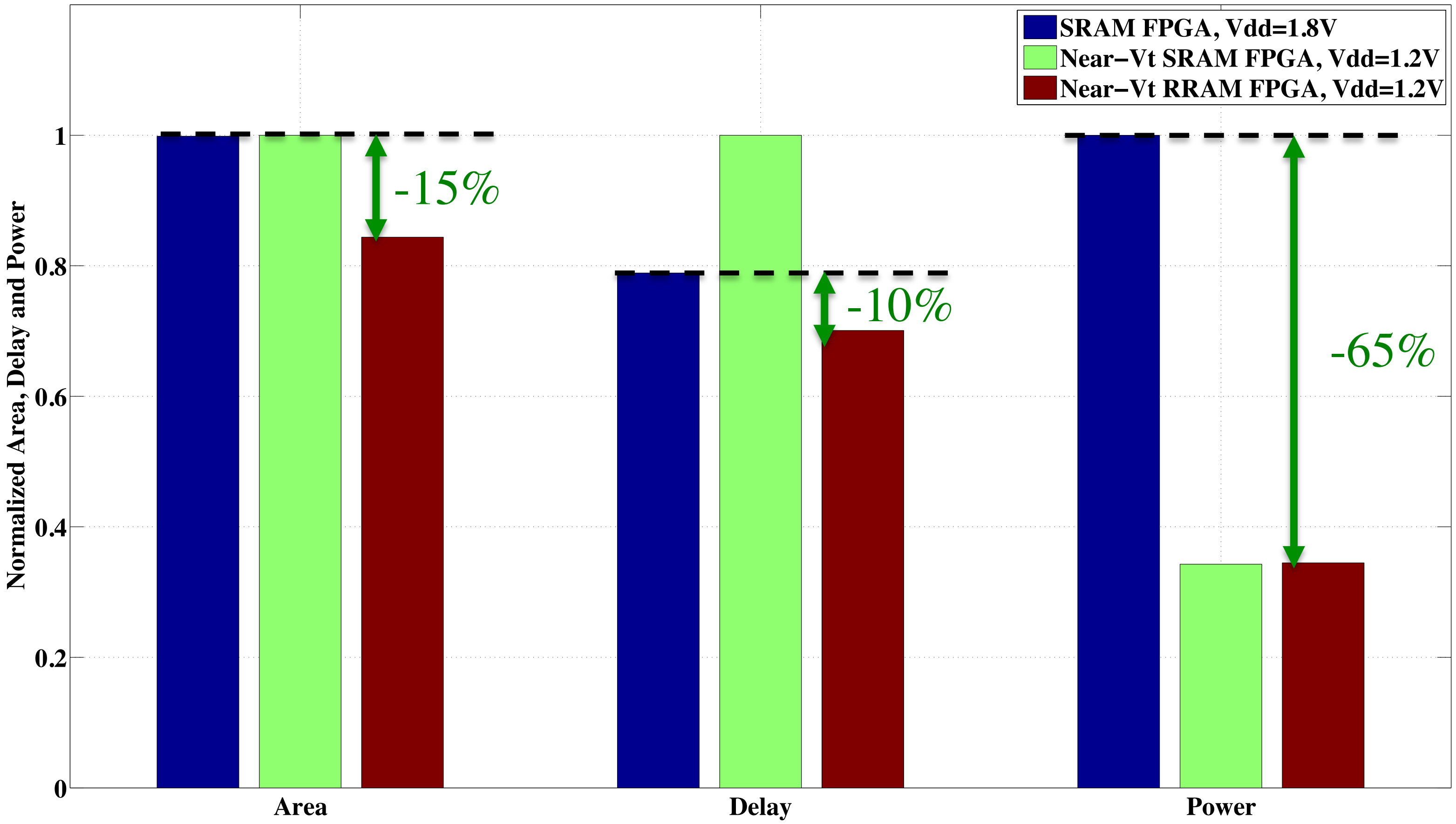
- ▲ High performance
- ▲ Power reduction
- ▲ Area-saving

$$V_{prog} = R_{on} \cdot I_{prog} = R_{on} \cdot W_{prog} \cdot I_d$$

[1] D. Sacchetto *et al.*, *Application of Multi-Terminal Memristive Devices: A Review*, IEEE CAS Magazine, Vol. 13, No. 2, pp. 23-41.

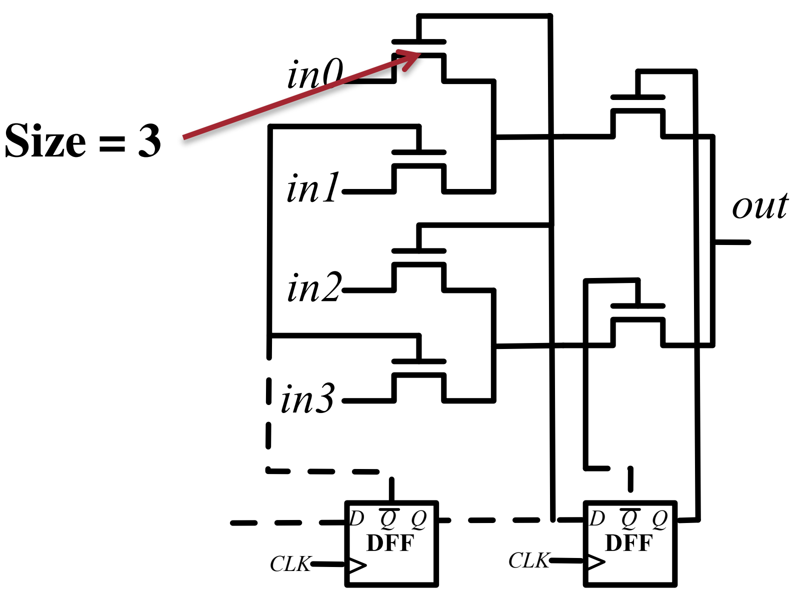
Near-Vt RRAM-based FPGA

- Methodology: VTR flow.
- Baseline Architecture: K=6, N=10, I=33, UMC 180nm Technology

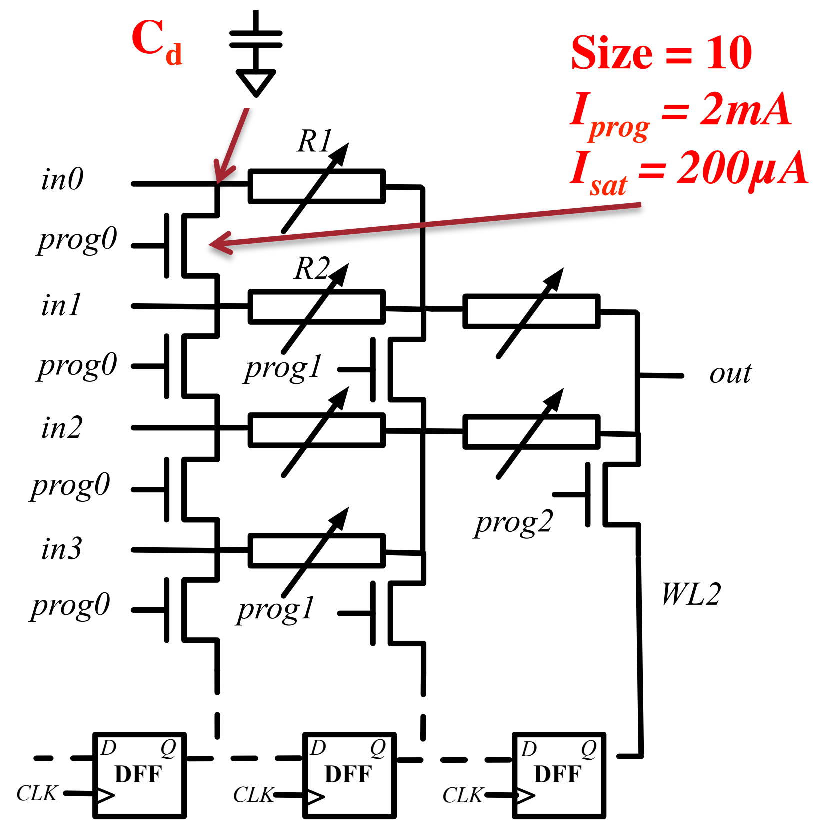


Part II: Transistor Sizing

- Non-negligible programming transistor size
  - Area Overhead
  - Parasitic Capacitance



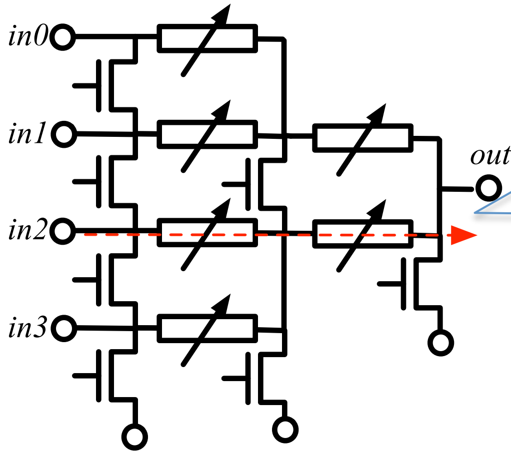
SRAM-based MUX

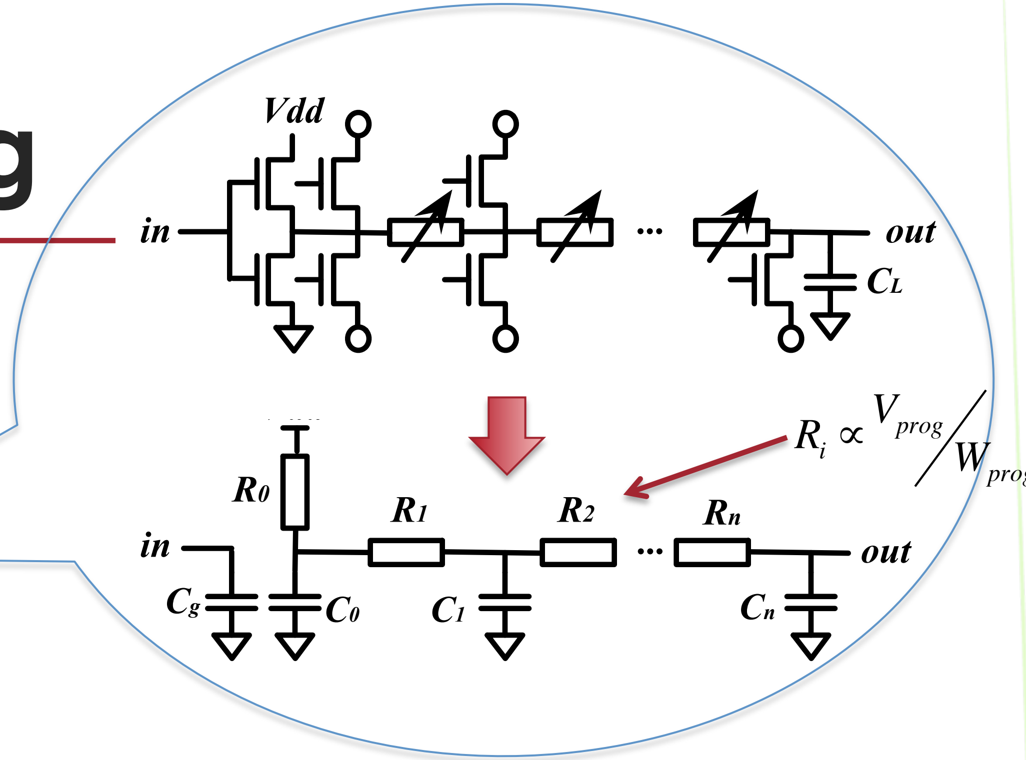


RRAM-based MUX

Transistor Sizing

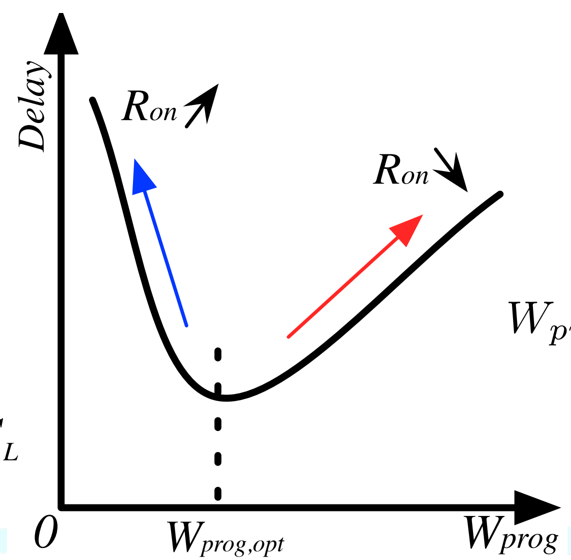
▲ RC Modeling





▲ Elmore Delay

$$
\begin{aligned}
\tau &= \sum_{i=0}^{n} R_i \sum_{j=i}^{n} C_j \\
&= R_{\min} C_{\min} + \frac{R_{\min}}{W_{inv}} C_L \\
&+ (2n+1) \frac{R_{\min}}{W_{inv}} W_{prog} C_{off} + n \cdot R_{on} C_L \\
&+ n^2 R_{on} W_{prog} C_{off}
\end{aligned}
$$



▲ Optimal W_{prog}

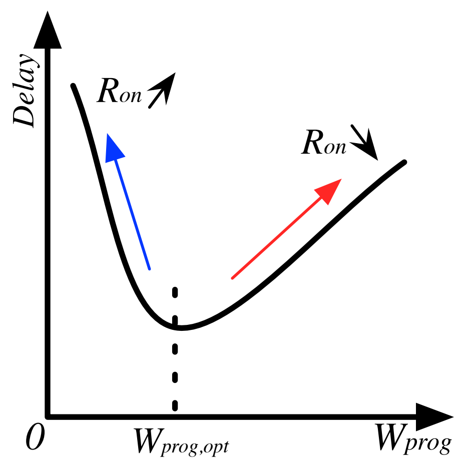
$$
W_{prog,opt} = \sqrt{\frac{n \boxed{V_{prog}} \boxed{C_L} W_{inv}}{(\boxed{2}n + 1) I_d R_{min} C_{off}}}
$$

Impact of Supply Voltage

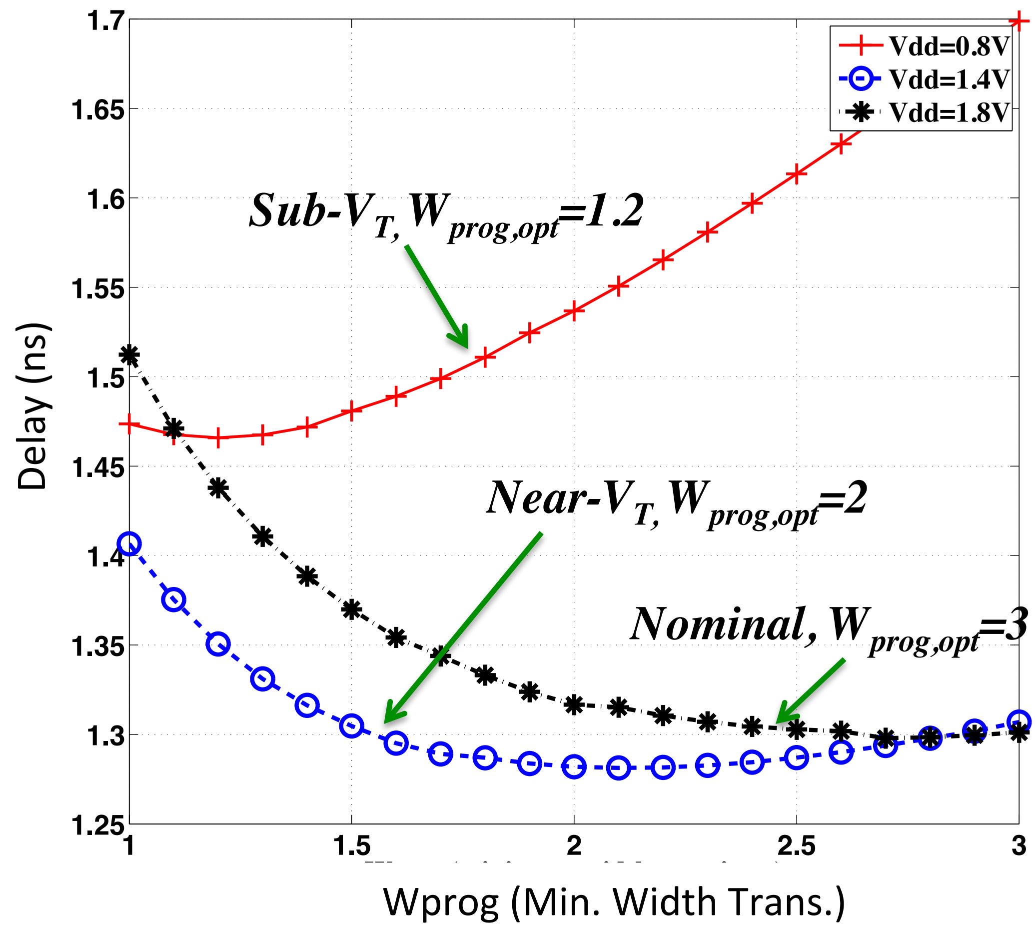
$$V_{prog} \propto Vdd$$

$$W_{prog,opt} = \sqrt{\frac{n\boxed{V_{prog}}C_L W_{inv}}{(2n+1)I_d R_{min} C_{off}}}$$

$V_{prog} \uparrow \quad W_{prog,opt} \uparrow$



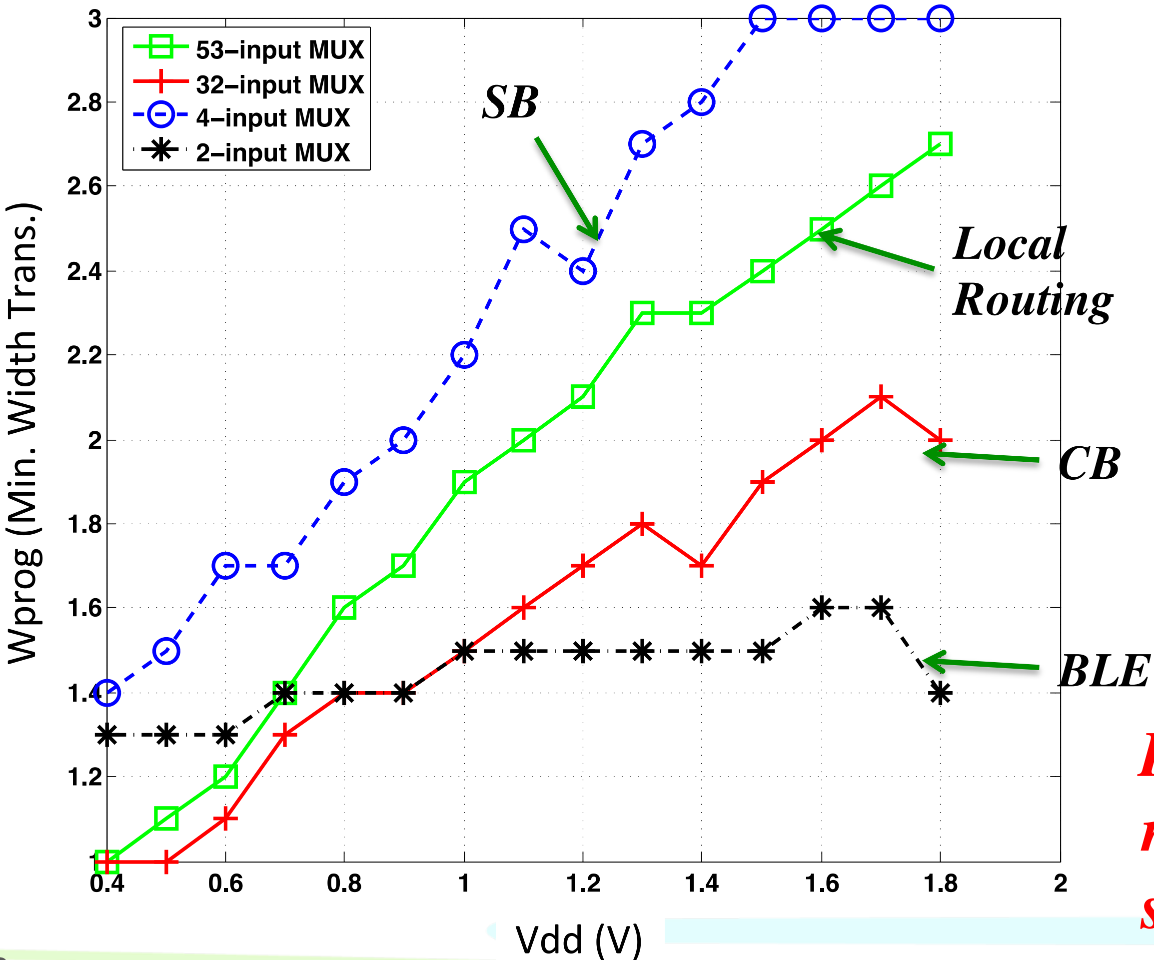
RRAM-based 32-input Multiplexer



Near-V_T produces even better delay with smaller transistor !

Study MUXes in FPGA

$$W_{prog,opt} = \sqrt{\frac{\boxed{n}V_{prog}\boxed{C_L}W_{inv}}{(\boxed{2n}+1)I_d R_{min} C_{off}}}$$



▲Impact of n

$n \uparrow \quad W_{prog,opt} \uparrow$

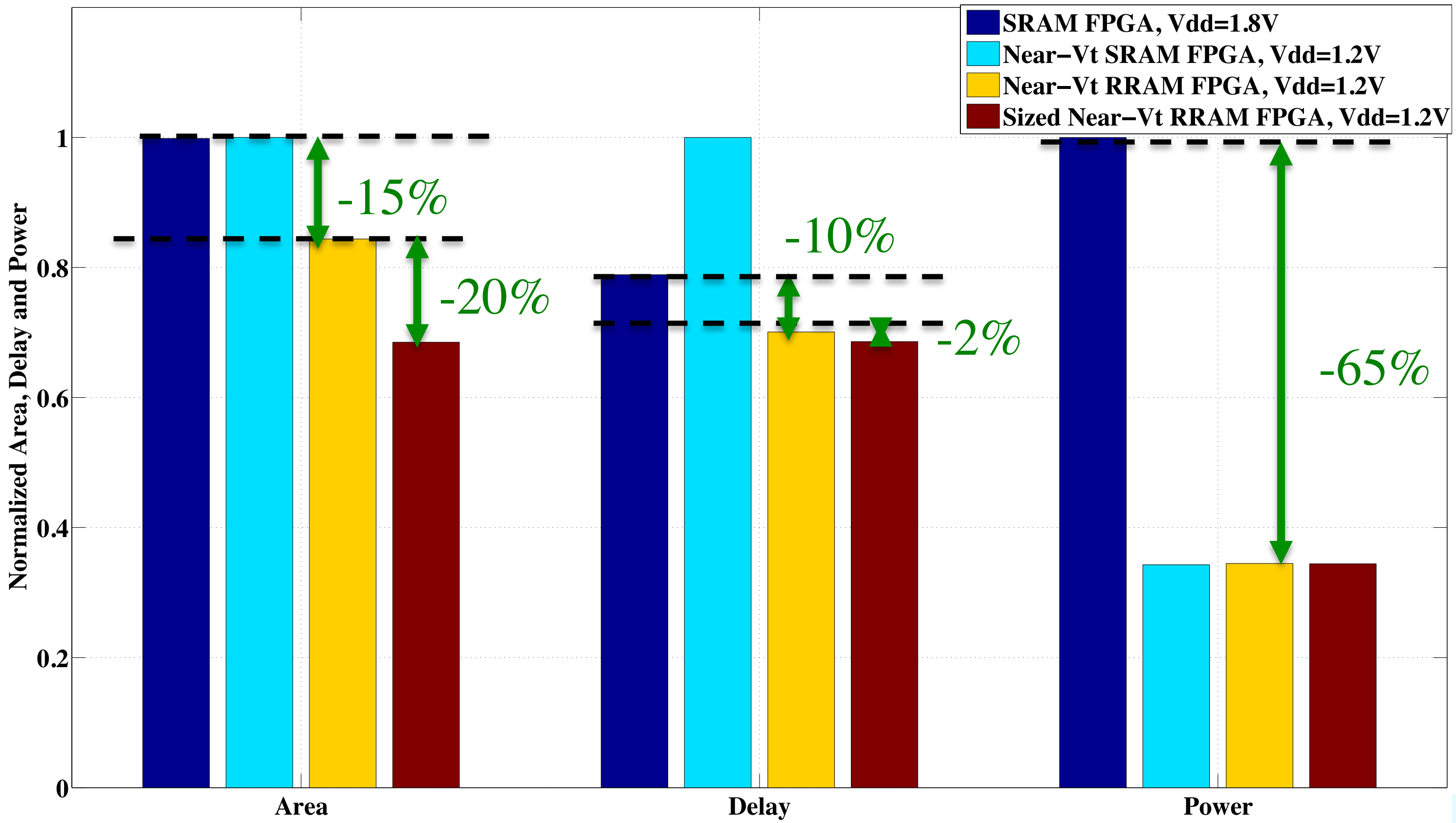
▲Impact of C_L

$C_L \uparrow \quad W_{prog,opt} \uparrow$

RRAM MUXes in FPGA need to be non-uniformly sized!

Sized Near-Vt RRAM FPGA

- Methodology: VTR flow.
- Baseline Architecture: K=6, N=10, I=33, UMC 180nm Technology



Conclusion

- ▲ Contribution I: Near-Vt RRAM-based FPGA
  - ▽ a low-power circuit without performance degradation
- ▲ Contribution II: Improved area efficiency
  - ▽ Non-uniformly sized routing transistors

Q&A !
Thanks.