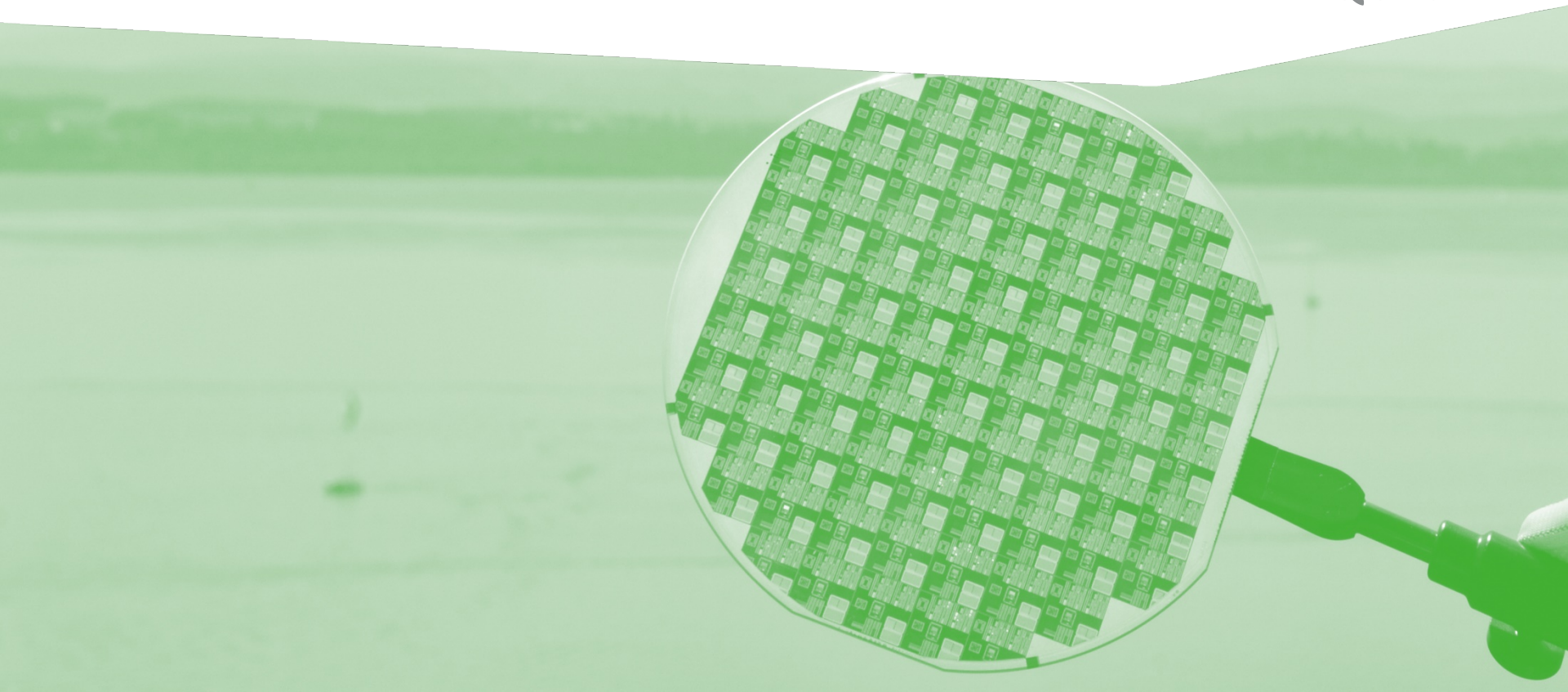
aptasic

MUCH MORE THAN ASIC TEST

DESIGN FOR TEST
DFX TECHNIQUES



Why doing DFT ?

Test is not the reproduction of verification

Verification need to target design specifications

Test should **cover** faults (**fabrication defects**)

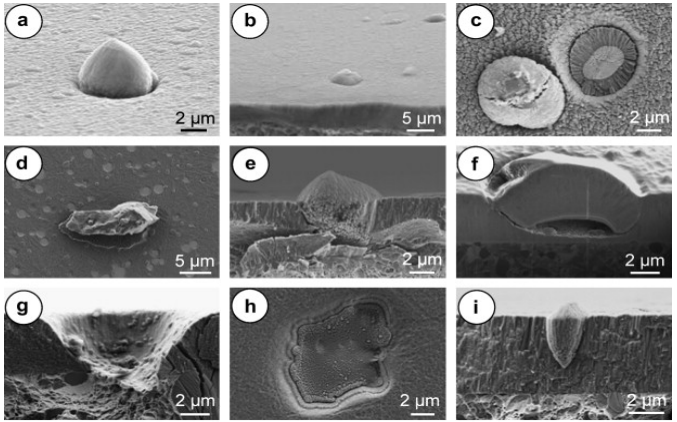
using direct or indirect methods and faults models

To guarantee

Economically **viable business**

Fast and **effective production** conditions

Handling of increased complexity (SoC, SiP)



DEFINITION

Fault – fabrication defect

Definition

A fault is present in the system when there is a **physical difference between the "good"** or "correct" system **and the current system**

Modeling

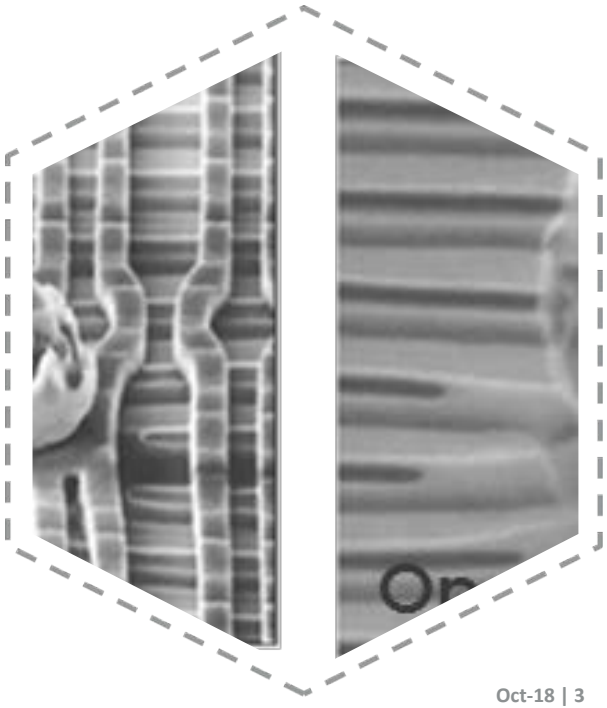
Geometric: define layout rules (FAB)

Switch-Level (transistor): **short, open**

Structural: stuck-at 1, stuck at 0

Delay: affect propagation delays

Functional: at speed test (RAMs, ROMs)



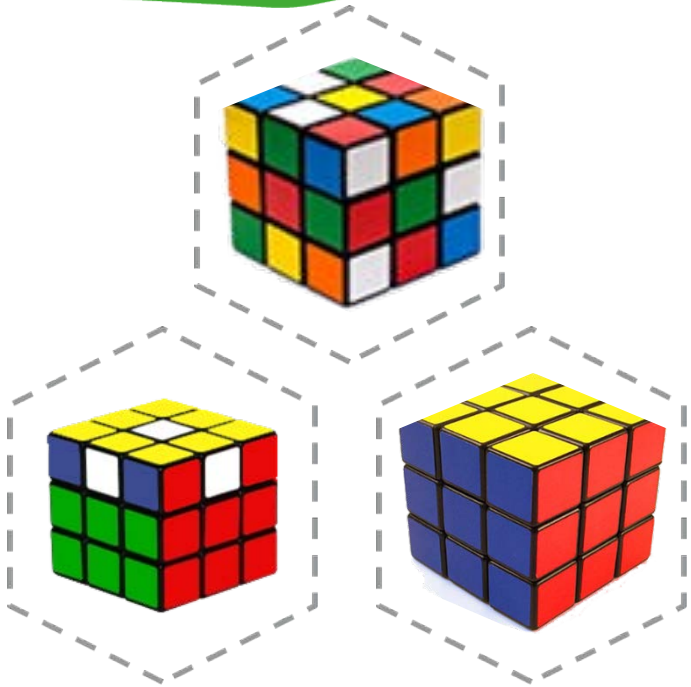
Impact of late detection

Late default detection can be huge

- 10x more for each step or even more …

Cost of delay

- Company image loss
- Missing window of opportunity
- Delayed time to cash (TT$)



DEFINITION

2 sides of same coin

DESIGN

Function

- IP blocks

JTAG

- Debug and trace

PINs/PADs out

- Interface (bus)
- Communication

Event triggered

TEST

Defects detection

- Fault models

JTAG

- Access mechanism

PINs/PADs out

- Access mechanism
- Test rail

Cycle based

What does DfT stands for?

Design for Test/Testability

- IC supply chain expert

Design for Troubles

- Designers view?
- Manager view?

… **Don’t forget Test!**

METHOD

Common Test Interfaces

JTAG (TAP)

- IEEE 1149.x standard, dedicated to test/debug

SPI

- Industry standard (Motorola)

I2C

- Industry standard (Philips/NXP)

Direct access

- Dedicated (ad-hoc) configuration

METHOD

Commonly used blocks

ADC/DAC

- Analog to Digital interfaces

RAM/ROMs

- Memories

OSC / PLL

- External / internal clocking

Digital cores

- Dedicated function

Test methods and existing concepts

Functional

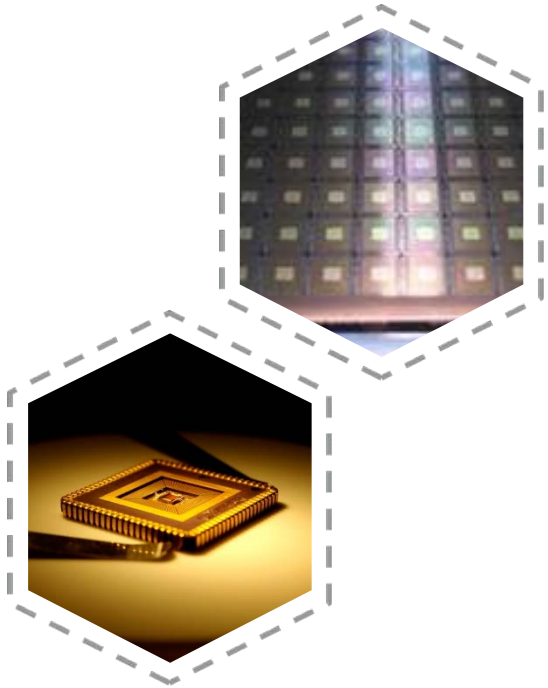
- Widely used but mostly inefficient

Structural

- Scan
- BIST

Idd(Q)

Analog BUS (IEEE1149.4)



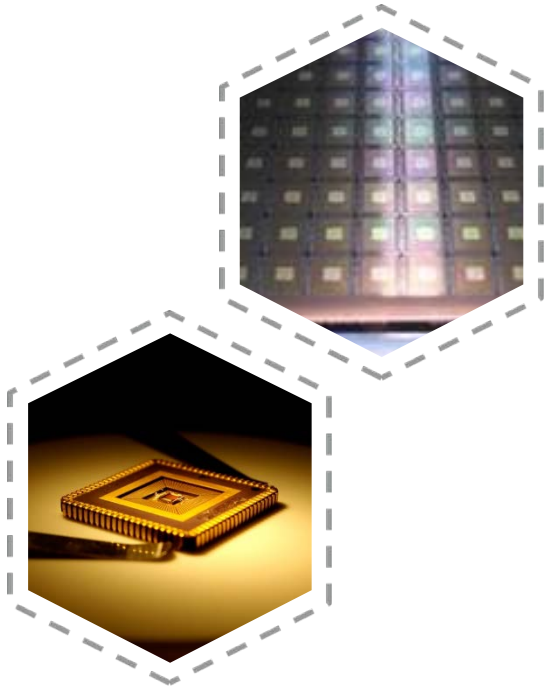
DfT methods I (ad-hoc)

Initialization facilities

Test point insertion

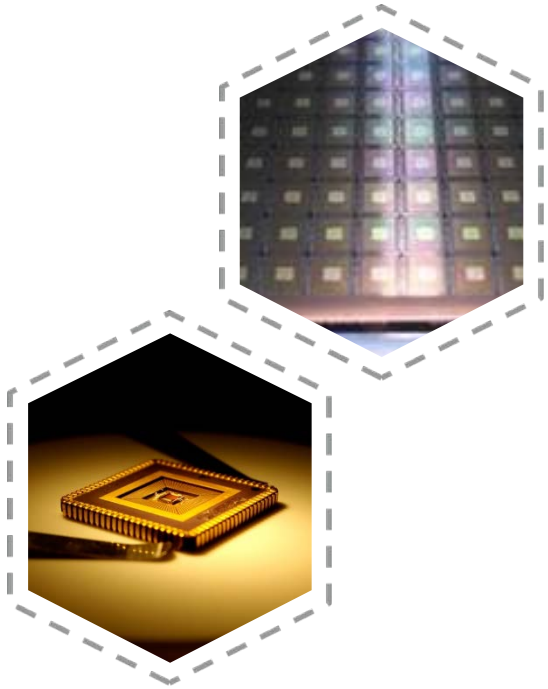
Partitioning

Access / multiplexing

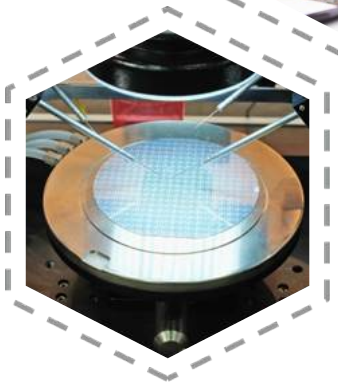


DfT methods II (structured)

- Scan
  - Full scan
  - Boundary scan
- IDDq
  - Quiescient current
- BIST
  - Memories
  - In situ test (e.g. PLL)



DfT summary



Target: **reduce test cost** of complex ICs

- **Shorten** test development time
- **Facilitate** testing at system level
- **Improve** product quality

Need: **ICs** that are **easy to test**

- Increase **accessibility / observability**
- Ensure **predictable** ICs responses

Trade-offs

- Technical: area, I/O pin, performance
- Economic: design time, yield, time to revenue

CONCLUSION

DfT starts with a KISS

Keep it simple and smart

- The best practices are the simplest ones

Not with long list of to do's

- Rather better a **short list of don't do**
- Or short list of **best practices**

Consider DfT very early in the design process

- Testability should be part of your SPECs

aptasic

MUCH MORE THAN ASIC TEST

THANKS FOR YOUR ATTENTION