A Secure Asynchronous Configurable Cell

An Embedded Programmable Logic for Smartcards

Laurent Fesquet,

Taha Beyrouthy

TIMA

Laurent.Fesquet@imag.fr

Taha.beyrouthy@imag.fr

Summary

- About FPGA security
- Asynchronous logic principles
- A secure asynchronous FPGA
- Conclusion

Summary

- About FPGA security
- Asynchronous logic principles
- A secure asynchronous FPGA
- Conclusion

Why secure FPGA?

- Flexible (security and application updates)
- Many possible attacks :
  - Invasive
  - Bit stream Reverse
  - Side Channel Attacks :
    - Timing Attack: TA
    - Consumption : SPA, DPA, CPA, Template
    - Electromagnetic Field : EMA
  - Attack by Fault Injection : FA

Side Channel Attacks easier in FPGAs

- Larger area :
  - Cells + Switches + Tracks + Configuration
  - Mean ratio ~35 time larger than ASICs (Kuon 2006)
- Higher Power Consumption
  - Mean ratio ~12 time higher than ASICs (Kuon 2006)
  - Consumption in Vdd^2 and Vdd^3 (Garcia 2000)
- Many Highly-buffered D Flip-Flops
  - 1 cell=1LUT+1DFF

Fault Attacks in FPGAs

- The most powerful attack
- But difficult to perform
  - Laser
  - Strong glitches
  - Setup violation
  - …

Side-Channel Countermeasures

- DPA/EMA
  - Randomize the Power consumption
    - **Masking**
  - Hide the Power Consumption
    - **Dual-Rail**
- FA
  - Redundancy, coding
  - Fault detection

Securing FPGA

- Intrinsic: FPGA internal
  - Cell : Architecture and Layout balancing
  - Interconnection : Topology and Layout balancing
- Extrinsic: FPGA application
  - Architecture
  - Logic synthesis
  - P/R
  - Dynamic reconfiguration

SAFE Project

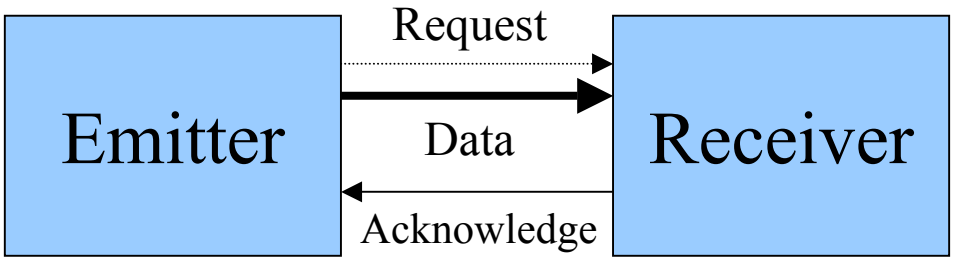
Secure Asynchronous FPGA for Embedded systems.

- Design of an Asynchronous FPGA which
  - supports multi-asynchronous style of logic (4-phase and 2-phase),
  - supports many asynchronous data encoding (2 rails, 3 rails, 4 rails, …)
  - manages security issues at architectural and electrical levels

Summary

- About FPGA security
- Asynchronous logic principles
- A secure asynchronous FPGA
- Conclusion

Asynchronous circuit design principles



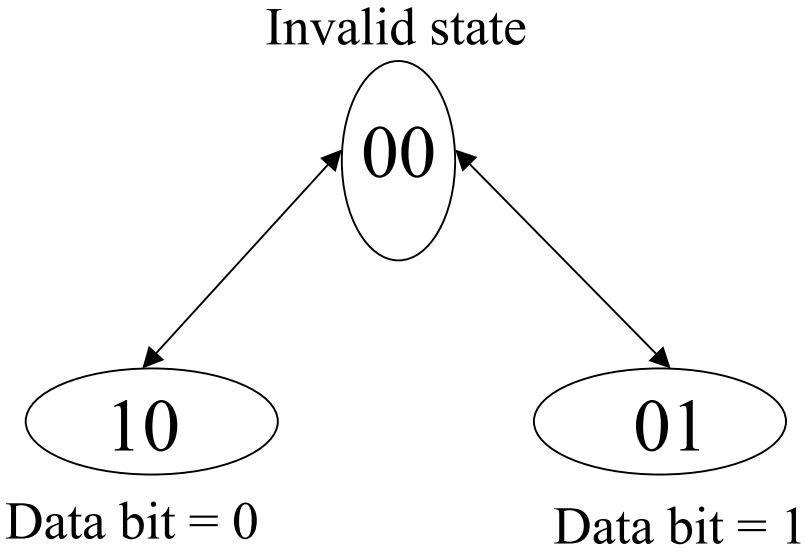
- Unified asynchronous interface
- Control and data are encoded (together or not)
- Two basic rules :
  - the emitter issues a request when a data is valid
  - the receiver issues an acknowledge when the data is processed

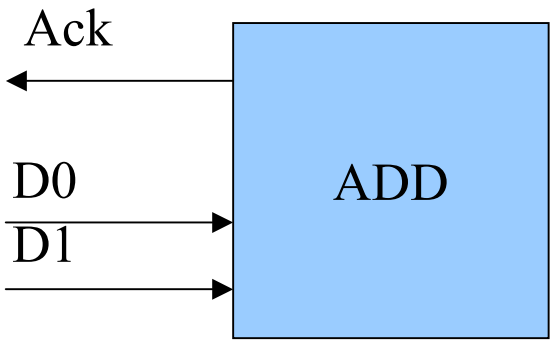
→ **Several hardware implementation of the handshake protocol**

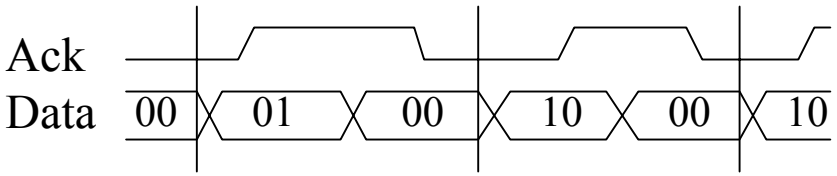
→ **Delay insensitivity**

Asynchronous circuit design principles

Data encoding : Three states (dual rail)

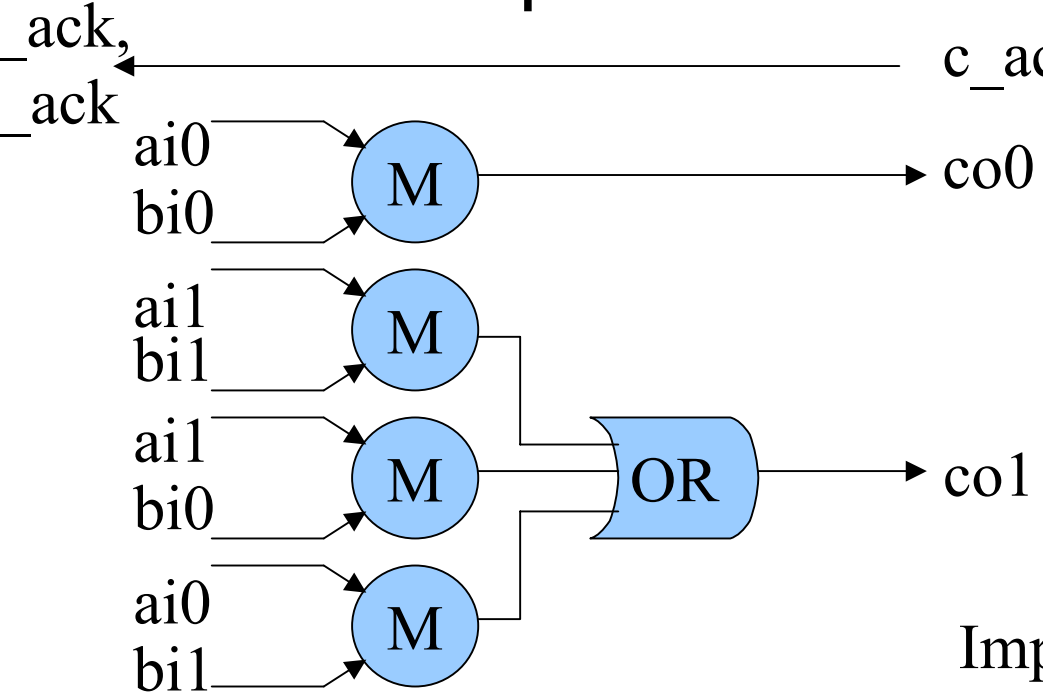


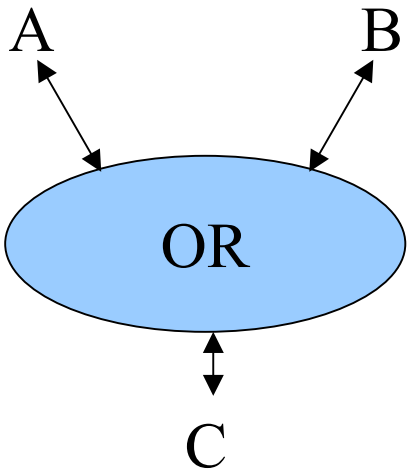




QDI asynchronous circuits

- An example : dual-rail OR Gate





Implement both the function and the protocol

About asynchronous logic…

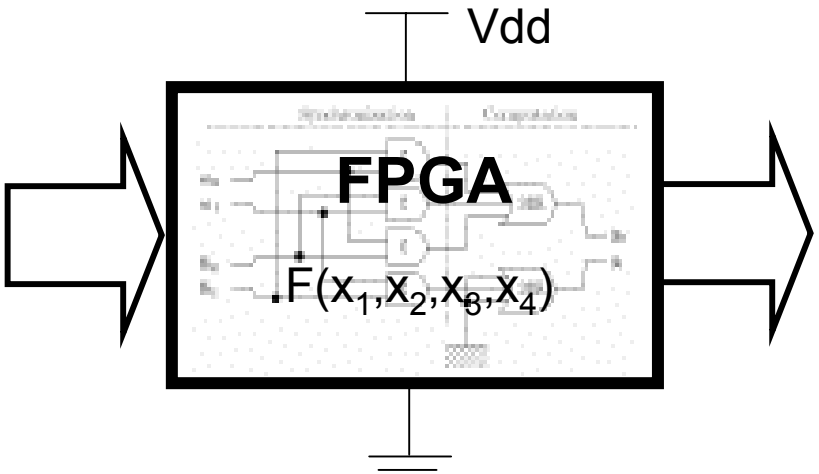
- … for securing circuits
  - Dual-rail (QDI)
  - Multi-rail (QDI)
  - Timing attack difficult to perform
  - No glitches
  - Smooth and low power consumption
  - Difficult to synchronize

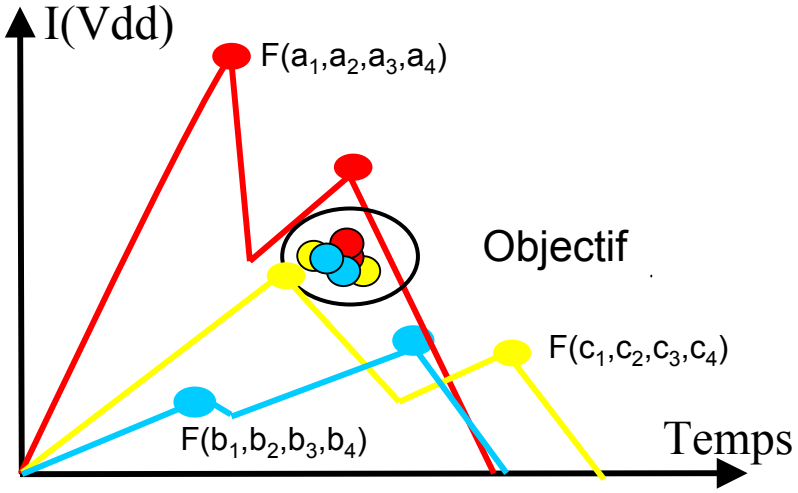
Summary

- About FPGA security
- Asynchronous logic principles
- A secure asynchronous FPGA
- Conclusion

A secure asynchronous configurable cell

- Security constraint to be robust against Side channel attacks :
  - Data independent power consumption

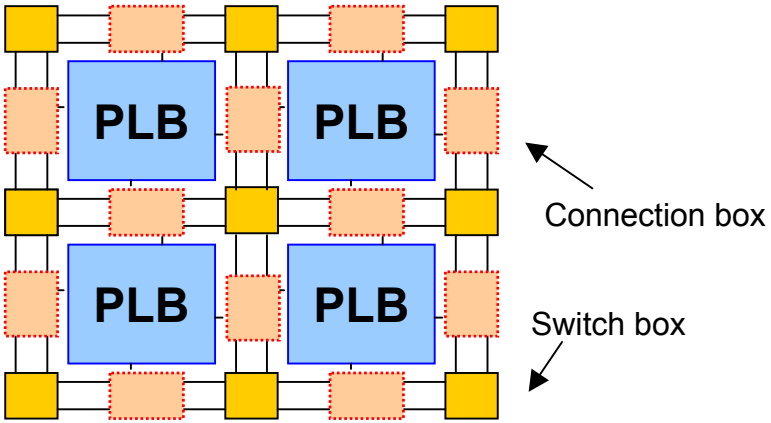




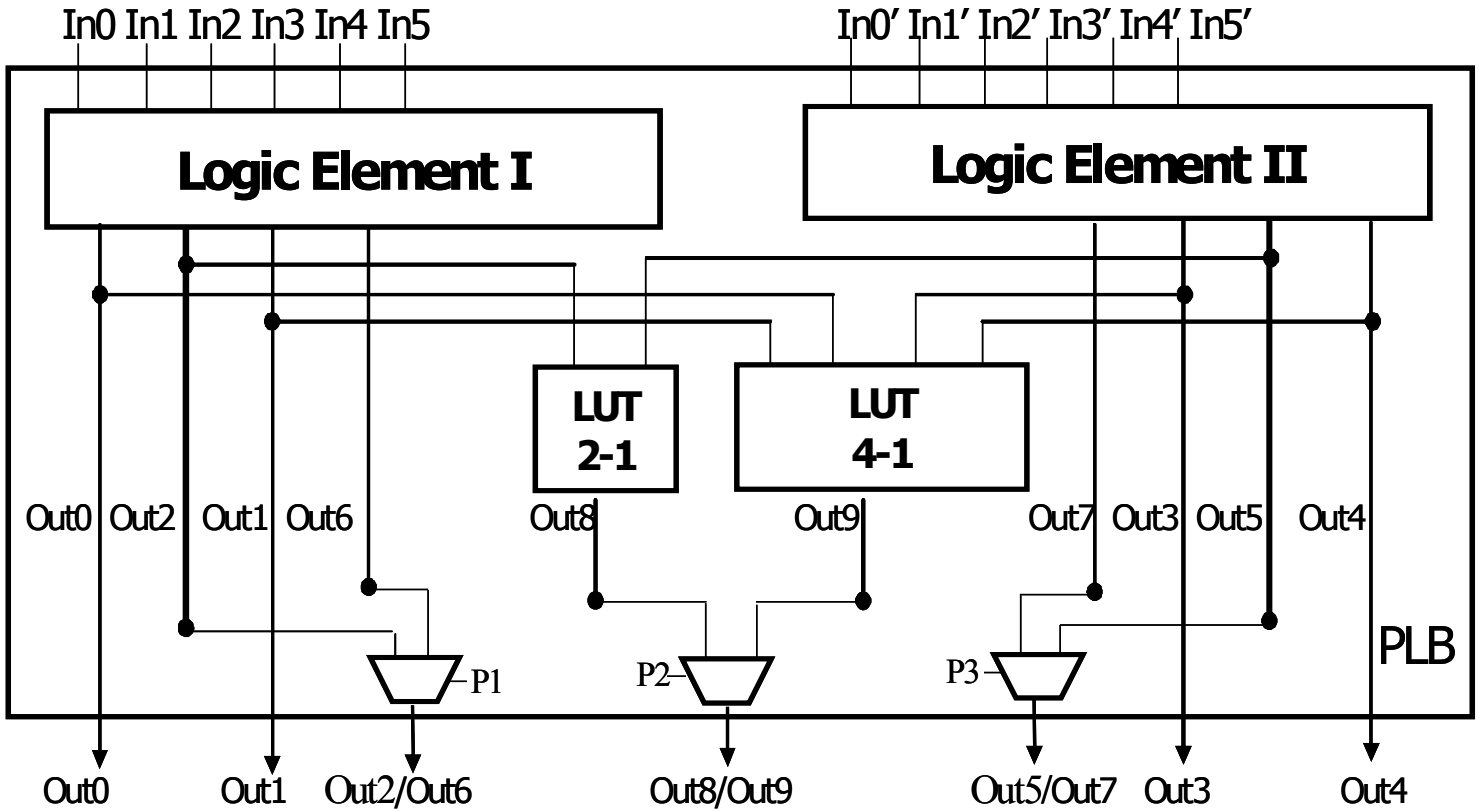
FPGA architecture overview

- Island style, designed to be :
  - Electrically balanced (Balanced input capacitances).
  - Logically balanced (Same logical depth for all inputs).

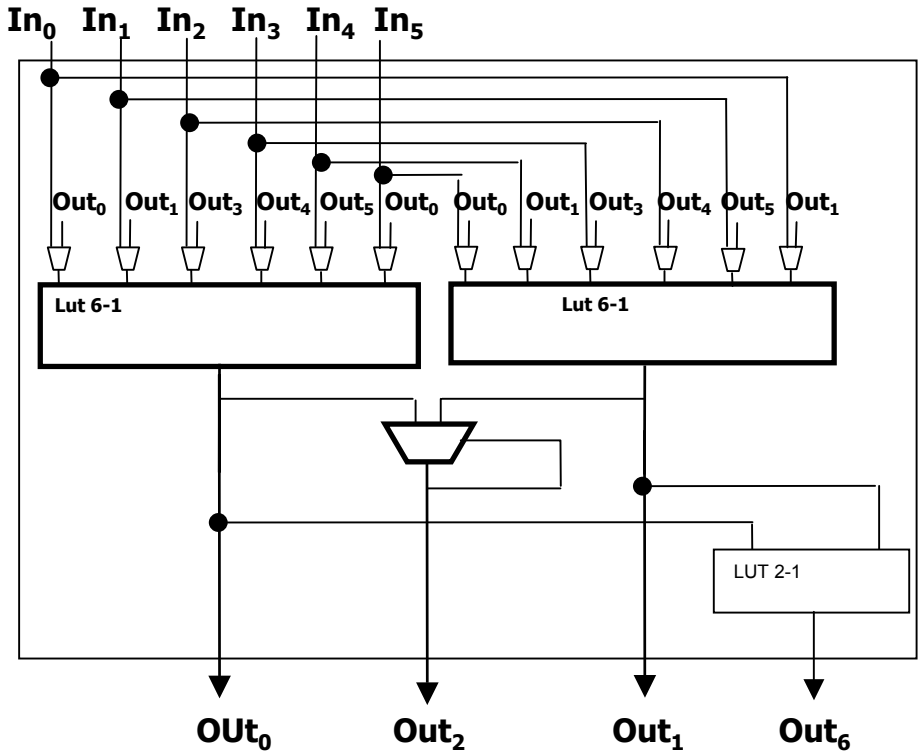
  + Constant Hamming weight using 1 of n encoding



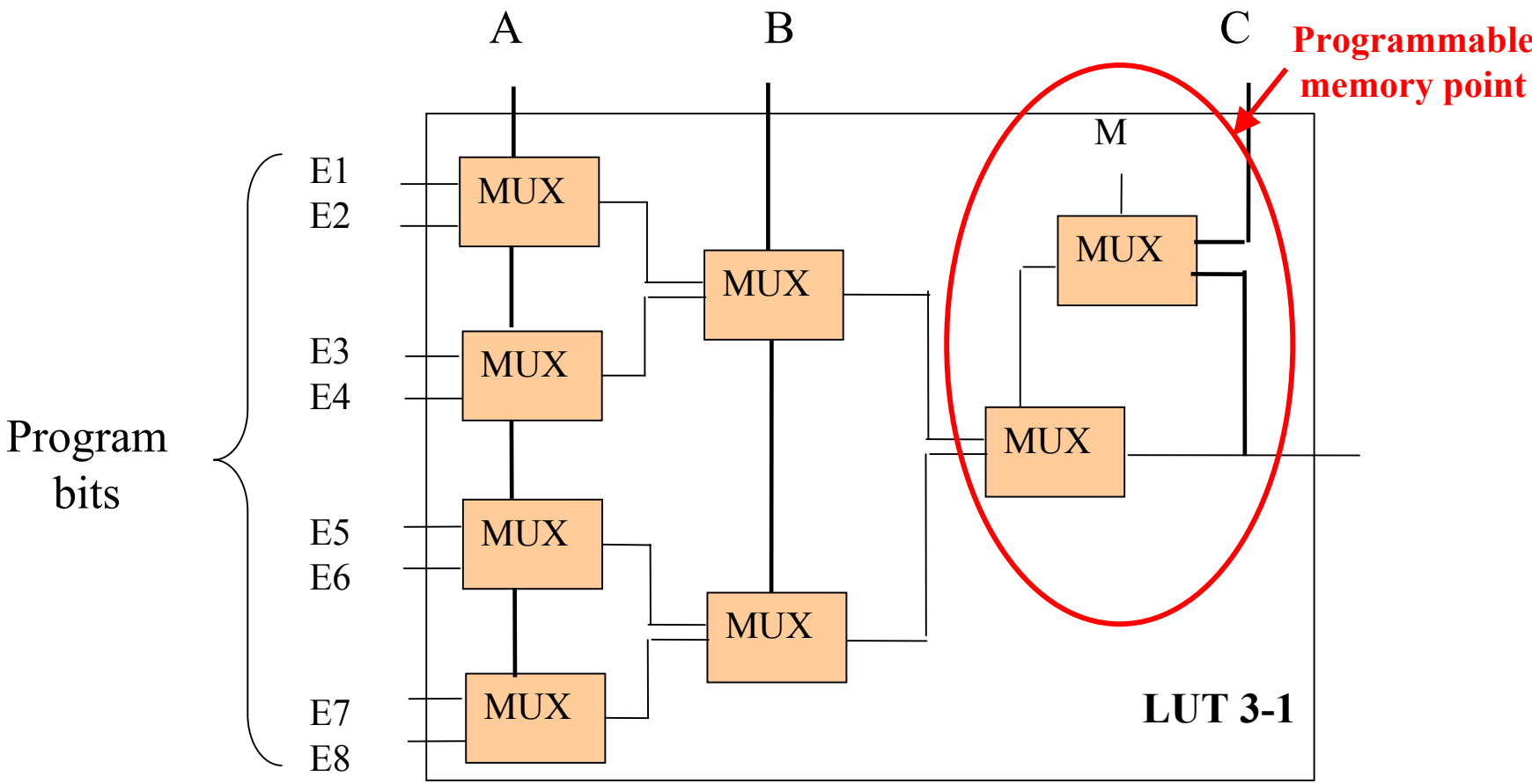
Programmable Logic Bloc PLB



Logic Element (LE)

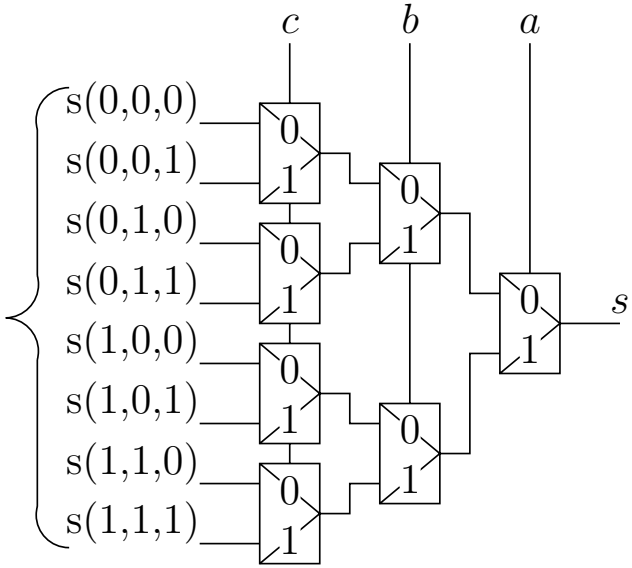


State-holding in LUT

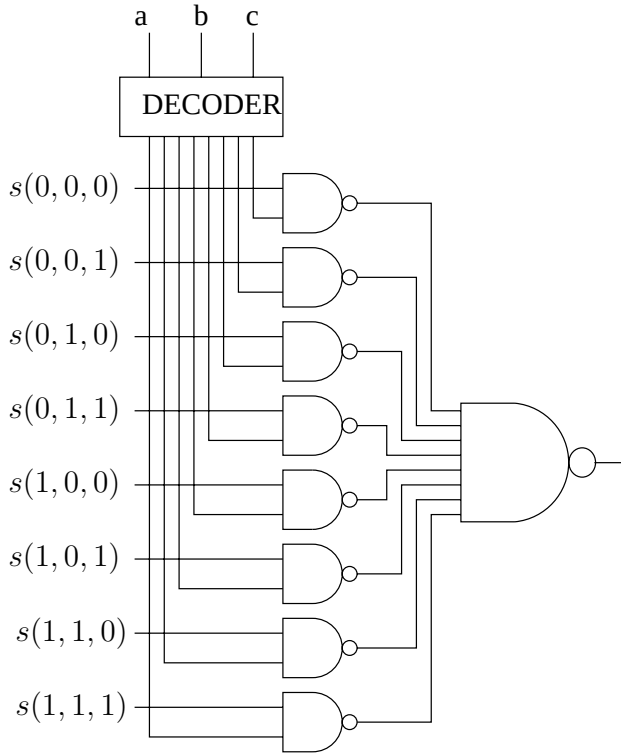


Cell balancing

Truth Table

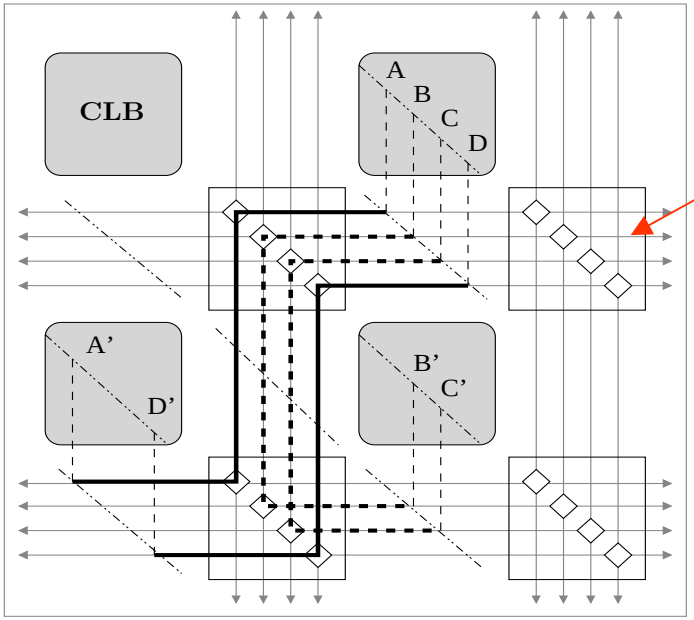


LUT3



Interconnect balancing 1

- Elmore's model: identical shape and length



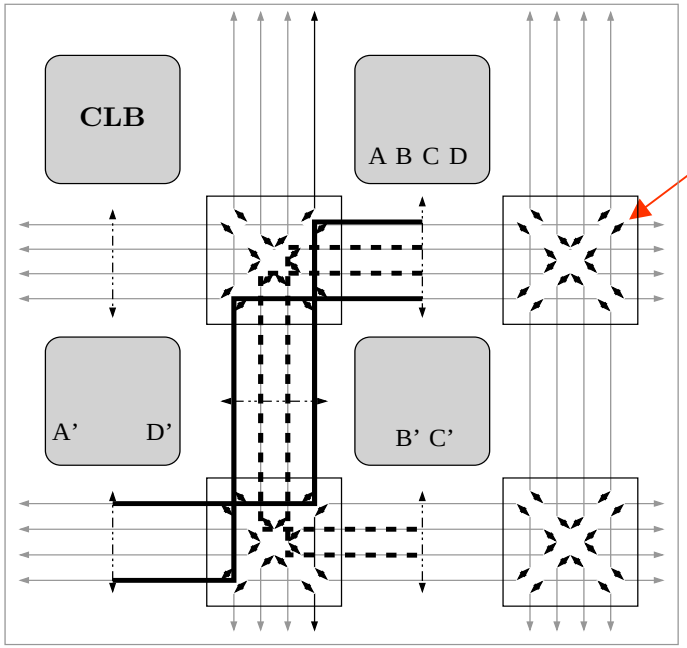
"subset" switch matric

TELECOM ParisTech

Coop. ENST

Interconnect balancing 2

- Crosstalk protection: twisted pairs

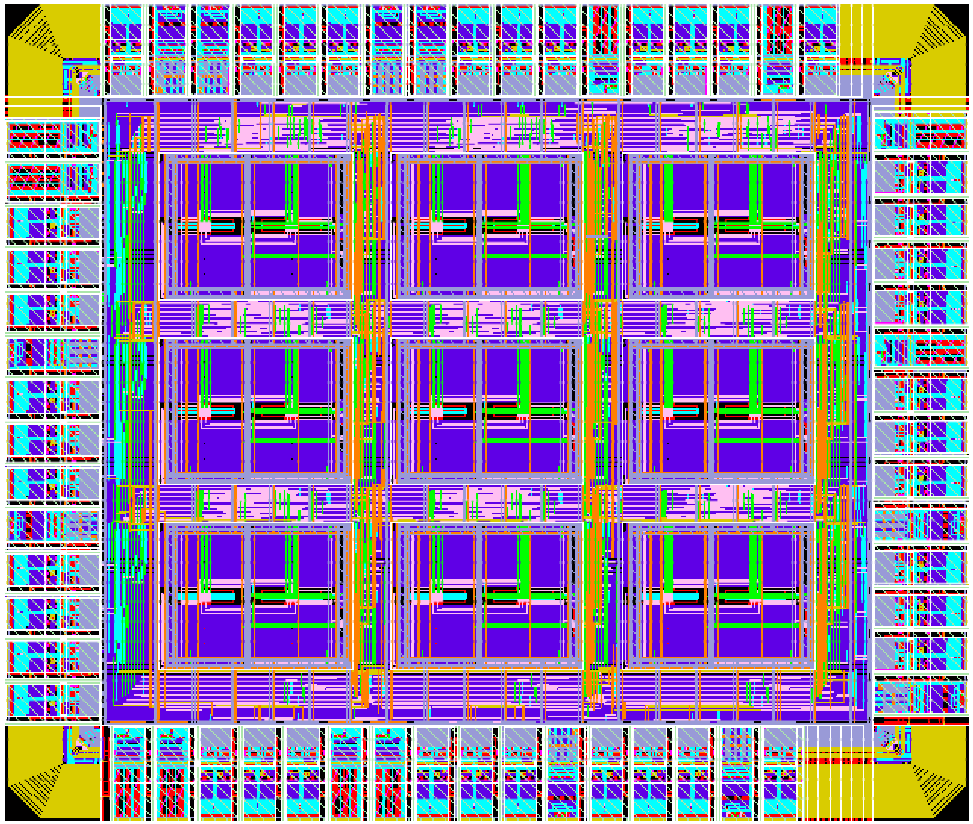


"twisted pair" switch Matrix

TELECOM ParisTech

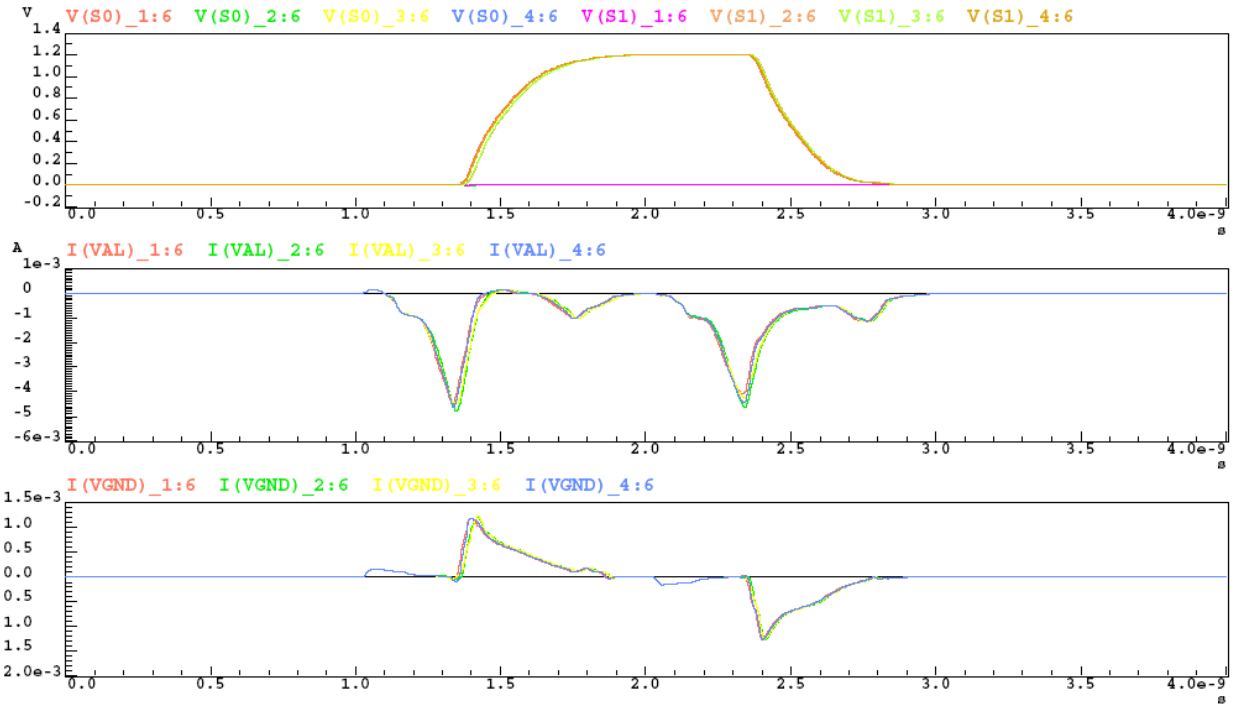
Coop. ENST

SAFE test chip layout

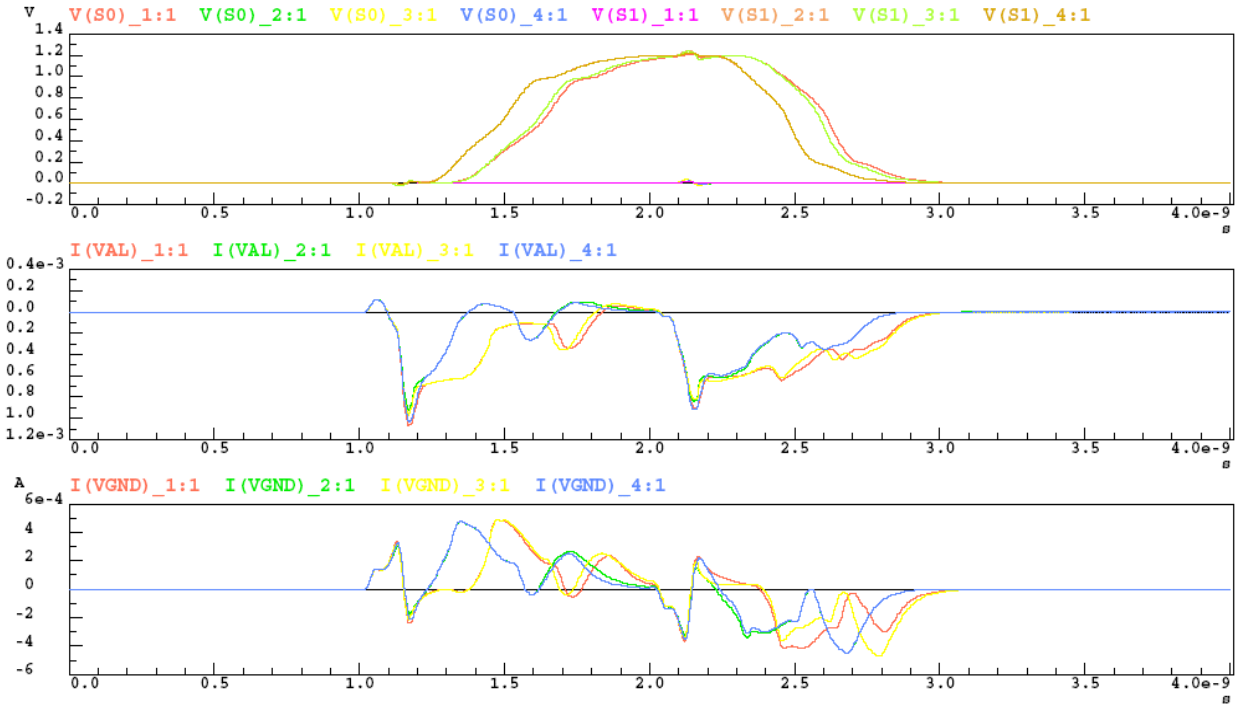


Dual rail QDI logic Mapping on SAFE FPGA

- The power consumption is data independent.



Dual rail QDI logic
Mapping on commercial FPGA



Summary

- About FPGA security
- Asynchronous logic principles
- A secure asynchronous FPGA
- Conclusion

Conclusion (FPGA)

- Mapping asynchronous logic onto standard FPGA is always possible
  - Designers have to deal with hazard-free and bounded-delay logics
  - Design cell libraries
- Many dedicated FPGAs to asynchronous logic have been designed
  - Most of them are style oriented
  - Multi-style logic is costly in area but …
  - Extremely flexible (protocol evaluation , EMA, …)

Conclusion

- Asynchronous Logic is a good candidate to counteract attacks at the logical and electrical levels
- Secure FPGAs offer a multi-level set of protections:
  - Intrinsic protections (internal : logic+ interconnect)
  - Extrinsic protections (application : logic+ interconnect + dynamic configuration)

But …

- Attacks will always be possible
- The security depend on the countermeasures complexity

Perspectives

- Evaluate the security of the asynchronous FPGA (test chip under fabrication)
- Define the best asynchronous logic styles (encoding, protocol) and interconnect structures
- Design an optimized asynchronous FPGA (Security and area)
- Develop a complete CAD flow for security applications (design and FPGA programming)
- Explore new FPGA topologies