Scaling Trends for Dual-Rail Logic Styles against Side-Channel Attacks: a Case-Study

Kashif Nawaz, Dina Kamel,
François-Xavier Standaert, Denis Flandre

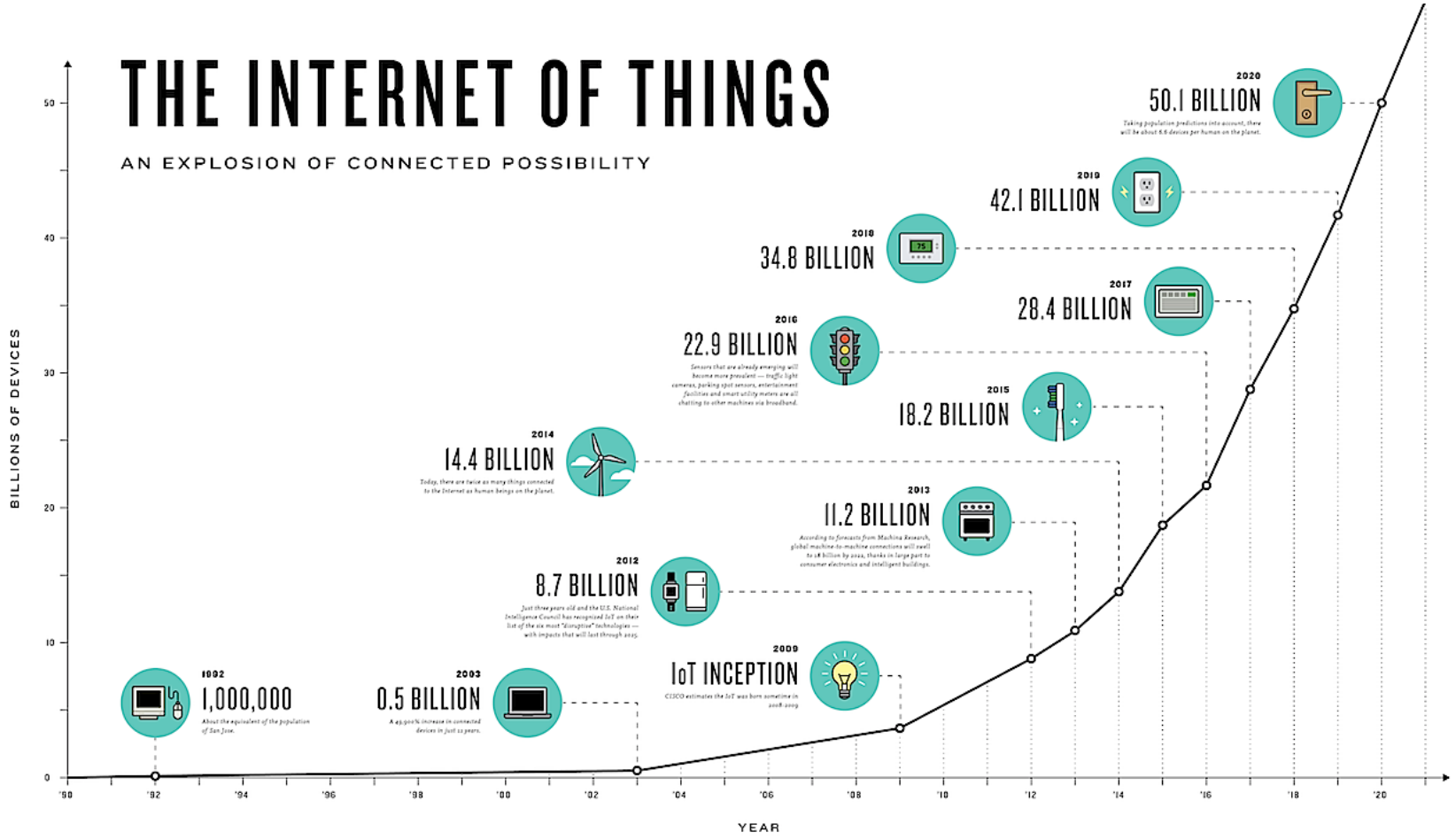
UCL Crypto Group, Belgium

COSADE, April 2017

Outline

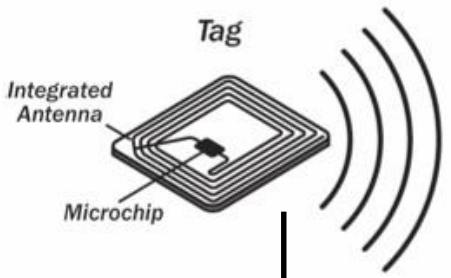
- Motivation
- Shrink but Think…
- Countermeasures against SCA
- Contributions
- Performance evaluation methodology
- Security evaluation methodology
- Simulation Parameters
- Security analysis
- Conclusion

Internet of things



Courtesy: i-scoop.com

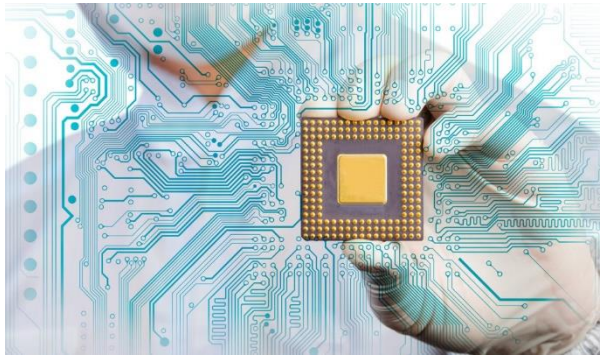
Side-Channel Attacks



Side channel attack

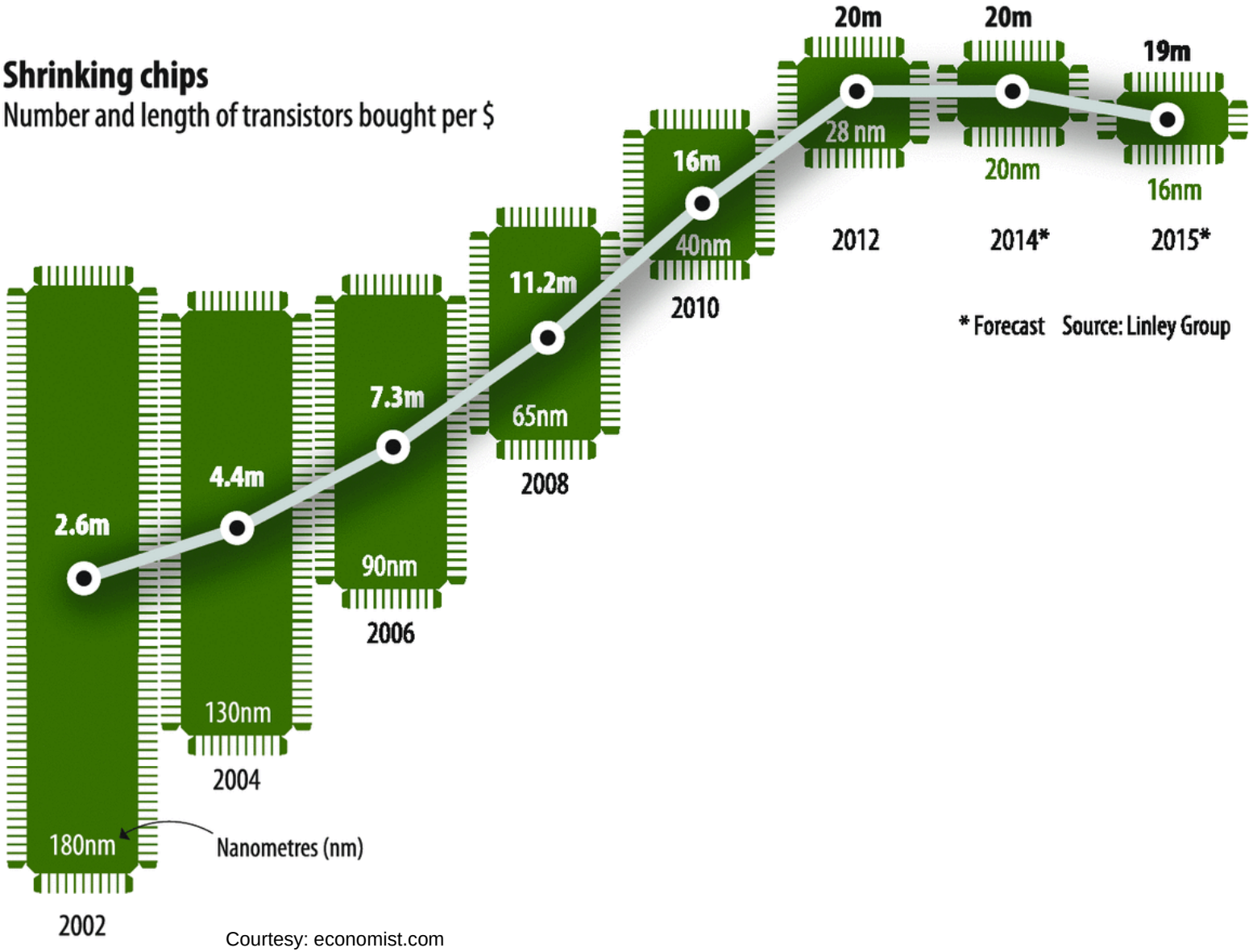
Courtesy: imimg.com

Power Analysis based attacks
Electromagnetic based attacks



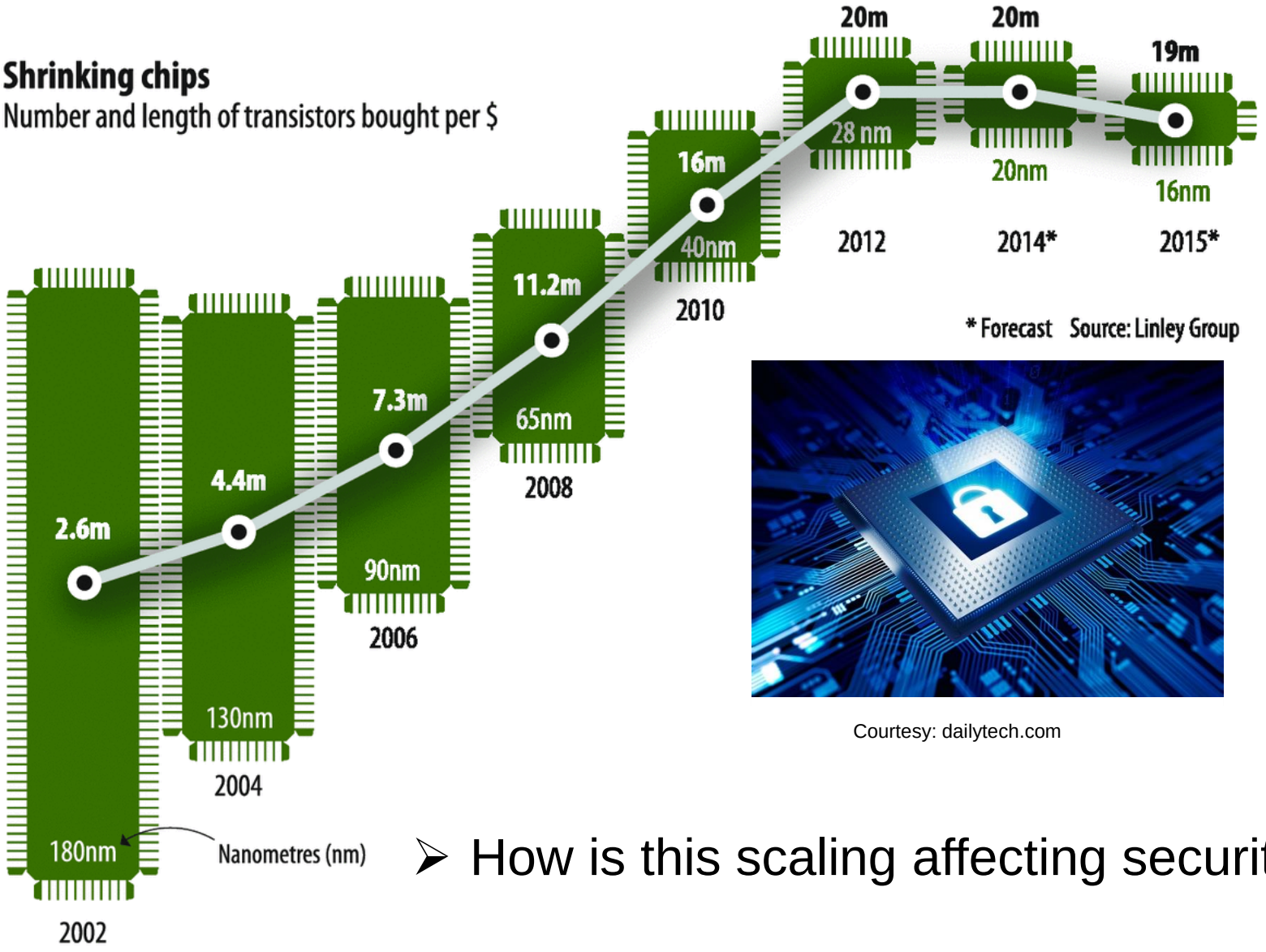
Courtesy: venturebeat.com

To Moore’ and Beyond…



Courtesy: economist.com

Scaling – but Thinking security



Courtesy: dailytech.com

- How is this scaling affecting security?

Side-Channel Countermeasures

$$\frac{S}{N}$$

Side-Channel Countermeasures

$$\frac{S}{N} \rightarrow \frac{S}{N}$$

Side-Channel Countermeasures

$$\frac{S}{N}$$

$$\frac{S}{N}$$

$$s$$

- Filtering
- Equalization
- Decorrelation
- …

Side-Channel Countermeasures

$$\frac{S}{N} \rightarrow \frac{S}{N} \leftarrow N$$

- Additive noise (algorithmic)
- Randomization
- …

Side-Channel Countermeasures

$$\frac{S}{N}$$

$$\frac{S}{N}$$

$$s$$

$$N$$

- Filtering
- Equalization
- Decorrelation
- …

- Additive noise (algorithmic)
- Randomization
- …

Hardware countermeasures

Side-Channel Countermeasures

$$\frac{S}{N} \rightarrow \frac{S}{N} \rightarrow \frac{s}{N}$$

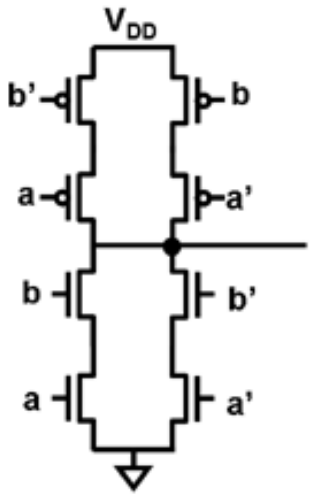
- Filtering
- Equalization
- Decorrelation
- …

- Additive noise (algorithmic)
- Randomization
- …

Hardware countermeasures

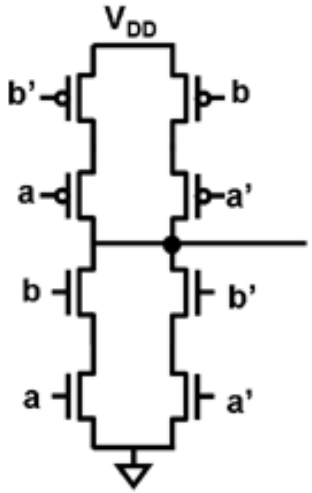
Thanks to dual-rail logic, we deal with signal reduction and achieve lower signals compared to CMOS…

CMOS vs Dual-rail logic-Schematic & Costs



XOR
CMOS

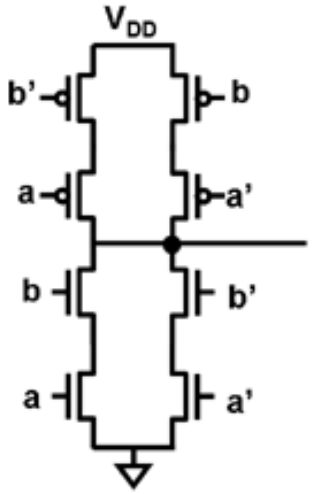
CMOS vs Dual-rail logic-Schematic & Costs



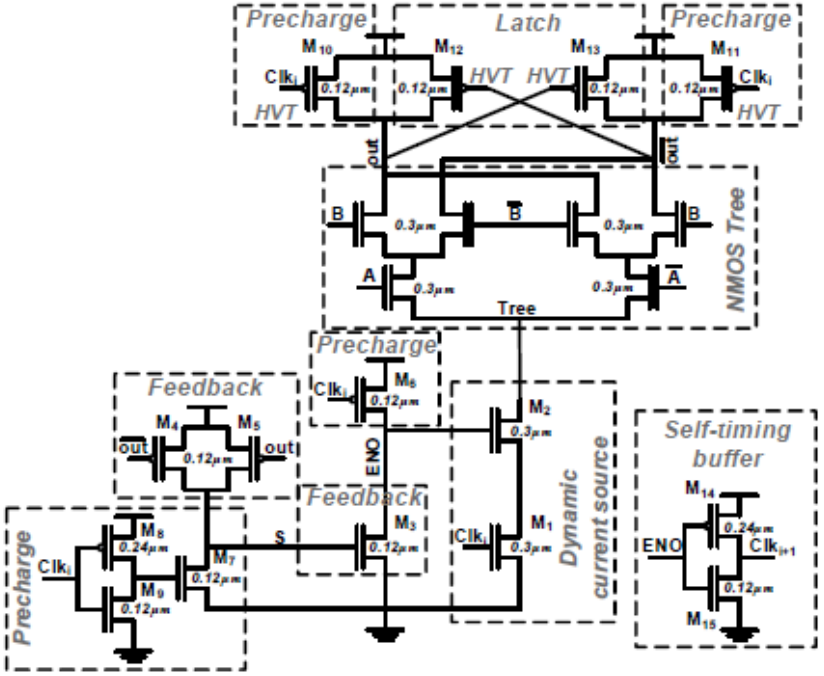
XOR
CMOS

- Easier to implement
- Lesser area overhead
- Faster design time
- Can be implemented using standard cell libraries
- Vulnerable to side-channel attack as power consumption is dependent on manipulated data

CMOS vs Dual-rail logic-Schematic & Costs



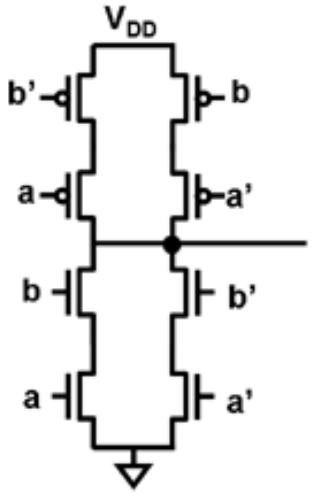
XOR
CMOS



DDSLL XOR

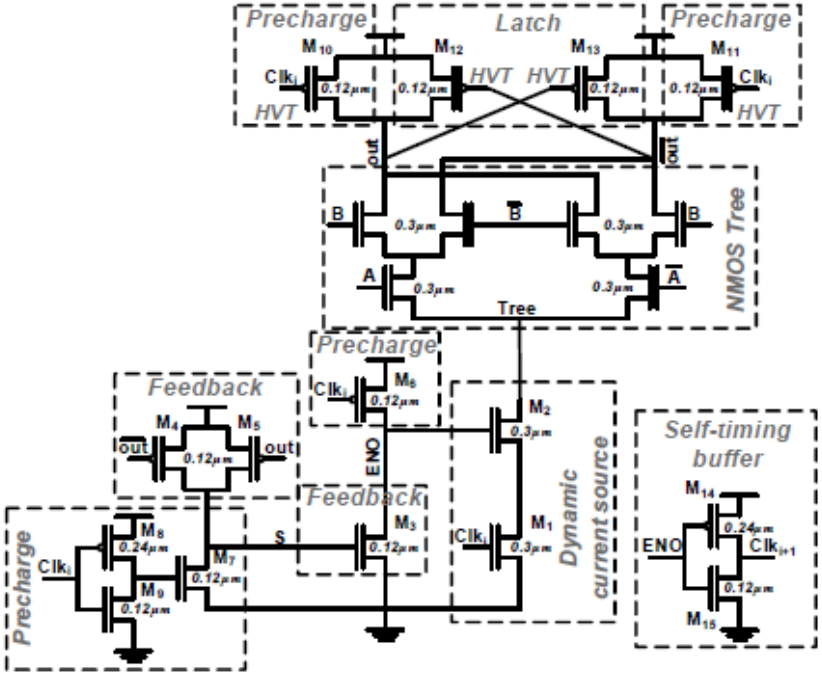
- Easier to implement
- Lesser area overhead
- Faster design time
- Can be implemented using standard cell libraries
- Vulnerable to side-channel attack as power consumption is dependent on manipulated data

CMOS vs Dual-rail logic-Schematic & Costs



XOR
CMOS

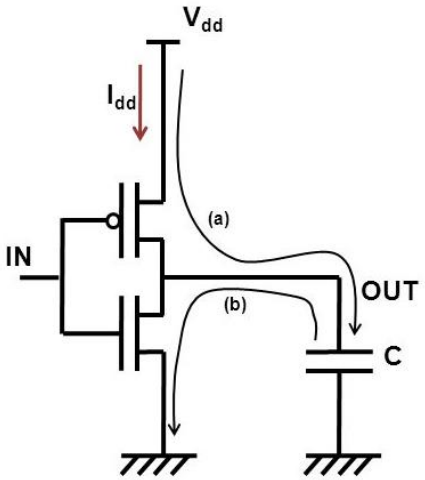
- Easier to implement
- Lesser area overhead
- Faster design time
- Can be implemented using standard cell libraries
- Vulnerable to side-channel attack as power consumption is dependent on manipulated data

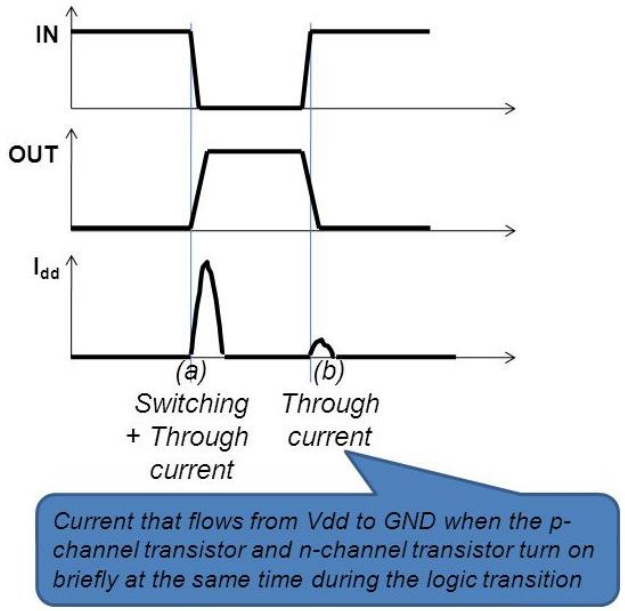


DDSLL XOR

- Full custom design
- Larger area overhead
- Requires additional circuitry like buffers, precharge clocks and feedback loop
- Consumes slightly lesser power compared to CMOS
- Ideally equalized power consumption irrespective of manipulated data

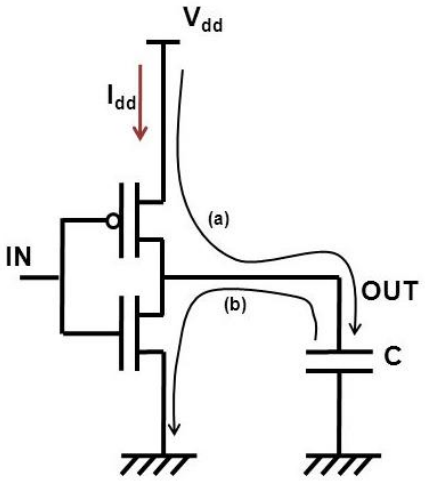
CMOS vs Dual-rail logic Operation

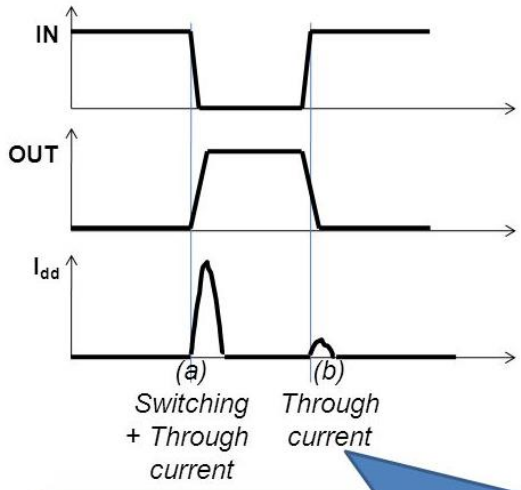




I_{DD} in CMOS

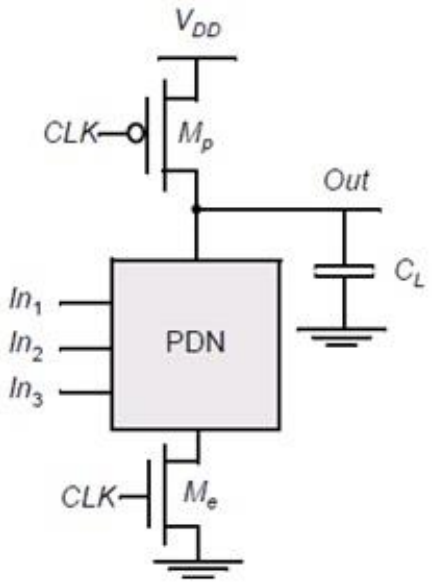
CMOS vs Dual-rail logic Operation



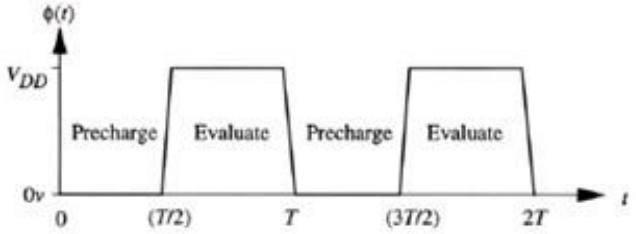


I_{DD} in CMOS

Current that flows from Vdd to GND when the p-channel transistor and n-channel transistor turn on briefly at the same time during the logic transition

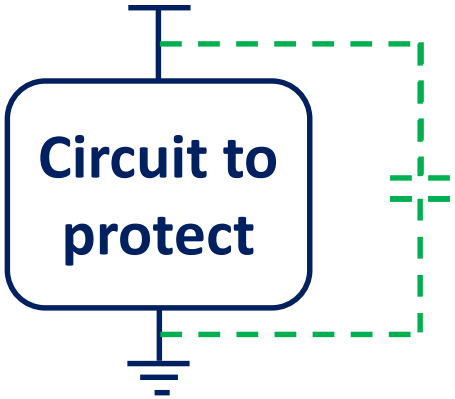


Phase	Clk	Inputs	Outputs
Precharge	Low	Don't care	High
Evaluation	High	Valid	Valid

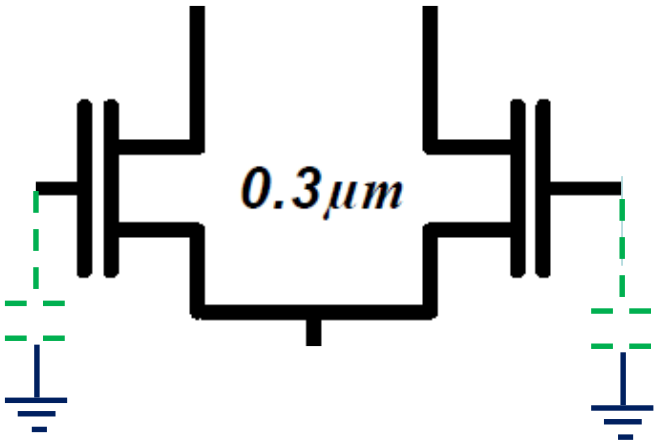


I_{DD} in Dual-rail

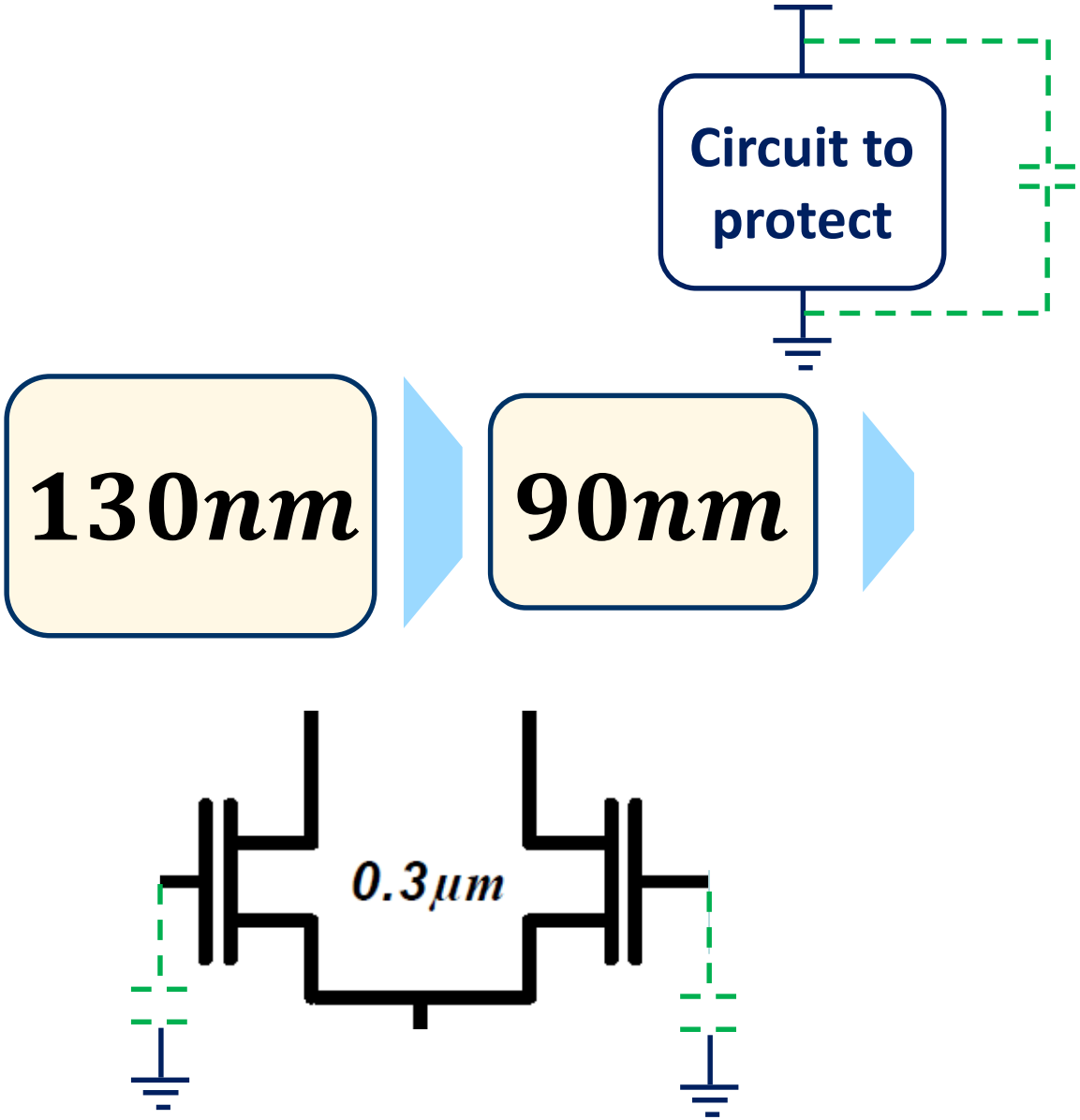
Scaling down: CMOS & Dual-rail logic



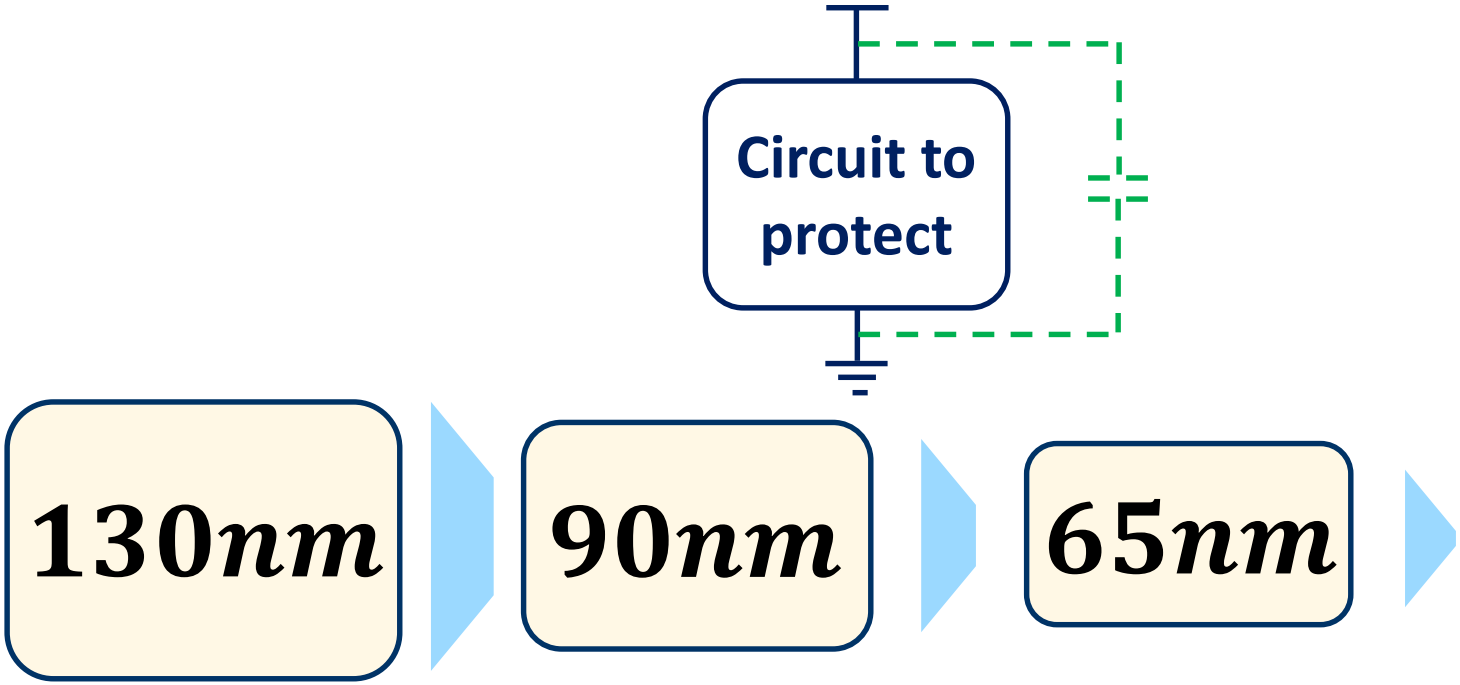
$130nm$

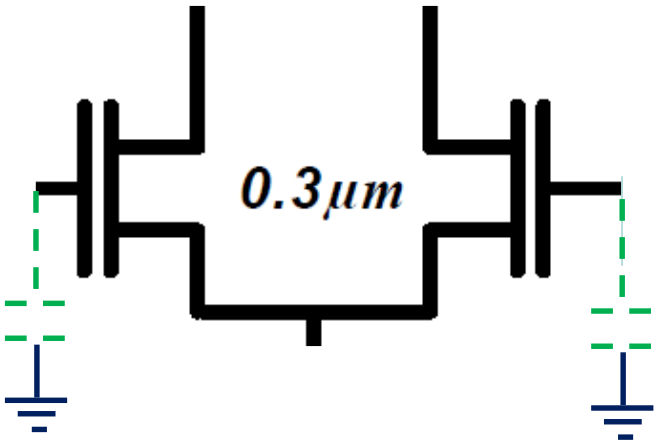


Scaling down: CMOS & Dual-rail logic

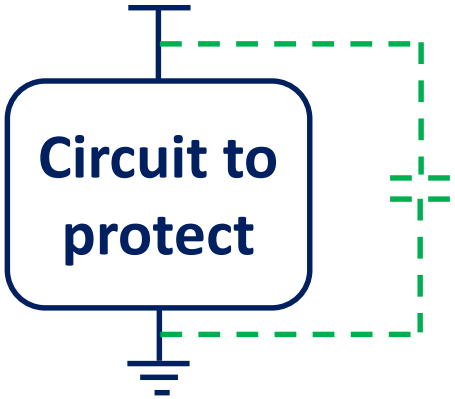


Scaling down: CMOS & Dual-rail logic

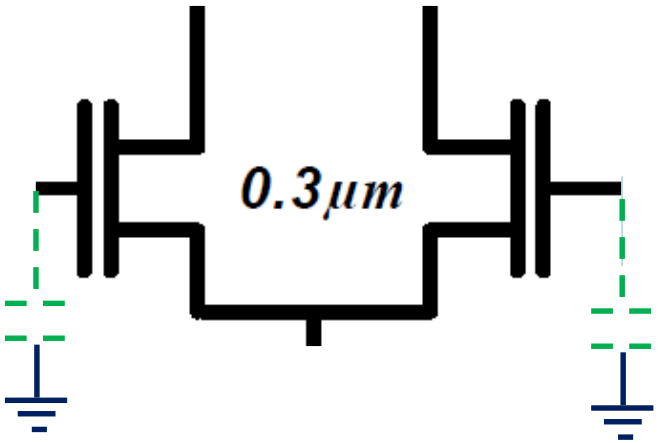


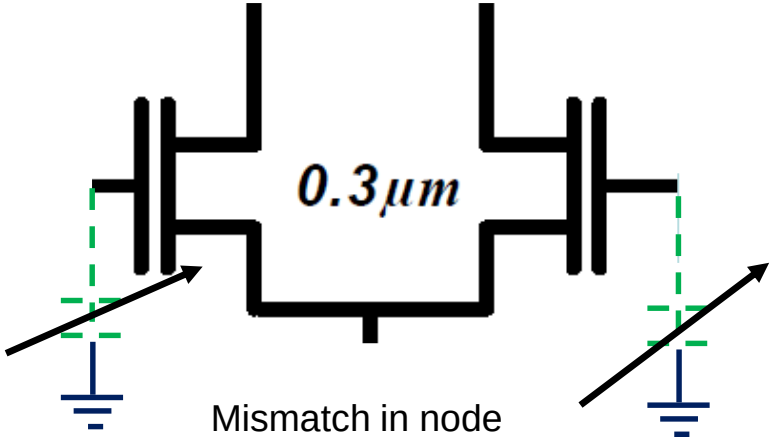


Scaling down: CMOS & Dual-rail logic



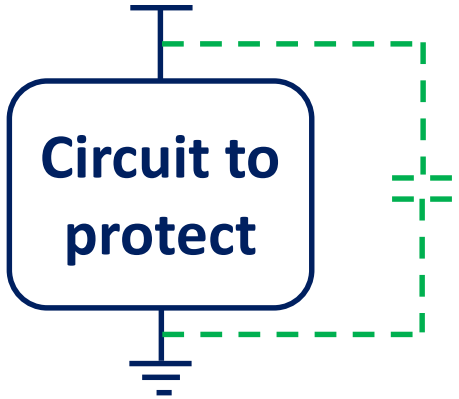
$130nm$ → $90nm$ → $65nm$ → $28nm$





Mismatch in node caps

Scaling down: CMOS & Dual-rail logic

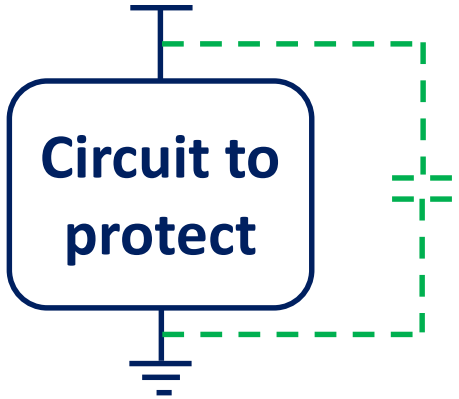


130*nm* ▶ **90*nm*** ▶ **65*nm*** ▶ **28*nm***

DDSLL provided a security level of x10 compared to CMOS in 65nm [15]

Imbalances in capacitances (dual-rail logics) lead to data dependencies which become exploitable [34,19,26,29,16]

Scaling down: CMOS & Dual-rail logic



130*nm* ▶ **90*nm*** ▶ **65*nm*** ▶ **28*nm***

DDSLL provided a security level of x10 compared to CMOS in 65nm [15]

Imbalances in capacitances (dual-rail logics) lead to data dependencies which become exploitable [34,19,26,29,16]

▶ *Does it get **worse** with scaling?*

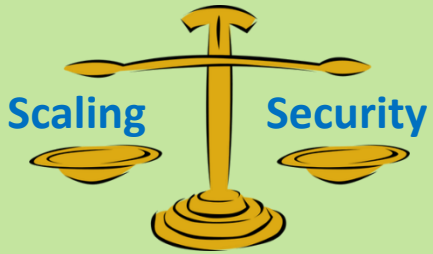
Contributions

$\frac{S}{N}$ | CMOS compared to Dual rail at 65nm

$\frac{S}{N}$ | CMOS compared to Dual rail at 28nm

Implementation

Evaluation



Contributions

$\frac{S}{N}$ | CMOS compared to Dual rail at 65nm

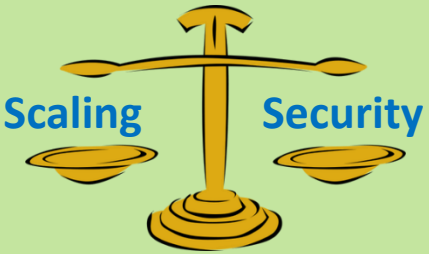
$\frac{S}{N}$ | CMOS compared to Dual rail at 28nm

Implementation

Evaluation

AES 8-bit S-box

- CMOS styles implemented in 65nm and 28nm
- DDSLL styled using BDD implemented in 65nm and 28nm



Low power operation

- All S-boxes operated from nominal voltage of technology to (nominal voltage – 500 mV)

Contributions

$\frac{S}{N}$ | CMOS compared to Dual rail at 65nm

$\frac{S}{N}$ | CMOS compared to Dual rail at 28nm

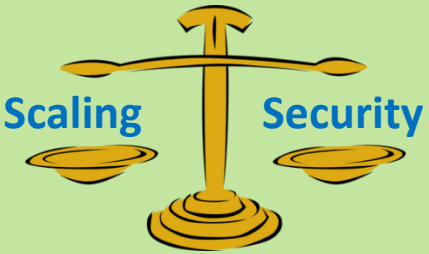
Implementation

AES 8-bit S-box

- CMOS styles implemented in 65nm and 28nm
- DDSLL styled using BDD implemented in 65nm and 28nm

Low power operation

- All S-boxes operated from nominal voltage of technology to (nominal voltage – 500 mV)



Evaluation

Performance Metric

- Energy-per-operation

Security Metric

- SNR

Distinguisher

- Points-of-Interest
- (PCA and Maximum signal)

Noise level

- Noiseless

Contributions

- Cautionary note: no routing parasitics or PVT variations are considered, so this is the ideal case; when included it should only add to the performance degradation in dual rail circuits.

Performance Evaluation methodology

- Performance Metrics

Performance Evaluation methodology

- Performance Metrics
  - We have 2 different logic styles, CMOS & DDSLL

Performance Evaluation methodology

- Performance Metrics
  - We have 2 different logic styles, CMOS & DDSLL
  - Both are being scaled down from 65nm to 28nm

Performance Evaluation methodology

- Performance Metrics
  - We have 2 different logic styles, CMOS & DDSLL
  - Both are being scaled down from 65nm to 28nm
  - Voltages are also scaled from nominal to nominal-0.5V

Performance Evaluation methodology

- Performance Metrics
  - We have 2 different logic styles, CMOS & DDSLL
  - Both are being scaled down from 65nm to 28nm
  - Voltages are also scaled from nominal to nominal-0.5V
  - Energy per operation, a relatively discriminant metric, is used for performance comparison since it integrates the total power over the time.

$$E_{op} = \int_t (P_{dyn} + P_{stat})\, dt,$$

$$= \underbrace{\frac{1}{2}\, N_{sw}\, C_L\, V_{DD}\, V_{swing}}_{\text{Dynamic}} + \underbrace{V_{DD}\, I_{leak}\, T_{del}}_{\text{static}},$$

Security evaluation methodology

- Security Metrics
  - Simulated traces without any physical noise

$$L_t^i(X, N) = L_t^{simu}(X, N)$$

Security evaluation methodology

- Security Metrics
  - Simulated traces without any physical noise

$$L_t^i(X, N) = L_t^{simu}(X, N)$$

  - A multivariate power trace, $\boldsymbol{l}$, reduced to univariate using PCA

$$l = PCA(\boldsymbol{l})$$

Security evaluation methodology

- Security Metrics
  - Simulated traces without any physical noise

$$L_t^i(X,N) = L_t^{simu}(X,N)$$

  - A multivariate power trace, $\boldsymbol{l}$, reduced to univariate using PCA

$$l = PCA(\boldsymbol{l})$$

  - Using Mangard's SNR,

$$\mathrm{SNR} = \frac{\hat{\mathrm{var}}_x\left(\hat{\mathrm{E}}_i\left(L_x^i\right)\right)}{\hat{\mathrm{E}}_x\left(\hat{\mathrm{var}}_i\left(L_x^i\right)\right)},$$

Security evaluation methodology

- Security Metrics
  - Simulated traces without any physical noise

$$L_t^i(X,N) = L_t^{simu}(X,N)$$

  - A multivariate power trace, $\boldsymbol{l}$, reduced to univariate using PCA

$$l = PCA(\boldsymbol{l})$$

  - Using Mangard's SNR,

$$\mathrm{SNR} = \frac{\hat{\mathrm{var}}_x\left(\hat{\mathrm{E}}_i\left(L_x^i\right)\right)}{\hat{\mathrm{E}}_x\left(\hat{\mathrm{var}}_i\left(L_x^i\right)\right)},$$

Since we have *noise-free* traces, we compute only the numerator corresponding to ***maximizing*** the signal,

$$\hat{\mathrm{var}}_x\left(\hat{\mathrm{E}}_i\left(L_x^i\right)\right)$$

Simulation Parameters

- Simulation settings:

Simulation Parameters

- Simulation settings:
  - Standard V_t transistors (no forward or reverse biasing effects)

Simulation Parameters

- Simulation settings:
  - Standard V_t transistors (no forward or reverse biasing effects)
  - From 1.2V nominal for 65nm LP and 1V nominal for 28nm FDSOI, simulated up to nominal-500 mV

Simulation Parameters

- Simulation settings:
  - Standard V_t transistors (no forward or reverse biasing effects)
  - From 1.2V nominal for 65nm LP and 1V nominal for 28nm FDSOI, simulated up to nominal-500 mV
  - 10MHz frequency of operation

Simulation Parameters

- Simulation settings:
  - Standard V_t transistors (no forward or reverse biasing effects)
  - From 1.2V nominal for 65nm LP and 1V nominal for 28nm FDSOI, simulated up to nominal-500 mV
  - 10MHz frequency of operation
  - Input signal which transitions from 0-1,0-2,….0-N where N=255 for security analysis

Simulation Parameters

- Simulation settings:
  - Standard V_t transistors (no forward or reverse biasing effects)
  - From 1.2V nominal for 65nm LP and 1V nominal for 28nm FDSOI, simulated up to nominal-500 mV
  - 10MHz frequency of operation
  - Input signal which transitions from 0-1,0-2,….0-N where N=255 for security analysis
  - Random 1000-bit input signal for E_{op} calculation

Simulation Parameters

- Simulation settings:
  - Standard V_t transistors (no forward or reverse biasing effects)
  - From 1.2V nominal for 65nm LP and 1V nominal for 28nm FDSOI, simulated up to nominal-500 mV
  - 10MHz frequency of operation
  - Input signal which transitions from 0-1,0-2,….0-N where N=255 for security analysis
  - Random 1000-bit input signal for E_{op} calculation
  - signal, S_{PCA} and energy per operation, E_{op} w.r.t V_{DD} computed

Security analysis

Security vs Performance– PCA applied signal

- The x-axes represents the Energy per operation **ratio** between CMOS & DDSLL, i.e

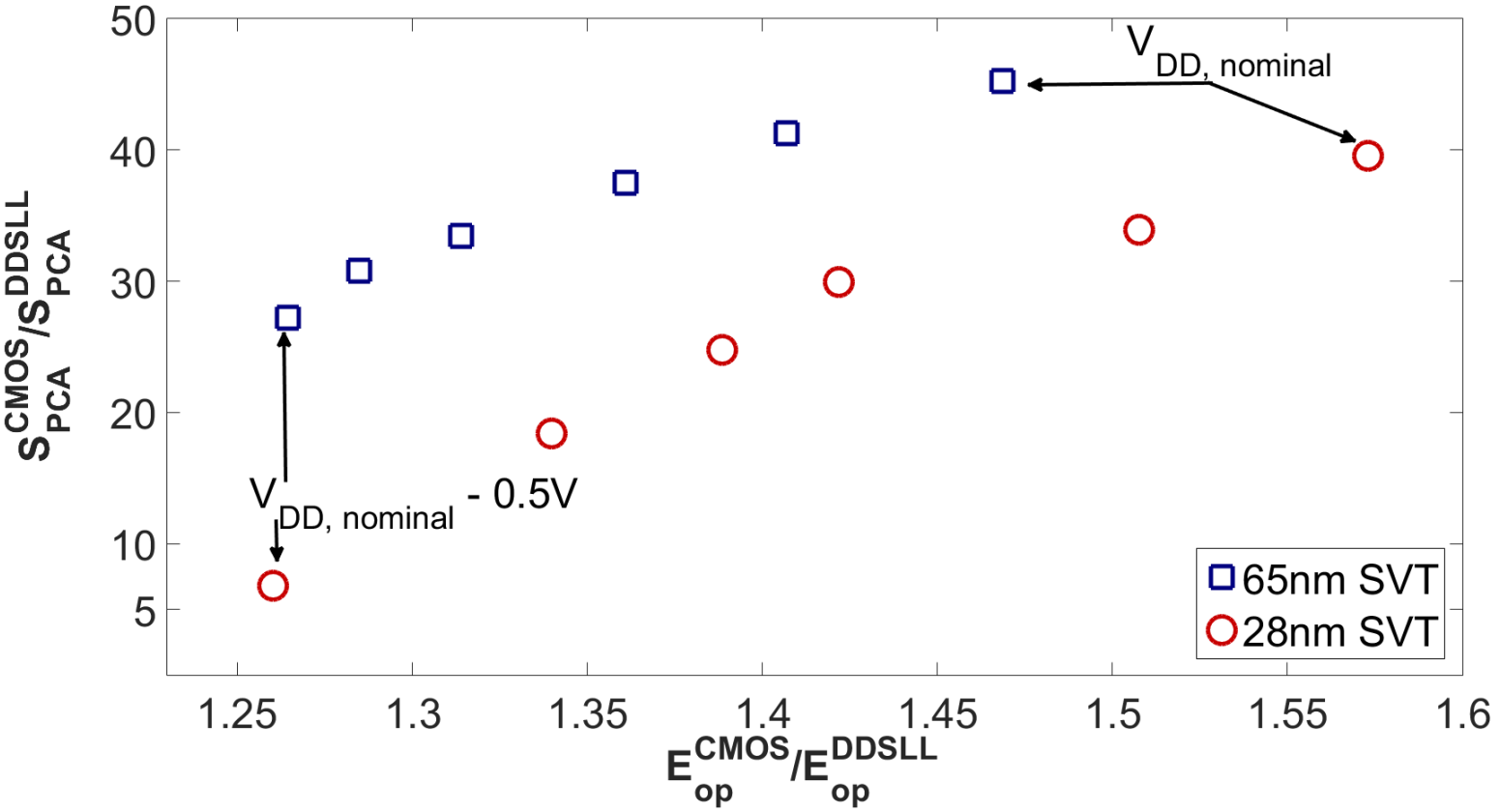
$$\frac{E_{op}{}^{CMOS}}{E_{op}{}^{DDSLL}}$$

- The y-axes represent the **ratio** of the PCA-applied Signal between CMOS & DDSLL, i.e

$$\frac{S^{PCA}{}_{CMOS}}{S^{PCA}{}_{DDSLL}}$$

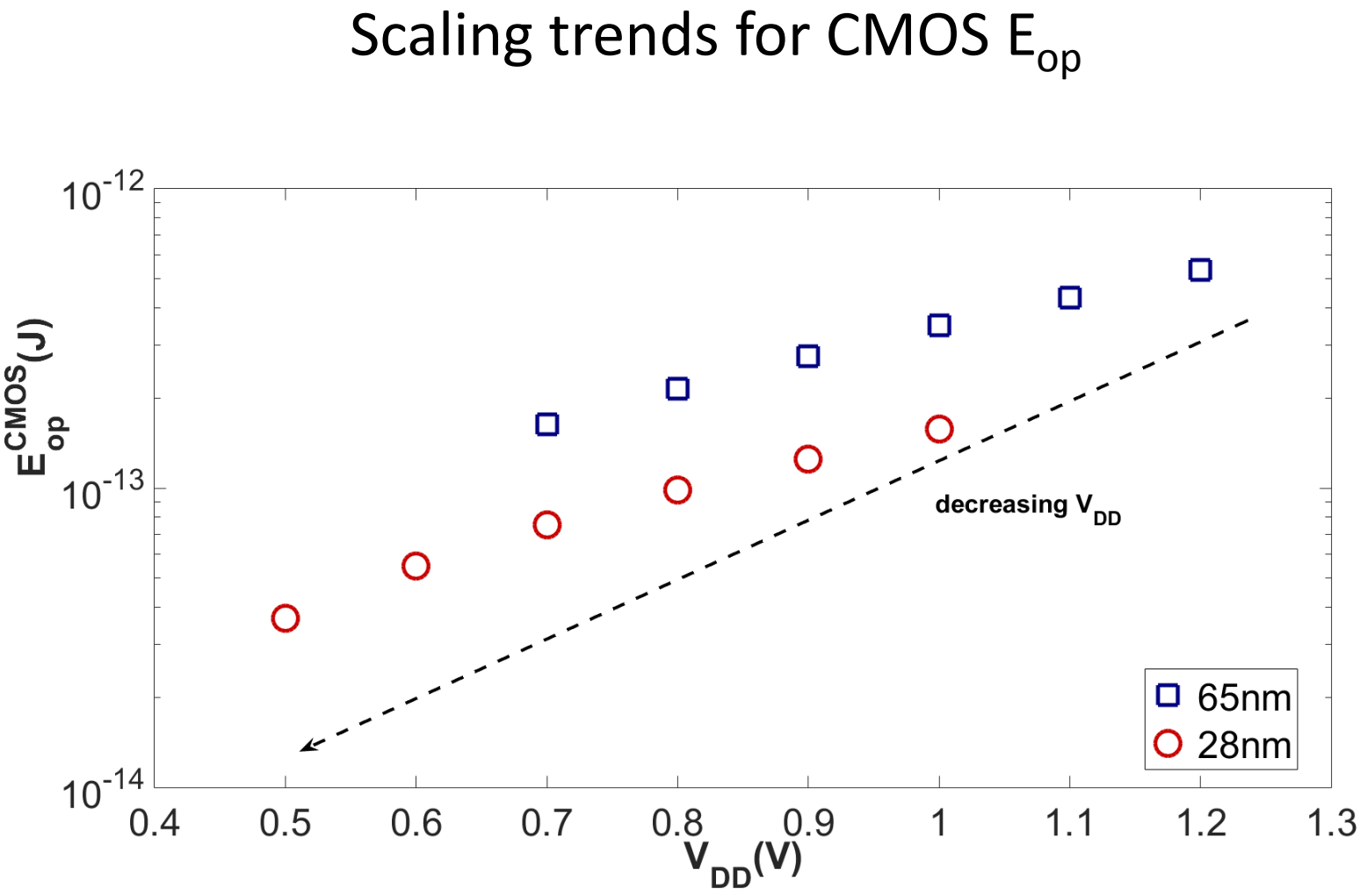
Scaling Trends-CMOS/DDSLL- 65nm/28nm

Security vs Performance– PCA applied signal

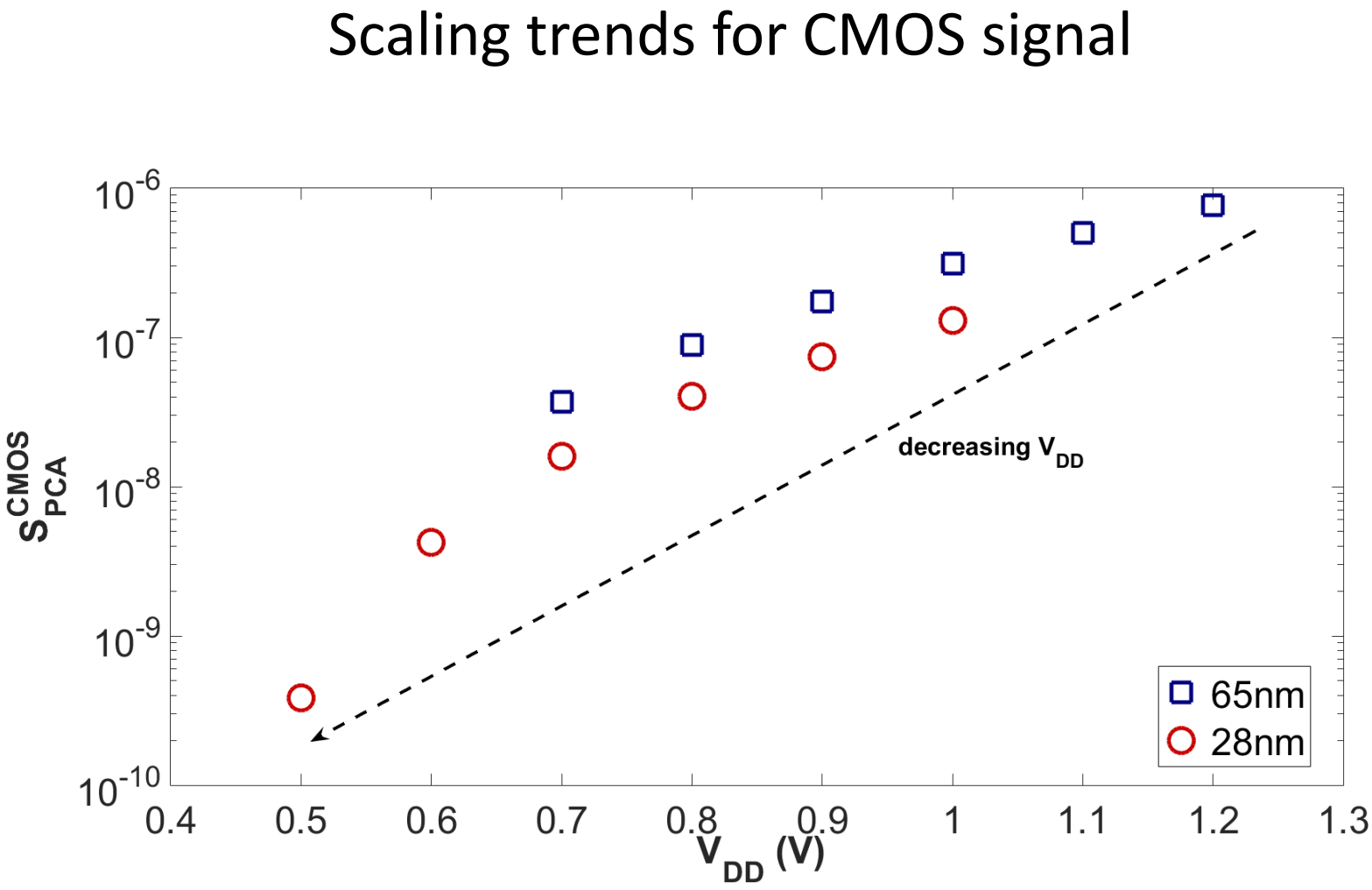


- Nominal voltage for 65nm technology is 1.2V (1.2-0.7V)
- Nominal voltage for 28nm technology is 1V (1 – 0.5V)

Scaling trends for CMOS - Performance



Scaling trends for CMOS - Security



Conclusion

- Interest in DDSLL over CMOS vanishes as circuit sizes shrink.
  - Because the impact of imbalanced capacitances increases at lower-scaled technologies.
  - We believe this trend hold true for other types of dual-rail logic styles (like WDDL, DyCML, SABL)

Conclusion

- Interest in DDSLL over CMOS vanishes as circuit sizes shrink.
  - Because the impact of imbalanced capacitances increases at lower-scaled technologies.
  - We believe this trend hold true for other types of dual-rail logic styles (like WDDL, DyCML, SABL)

- SNR reduction via signal reduction is likely to become increasingly challenging.
  - By contrast, scaling trends are positive for CMOS because of increase in (intrinsic) noise.

Conclusion

- Interest in DDSLL over CMOS vanishes as circuit sizes shrink.
  - Because the impact of imbalanced capacitances increases at lower-scaled technologies.
  - We believe this trend hold true for other types of dual-rail logic styles (like WDDL, DyCML, SABL)

- SNR reduction via signal reduction is likely to become increasingly challenging.
  - By contrast, scaling trends are positive for CMOS because of increase in (intrinsic) noise.

- Designing efficient and noisy CMOS implementations is an interesting research challenge
  - Dual-rail logics may still be useful for other purposes such as ensuring independence for masking

Thank you