CZECH
TECHNICAL
UNIVERSITY
IN PRAGUE

Faculty of Information Technology

The Design-Time Side-Channel Information Leakage Estimation

Jan Bělohoubek

jan.belohoubek@fit.cvut.cz

Czech Technical University in Prague
Prague, Czech Republic

CryptArchi 2017, Smolenice – Slovakia

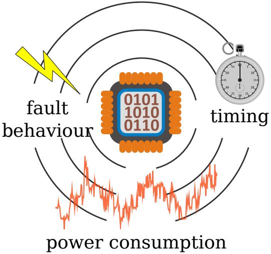
Motivation

Information Leakage

- Digital circuits offer sensitive information while computation(side-channel)
- Today circuit designers compete with attackers:
  - Designers are trying to build circuits resistant to SPA, DPA, Fault-attack, Combined (Fault + PA) . . .
  - → **Decrease** the information offered thru side-channel
  - → **Measure** the information offered thru side-channel

Motivation

Leakage Sources



- **Unbalanced data/control paths** (Different loads, Place&Route, Early evaluation)
- **Unbalanced computation (data-dependent algorithms)**
- Completion detection – asynchronous circuits

Motivation

Localize Weakness and Estimate Potential

- **How to distinguish good idea[1] and bad idea during the different design phases?**
    - post-Synthesis – what can be achieved with current design?
    - post-Map – what can be achieved with current cell library?
    - post-Place&Route – how will behave the physical design?

[1] Is a certain circuit implementation better from the side-channel vulnerability point of view?

Motivation

How To Measure Vulnerability?

- **Production time**
  - Number of traces needed to break the circuit (get AES key)
- **Design time**
  - Use number of traces [2] – accurate simulation + many traces → time !?
  - Use well established methods – make conservative (but subjective) estimation → accuracy !?
  - **Do we have objective and efficient metric?**

[2] K. Smith and M. Łukowiak, "Methodology for simulated power analysis attacks on AES," 2010 - MILCOM 2010 MILITARY COMMUNICATIONS CONFERENCE, San Jose, CA, 2010, pp. 1292-1297.

The Method
Using Power Traces

- **The sensitive information leaking from the circuit influences the character of the power traces**
  - Timing - differential peak position; duration of the computation
  - Fault - differential peak position, width or height; duration of the computation
  - Unbalanced paths - differential peak position, width or height

→ Many types of information leakage are aggregated in power traces

→ **Using only power traces for vulnerability evaluation is sufficient**

The Method

- **What is Required?**
  - Fast vulnerability estimation allowing incorporation into the design flow process
  - Measure the information contained in power trace
  - Estimation at different design levels – post-Synthesis, post-Map, post-Place&Route
- **What is Observed?**
  - The information in the power trace is proportional to the similarity of traces
    - → If all traces would be equal, the attacker can extract no information
    - → If there is a dependency between the processed data and power trace patterns, the attacker may extract information

The Method

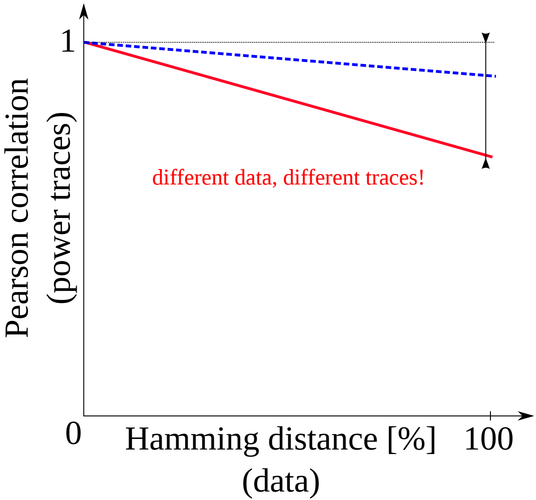
Data vs. Power Trace Dependency

- **Let’s search for data vs. power trace dependency**
  - Data similarity metric: **Hamming distance**
  - Power trace similarity metric: **Pearson correlation**

  → **Is correlation of traces for similar data high and for different data (significantly) low?**

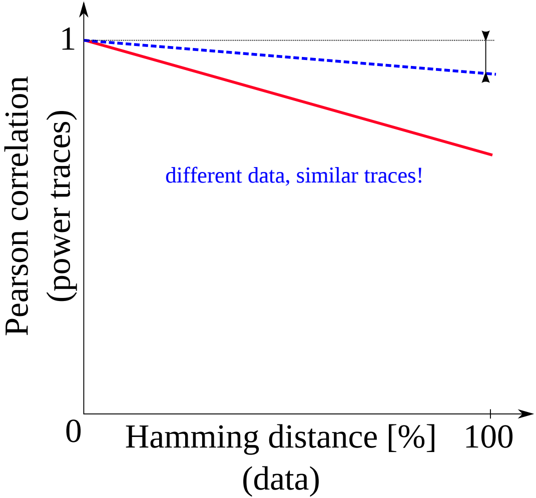
The Method

Data vs. Power Trace Dependency



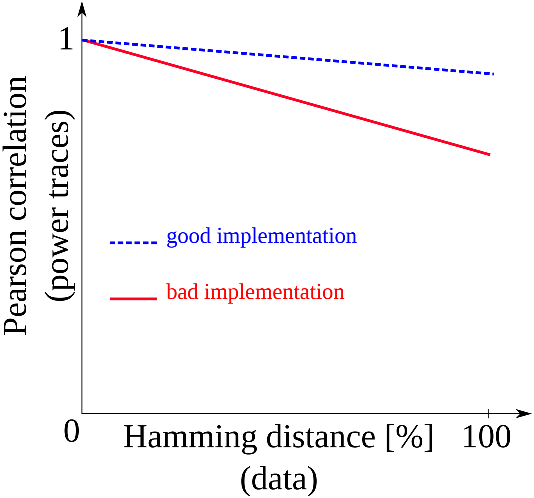
The Method

Data vs. Power Trace Dependency



The Method

Data vs. Power Trace Dependency

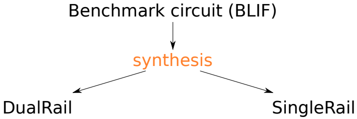


Methodology
The Current Design Potential

- post-Synthesis – what can be achieved with current design?
  - No physical layer information!
  - Is simulation-based estimation possible? **It is not possible without any assumption about technology!**
- post-Map – what can be achieved with current cells?
  - Take information about cells only (parasitic capacitances, conductivity, . . .)
  - Interconnection is assumed ideally balanced (or zero delay/power)
  - Place&Route can make things worse
- post-Place&Route – the "reality"
  - Should be close to physical design

Methodology

Design Flow



Benchmark circuit (BLIF)

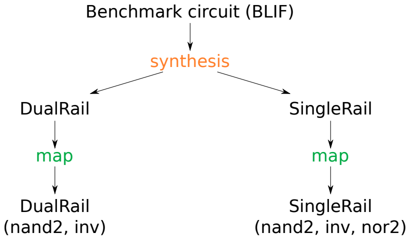
synthesis

DualRail

SingleRail

Methodology

Design Flow



Benchmark circuit (BLIF)

synthesis

DualRail

SingleRail

map

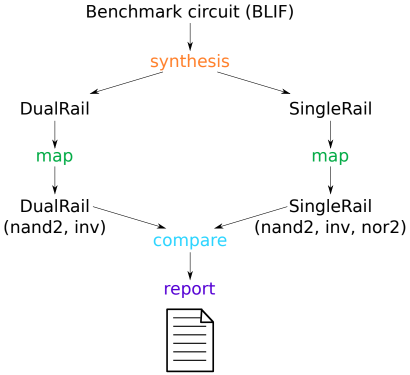
map

DualRail
(nand2, inv)

SingleRail
(nand2, inv, nor2)

Methodology

Design Flow



Benchmark circuit (BLIF)

synthesis

DualRail

SingleRail

map

map

DualRail (nand2, inv)

SingleRail (nand2, inv, nor2)

compare

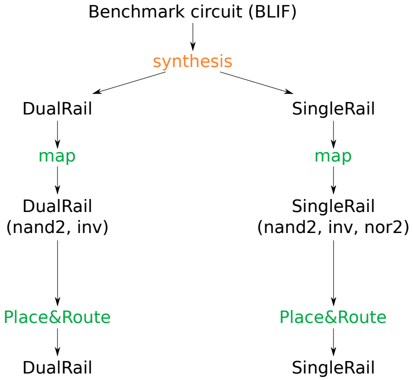
report

Methodology
Report

- post-Map – which implementation is better (with current cells)?

Methodology

Design Flow



Benchmark circuit (BLIF)

synthesis

DualRail

SingleRail

map

map

DualRail
(nand2, inv)

SingleRail
(nand2, inv, nor2)

Place&Route

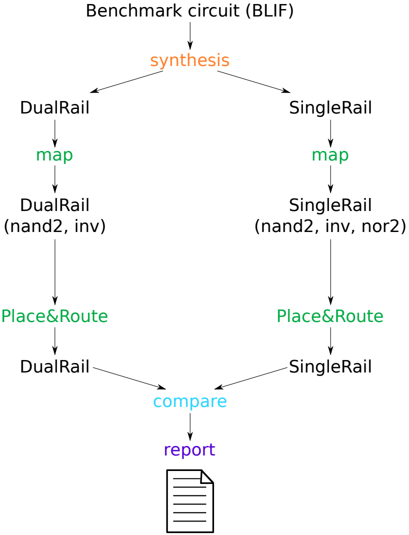
Place&Route

DualRail

SingleRail

Methodology

Design Flow



Benchmark circuit (BLIF)

synthesis

DualRail

SingleRail

map

map

DualRail
(nand2, inv)

SingleRail
(nand2, inv, nor2)

Place&Route

Place&Route

DualRail

SingleRail

compare

report

Methodology

Report

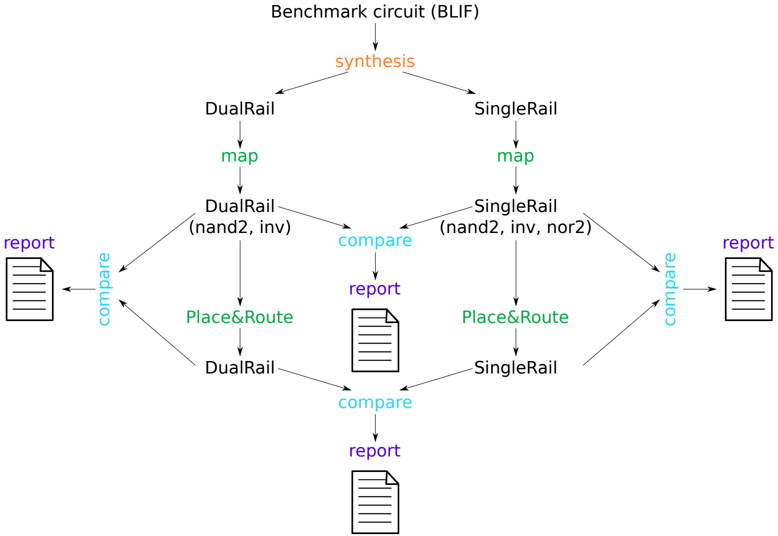
- post-Place&Route – how bad/good is the result after Place&Route?

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Faculty of Information Technology

Methodology

Design Flow



Methodology

Report

- post-Map → post-Place&Route – how bad/good is Place&Route itself?

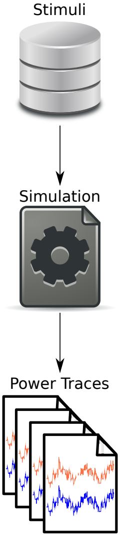
Methodology

Combinational Circuits

- Generate the stimuli set:
  - Initial vector is generated randomly
  - Other vectors are derived by inverting bits in the initial vector
  - → The stimuli set contains vectors with Hamming distances (0% – 100%)
- Use stimuli to get power traces (simulation)

Methodology

Combinational Circuits



Stimuli

Simulation

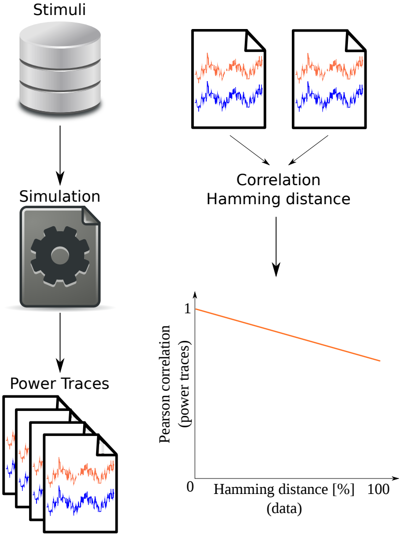
Power Traces

Methodology
Combinational Circuits

- Generate the stimuli set:
  - Initial vector is generated randomly
  - Other vectors are derived by inverting bits in the initial vector
  - → The stimuli set contains vectors with Hamming distances (0% – 100%)
- Use stimuli to get power traces (simulation)
- Compute Pearson correlation for all pairs of power traces
- Build a data-set containing pairs: [Hamming distance, Correlation] (plot . . .)

Methodology

Combinational Circuits



Stimuli

Simulation

Power Traces

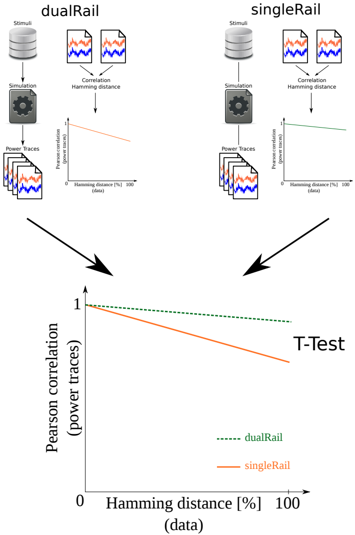
Correlation
Hamming distance

Methodology
Combinational Circuits

- Generate the stimuli set:
  - Initial vector is generated randomly
  - Other vectors are derived by inverting bits in the initial vector
  - → The stimuli set contains vectors with Hamming distances (0% – 100 %)
- Use stimuli to get power traces (simulation)
- Compute Pearson correlation for all pairs of power traces
- Build a data-set containing pairs: [Hamming distance, Correlation] (plot . . .)
- Compare different implementations: formulate hypothesis and test by using the t-test

Methodology

Combinational Circuits



Simulation

Tools

- Spice – open (ngSpice); too accurate; **too slow**
- Synopsys PrimeTime PX – commercial – looks fine (not tested yet)
- IRSIM – open alternative to PTPX?; fast; **too old**
  - Produces event times, not power traces (poweEst package is available)
  - Good for CMOS with lambda $\geq 1\ \mu m$ technology
  - For CMOS below 1 μm, the results looks bad – characterization failed ...

Simulation
Combinational circuits

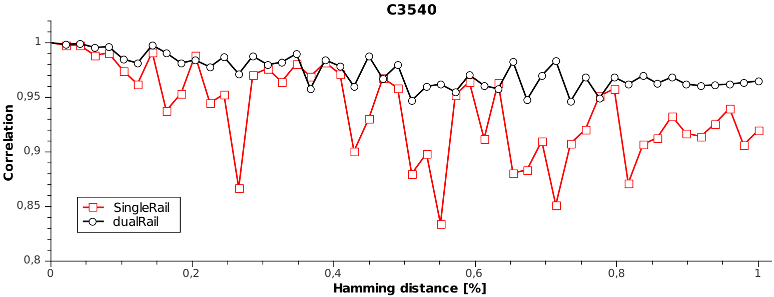
- Stimuli set contains i vectors, where i is equal to # of circuit inputs

→ We have $i^2/2$ pairs of vectors with all possible Hamming distances

  - The number of stimuli vectors is reduced
  - SPICE simulation is feasible for relatively small circuits like C3540:
    - $\approx$ 1000 gates
    - 50 inputs
    - 1250 input vector and power trace pairs

Simulation – IRSIM

Above 1 μm



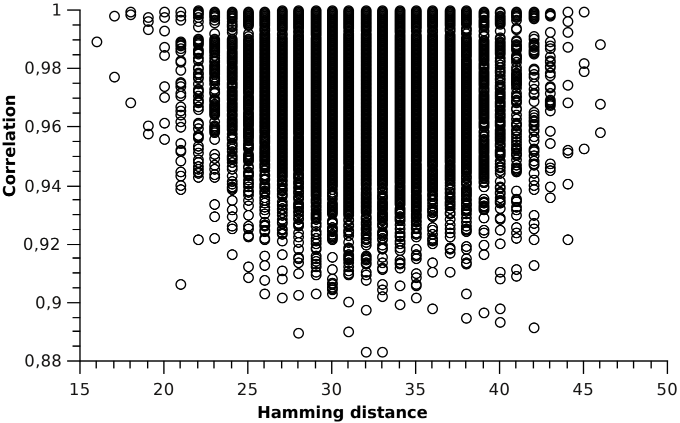
- Nice graph, looks as expected – T-test (and my eyes) says: **singleRail is** (much) **worse than dualRail**
- IRSIM gives similar results for TSMC180nm – here **disagrees with SPICE!** (wrong tech. characterization)

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Faculty of Information Technology

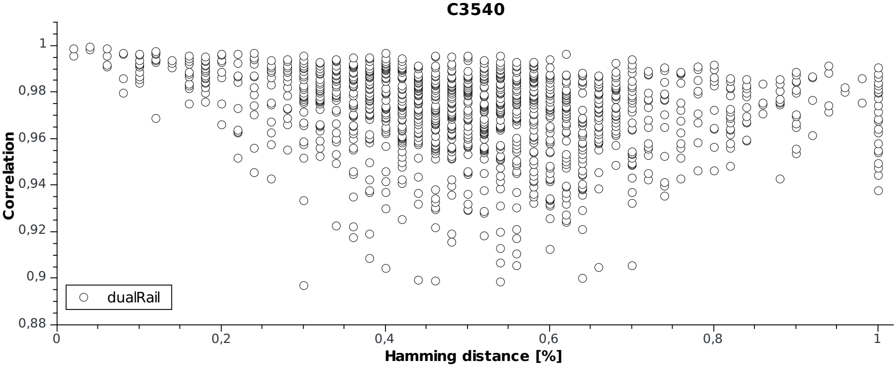
Measurements

3 Years Ago . . . CryptArchi 2014



- Real measurements – Asynchronous dualRail DES on FPGA

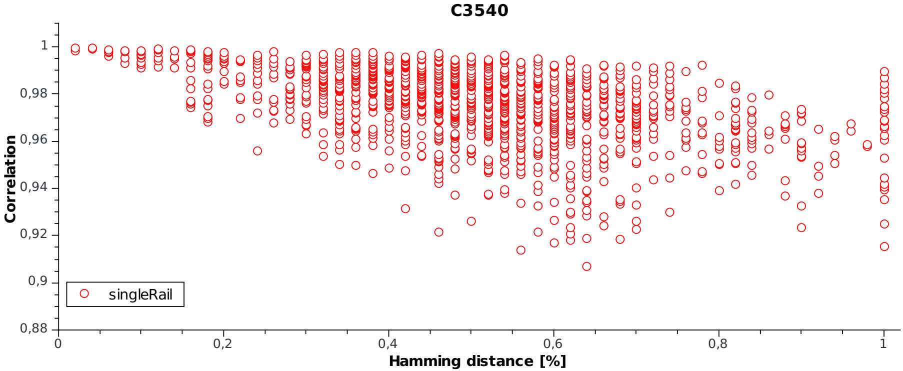
Simulation – SPICE
DualRail



- DualRail layout (TSMC180nm) of the benchmark circuit C3540
- Precise SPICE simulation looks very similar to measured data! (C3540 is similar to DES)

Simulation – SPICE

SingleRail



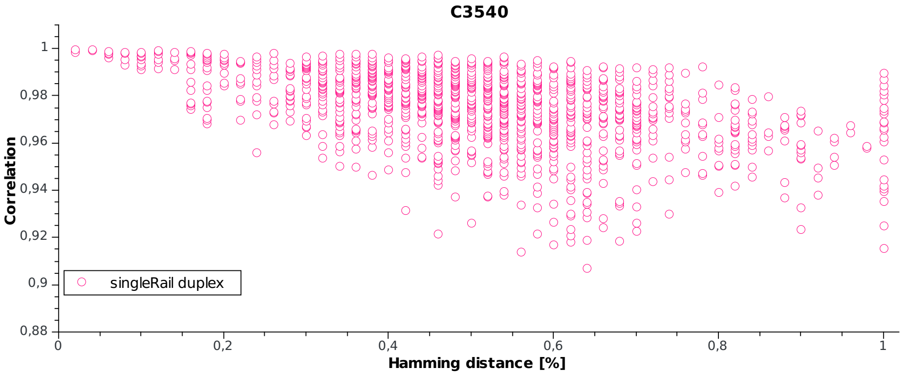
- SingleRail has less variation and the minimum of singleRail is above dualRail
- T-Test (not my eyes here!) says: **singleRail is better!** (a bit)

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Faculty of Information Technology

Simulation – SPICE

Duplex of SingleRails (no voter)



- The sum of two singleRails is equal to the single SingleRail – **no additional information leakage!**

Summary

Preliminary Results Show Interesting Facts

When no manufacturing variations were taken into account:

1. More logic working data-dependently is bad → information leakage is increased
   - both branches of DualRail circuits perform data-dependent computations → **balancing becomes extremely important!**
2. Adding more logic blocks producing exactly the same power traces is OK → NMR will not increase information leakage

When manufacturing variations will be taken into account, the 2. case will slightly become case 1!

Summary

Future Work and Challanges

- Is it possible to measure information leakage simpler?

→ the area of circuit parts performing data-dependent computations independently

- Is singleRail really better than dualRail in practice?
  . . . No!

→ Where are the limits of masking (balancing dual rails)?

→ What is the relationship of information leakage and circuit vulnerability?

→ Is the attacker's strength estimation – without focusing to the particular attack – possible?

- There is no (open) efficient and accurate simulator of CMOS producing power traces.

Thank you!

- The information leakage is proportional to the amount of logic working data-dependently!
- The presented method is able to estimate information leakage (fast open simulator is missing).
- Ideal duplex (no voters!) does not offer additional information to attacker.

This research has been partially supported by the grant GA16-05179S of the Czech Grant Agency and by CTU grant SGS17/213/OHK3/3T/18.
Computational resources were provided by the CESNET LM2015042 and the CERIT Scientific Cloud LM2015085, provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures".