Hardware Security

Paolo Maistri

paolo.maistri@univ-grenoble-alpes.fr

Part I
Embedded system design and architecture
Basic concepts

Outline – Part I

- **Embedded systems – general notions**
  - **Definition and applications**
  - **Vulnerabilities**
  - **Typical constraints**
  - **Overview of implementation technologies**

- **Basic blocks**
  - **Processors**
  - **Communication resources**
  - **Programmable arrays**

Embedded systems?

- It is generally accepted to be a **hw/sw system** designed to solve a specific (industrial) problem or task
  - This is in contrast to a general-purpose computer such as a PC or workstation
- One or several microprocessors (combined with other hardware and software)
- Embedded SW ranges from a small executable to a large real-time operating system (RTOS) with a graphical user interface (GUI)
- Typically, the embedded system software must **respond** to events **in a deterministic way** and should be guaranteed not to crash

Embedded systems: general properties

- **Single- (or limited-) function**
  - **Typically, is designed to perform predefined function**
- **Often tightly constrained (non functional requirements)**
  - **Tuned for low cost (+ predictable time to market)**
  - **Single-to-fewer components based, small foot-print**
    - **Limited amount of resources (memory, …)**
  - **Performs functions fast enough**
    - **Differs from high performance markets**
  - **Consumes minimum power (battery …)**
  - **High expectations: reliability, security, etc.**
- **Reactive and real-time**
  - **Must continually monitor the desired environment and react to changes**
  - **Worst Case Execution Time (WCET) evaluation constraints (determinism !)**
- **Hardware and software co-existence (+ HW/SW interface !)**

Embedded systems: application areas

- The embedded system landscape is as diverse as the world's population:

  … no two systems are the same.

- Embedded systems range from large computers such as an air traffic control system to small computers such as a handheld device that fits into any pocket.

- **Examples:**
  - **Communication devices:** wired and wireless routers and switches
  - **Automotive applications:** braking systems, traction control, airbag release systems, and cruise-control applications
  - **Aerospace applications:** flight-control systems, engine controllers, auto-pilots and passenger in-flight entertainment systems
  - **Defense systems:** radar systems, fighter aircraft flight-control systems, radio systems, and missile guidance systems

Embedded systems and dependability

- Increasing number of application fields ranging from consumer electronics to very critical application fields:
  - Transports
  - Energy (e.g. nuclear plants)
  - Human security/protection devices (e.g. factory secured areas)
  - Access control, identity
  - Pay-per-view TV or similar services, banking
  - …
- Note: criticality comes from several points of view – from human injuries to financial losses

Embedded systems => Integrated circuits

YAES (Yet Another Embedded System)

Networks, trusted infrastructures, trusted communications

Electronic payments

Electronic passport, access control

Ubiquitous computing and privacy

Counterfeiting, intellectual property

YAES (Yet Another Embedded System)

- Unauthorized access to services
- Illegal duplications
- Identity
- Financial frauds
- …

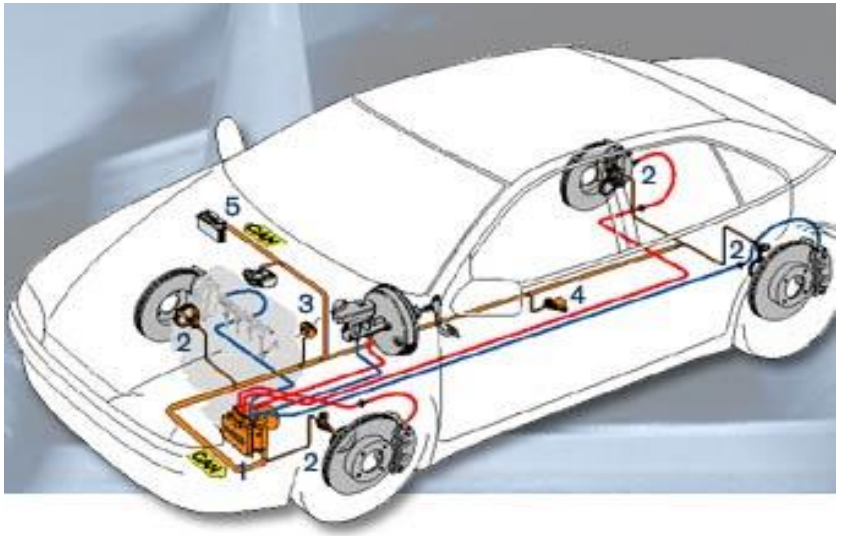
Automotive embedded systems: overview

- Today's vehicle networks are truly distributed electronic systems (70+ nodes (=ECUs) [1]).

- Cars contain numerous (10+) heterogeneous time or event driven bus systems
  - CAN, LIN, FlexRay, MOST

Most are critical

- x-by-wire
- steering aids, ABS, ESP(DSC)
- remote window and lock control
- engine control
- airbag control
- navigation systems
- entertainment systems



[1] P. Hansen. New s-class mercedes: Pioneering electronics. *The Hansen Report on Automotive Electronics*, 18(8):1–2, October 2005.

Automotive embedded systems: typical properties

- **Software is mission critical**
  - **Highly dependable**
  - **Hard real-time**
  - **Typically statically scheduled and bound**
- **Lifetime is rather long (10-14 years)**
  - **Modular design**
  - **Exchangeable components (modules)**
- **Systems are produced in high quantities (56.3 million cars in 2005)**
  - **Costs have to be small**
  - **Bug fixes are extremely expensive**

Note: Safety rather than security (BMW vs smoke) …
until today …

Car Hijacking

- The vents in the Jeep Cherokee started blasting cold air
- the radio switched to the local hip hop station
- the windshield wipers turned on
- a picture of the two hackers performing these stunts appeared on the car's digital display
- They cut the transmission. … my accelerator stopped working. … they cut the brakes, leaving the SUV slid into a ditch
- Practically all carmakers are turning the automobile into a smartphone […] whose cellular connection also lets anyone gain access from anywhere

Car Hijacking… again

> Security flaw affecting more than 100 car models exposed by scientists

- **A major security flaw in more than 100 car models**
  - **Citroën, Fiat, Honda, Volvo, Volkswagen…**
  - **… Porsches, Audis, Bentleys and Lamborghinis**
- **Arduino board + Software-Defined Radio**
- **4 different master keys + auxiliary secret value**
  - **Eavesdropping…**
- ***« Dismantling Megamos Crypto »* Submitted at Usenix '13, withdrawn, then published two years later**

Car Hijacking… Learning with experience?

News

Tesla Model X hacked with $195 Raspberry Pi based board

November 24, 2020 Nitin Dahad

- "Using a modified electronic control unit (ECU), obtained from a salvaged Tesla Model X, […] we were able to wirelessly compromise a key fob and take full control over it. Subsequently we could obtain valid unlock messages to unlock the car later on."
- After approaching the vehicle and unlocking it we can access the diagnostic connector inside the vehicle. By connecting to the diagnostic connector, we can pair a modified key fob to the car.

- **a Raspberry Pi computer ($35) with a CAN shield ($30), a modified key fob and ECU from a salvage vehicle ($100 on eBay) and a LiPo battery ($30)**

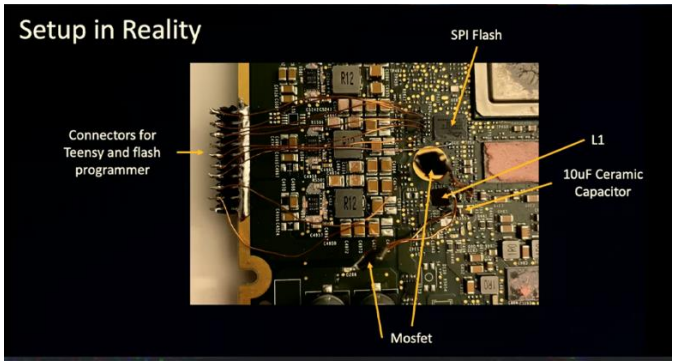
Lesson Learned… ?

SPIEGEL Net world

Berlin hackers crack Tesla's autopilot

Three IT security researchers were able to penetrate deep into the hardware of Tesla's driving assistant and access company secrets. They even managed to activate a powerful "Elon mode".

By **Max Hoppenstedt**
December 27, 2023, 6:37 a.m



Three IT security researchers from Technische Universität Berlin (TU Berlin) glitched Tesla's driving assistant into activating a powerful "Elon mode" and were able to access the company's secrets, Spiegel reported. Allegedly, all Tesla models are vulnerable to this attack.

With tools that cost around 600 euros, TU Berlin students Christian Werling, Niclas Kühnapfel, and Hans-Niklas Jacob induced a short two-second voltage drop by 560 millivolts and rooted the ARM64-based circuit board of Tesla's autopilot.

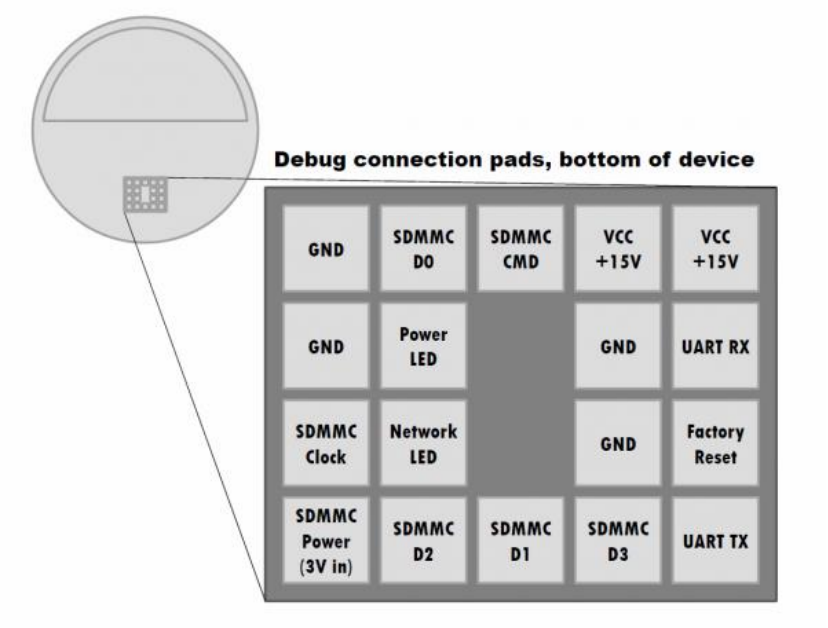
The voltage glitch enabled researchers to extract arbitrary code and user data from the system, including cryptographic keys and important system parts, allowing them to reconstruct how it works. Researchers even gained access to a video with GPS coordinates that had been deleted by the previous owner of the vehicle, as it was not overwritten.

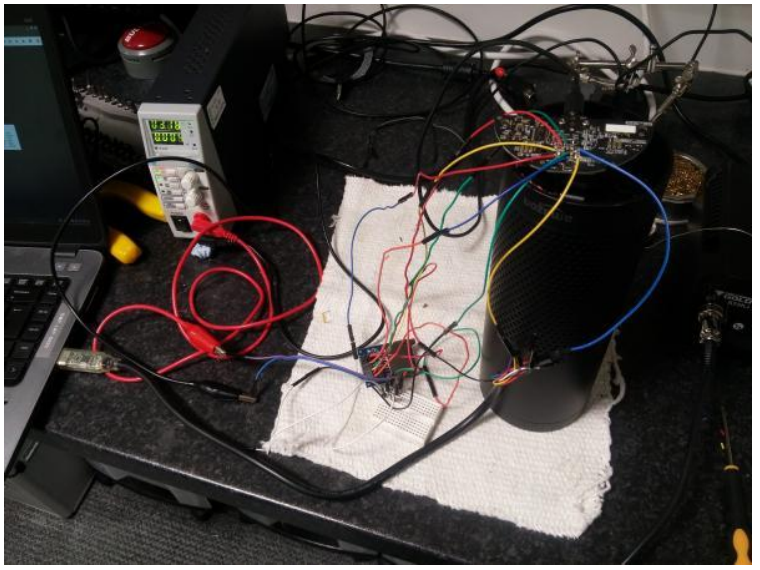
Tesla Model 3: "I would have thought that the company would protect this valuable intellectual property better" Photo:Tobias Schwarz / AFP

[https://www.youtube.com/watch?v=AgC9OiFrIPk]

Alexa, are you listening? [Mark Barnes, Aug 2017]

- **Amazon Echo 2015/16**
  - **Exposed debug pads on the base of the device**
  - **Hardware configuration setting which allows the device to boot from an external SD Card**





- **Grant remote access to the device; steal customer authentication tokens; stream live microphone audio to remote services**

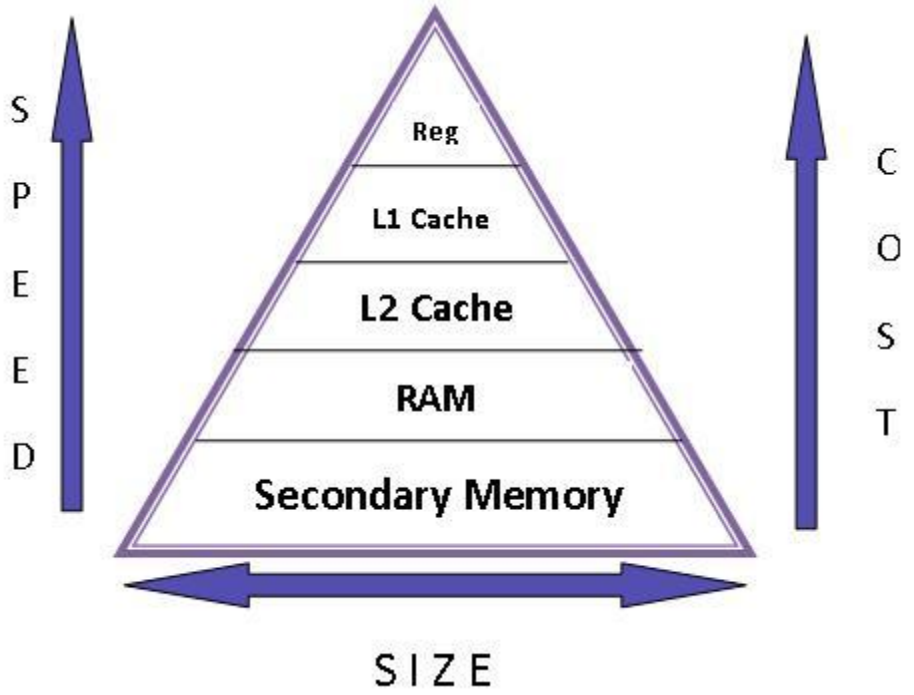
IoT Goes Nuclear: Zigbee Chain Reaction

- Compromise a single light globe from up to 400 metres away
- Worm spreads from a single smart bulb to those nearby
  - Universal key extracted using power analysis attack

https://www.youtube.com/watch?v=zcwz-lQtCwM

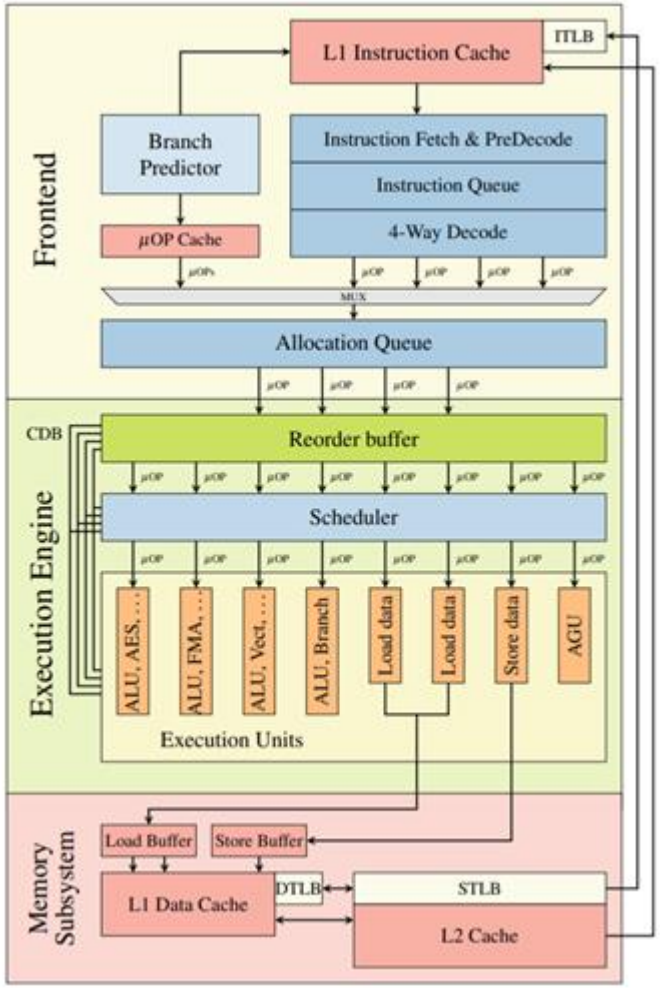
Cache Attacks [NeveJean-Pierre Seifert, 2006]

- Multi-process system
- Memory hierarchy: different speed granularity
- Shared memory architecture

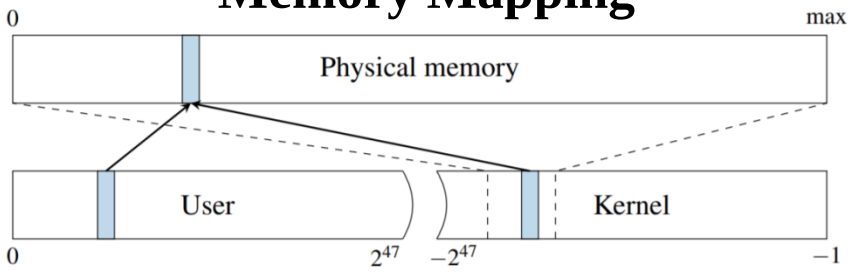


Meltdown [Google Project Zero + Many Academics, 2018]

Out-of-Order Execution



Memory Mapping



Attack

https://www.youtube.com/watch?v=RbHbFkh6eeE

Other example … implanted devices

From …

… to …

Today, a pace-maker is a small computer able to analyze the heart electrical signals that can be tested and reprogrammed by telemetry (contactless communications)

Dependability constraints (with minimum power consumption !):

Reliability: difficult replacement !

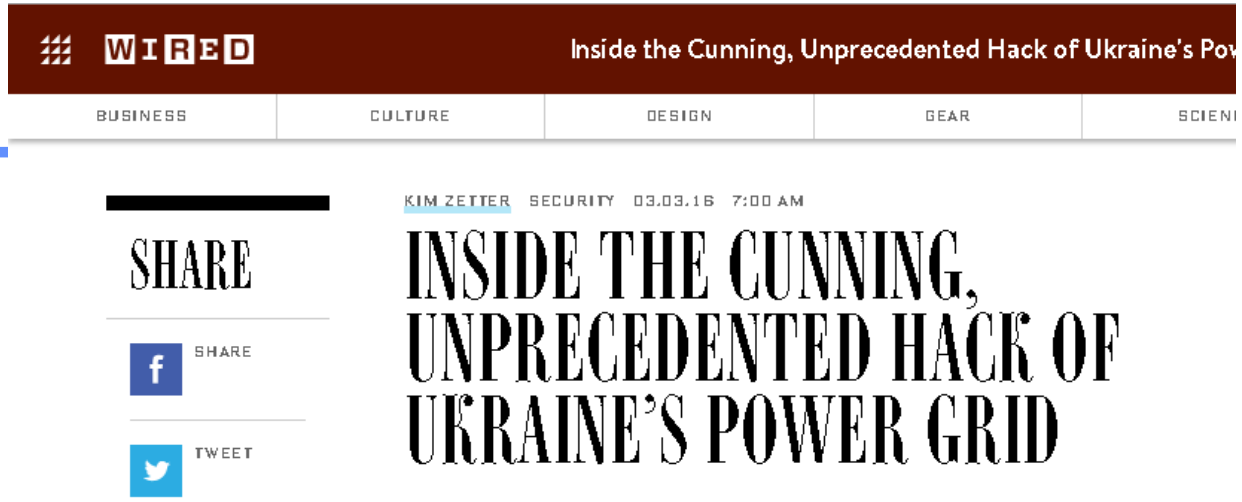
Safety: EM perturbations (e.g. iPod …)

http://www.techshout.com/ipod/2007/11/ipods-can-cause-pacemakers-to-malfunction-study/

Security: malicious activation/deprogrammation by a hacker

http://www.newscientist.com/blog/technology/2008/03/death-by-radio-waves-hacking-pacemaker.html

Power Grids



- **Three attacks. Thirty minutes apart. Against three electrical substations serving Ukraine's power grid.**
  - **First known cyber-attack of its kind**
  - **Confirmed by US Cyber Emergency Response Team**
- **Timeline**
  - **Phishing campaign opening backdoors**
  - **Explore and map the networks**
  - **Harvest VPN credentials**
  - **Access the SCADA network**

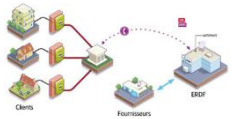
Targets are increasing with evolutions

- **Technological evolutions combined with social evolutions**

  - **Counterfeiting fighting: luxury products, but also automotive spare parts …**

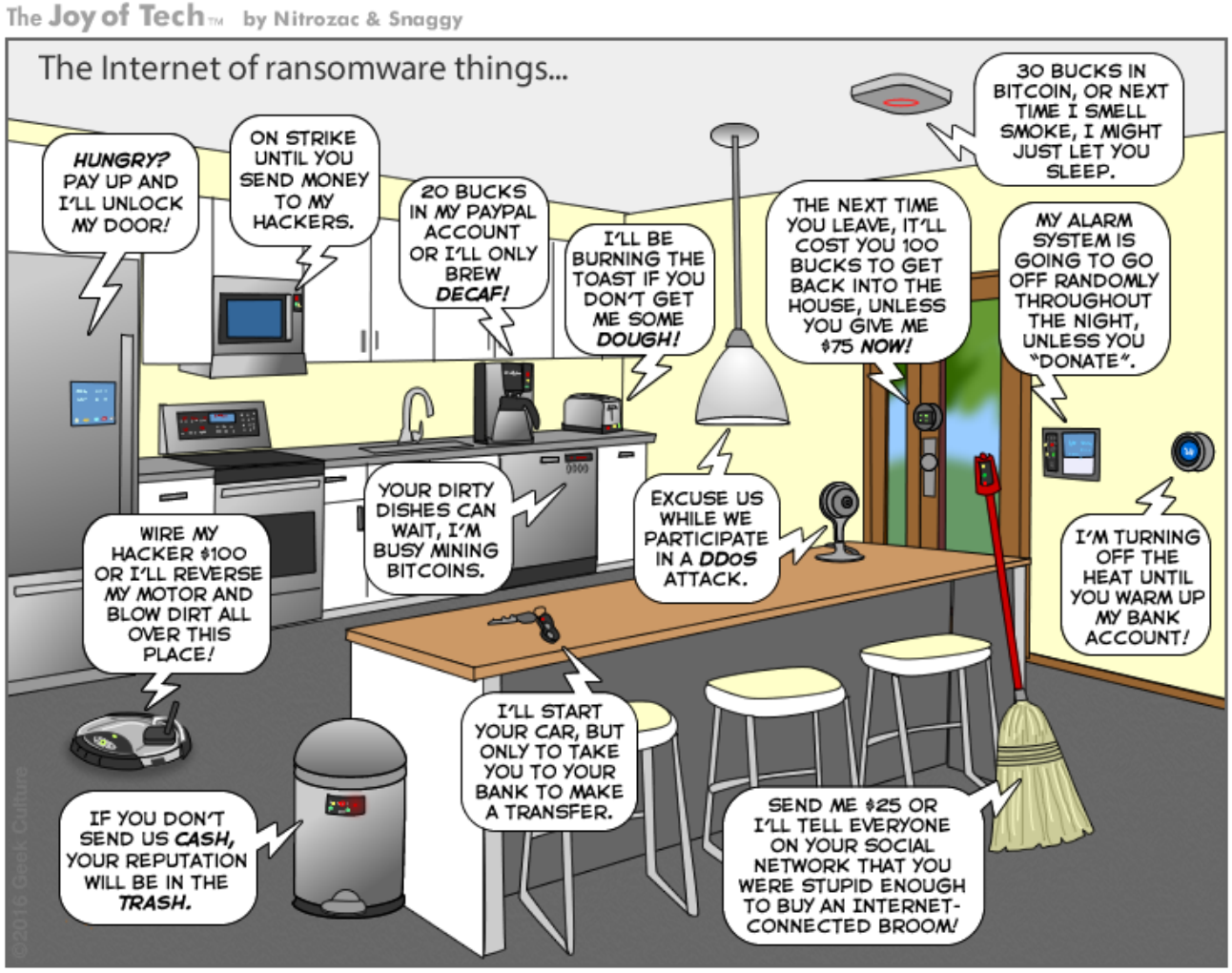
  - **More devices related to health (available on the market): communicating pacemakers, insuline pumps, …(tomorrow: NeuraLink ?)**

  - **More remote (electronic) control on energy-related devices ("smart grids"): a new target for potential hacking, must be secured**



  - **Future automotive systems ("X-by-wire"): brakes, but also all other controls (e.g. steering) => not only safety-related aspects have to be taken into account during design**

Internet of [Ransomware] Things



Security definition

- Security = one attribute of dependability
- Security is the concurrent existence of several pillars
  - Availability (but for authorized users only => Confidentiality),
  - Integrity = absence of improper system alterations; with "improper" meaning "unauthorized",
  - Authenticity (information comes from trusted source)
  - Resilience (or robustness w.r.t. attacks against previous pillars + limiting the potential damage in case of successful attack)
- Often linked to the use of cryptography (or steganography) => most examples will be given on crypto-processors or cryptographic accelerators
- System-level security policies will not be discussed here

Tampering secure circuits

- A secure circuit contains secret data (e.g. a secret cryptographic key)
- Knowing the secret grants unauthorized privileges (access, message interception, …)
- May allow in some cases to clone a device (e.g. pay-per-view TV decoder … or a bus/subway card …)

Outline – Part I

- **Embedded systems – general notions**
  - **Definition and applications**
  - **Typical constraints**
  - **Security concerns**

- **Basic blocks**
  - **Overview of implementation technologies**
  - **Processors**
  - **Communication resources**
  - **Programmable arrays**

Implementation technologies

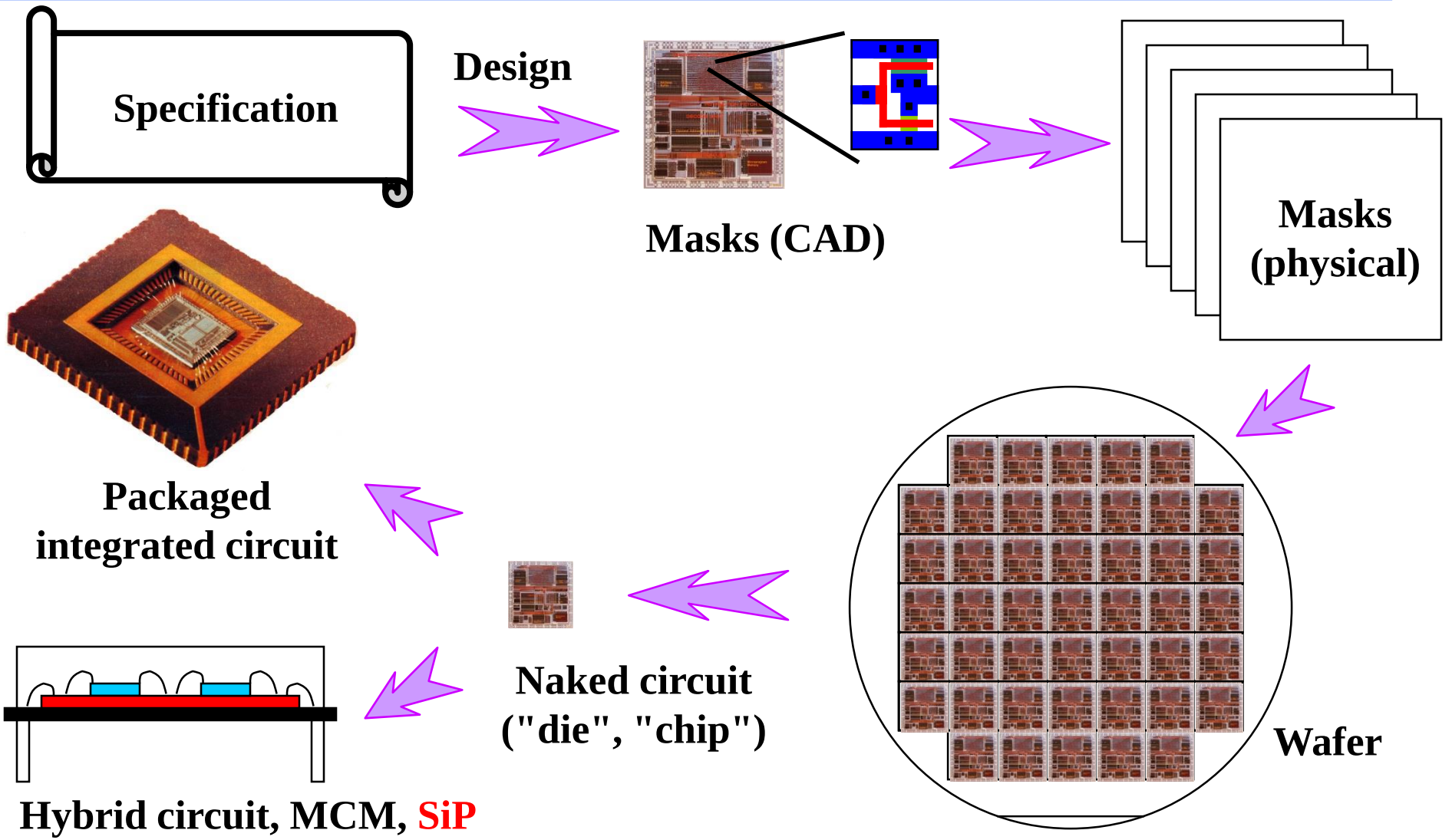
- Microcontroller-based systems
- DSP-based systems

} Electronics board or System on Chip (SoC)

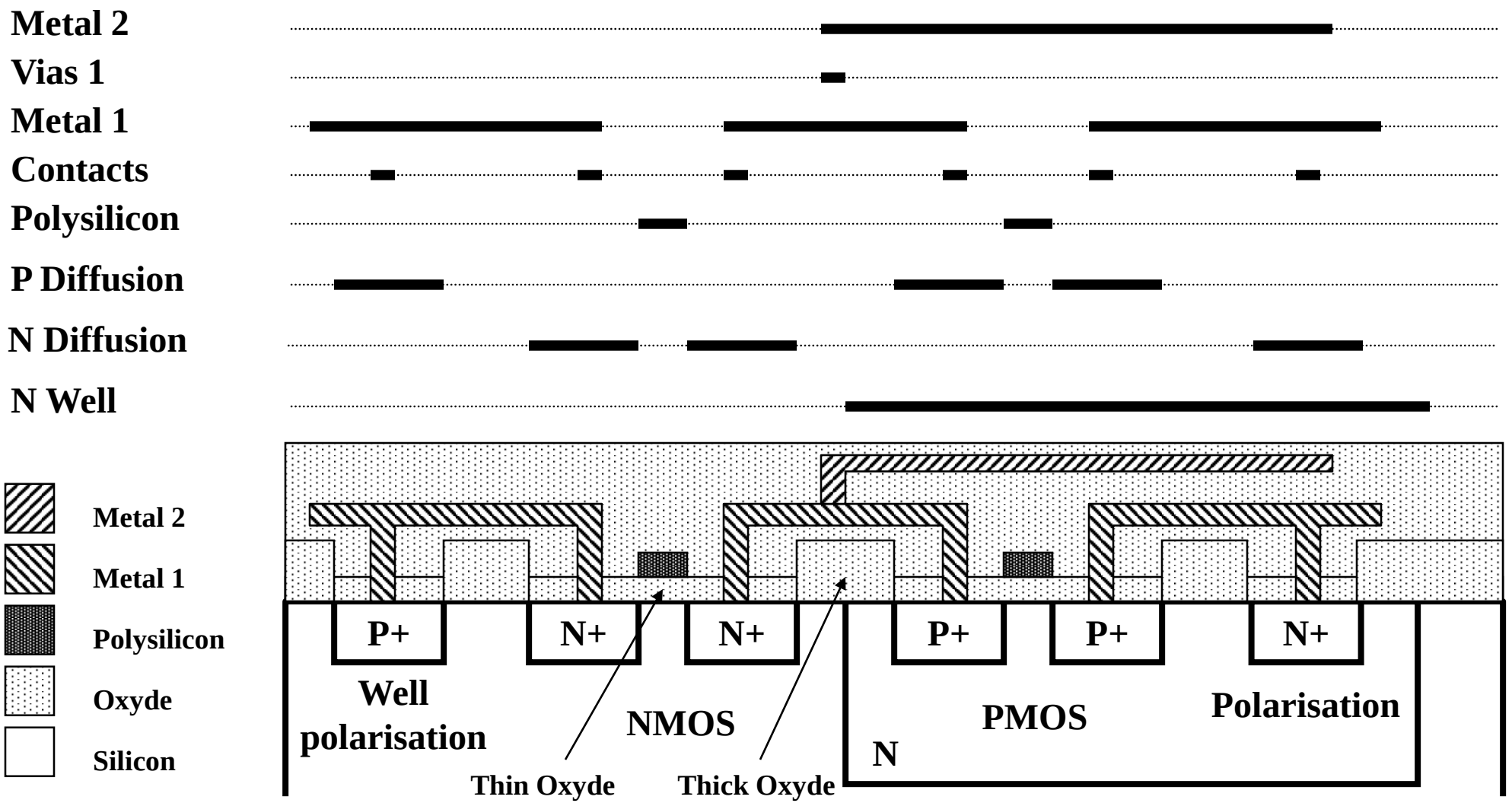
- ASIC technology
- Programmable array (FPGA) technology
  => Reconfigurable architectures

- Embedded system design involves:
  - Hardware (Board/ASIC/FPGA) design
  - Drivers for hardware (C …)
  - Software development

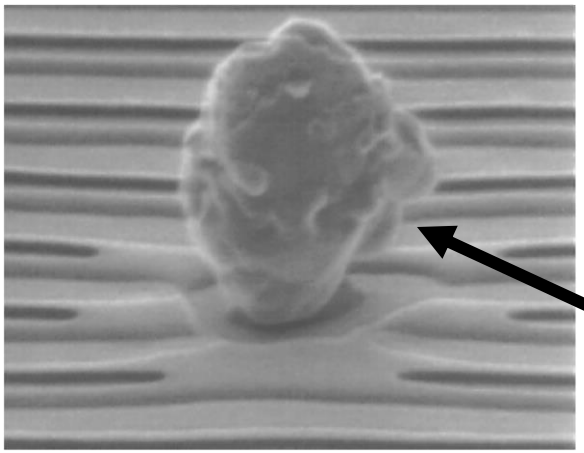
From design to physical component



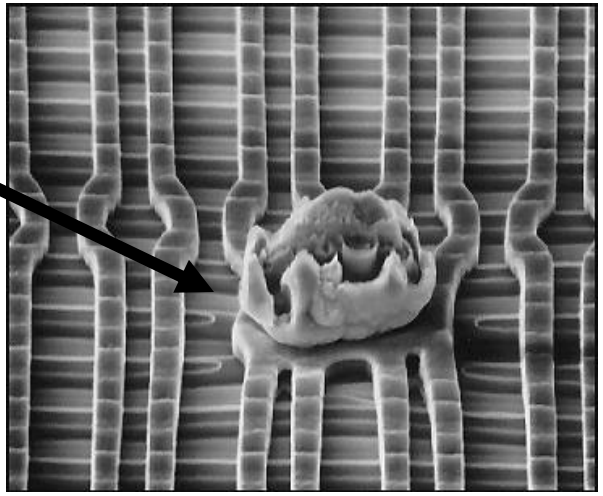
Simplified masks (CMOS inverter)



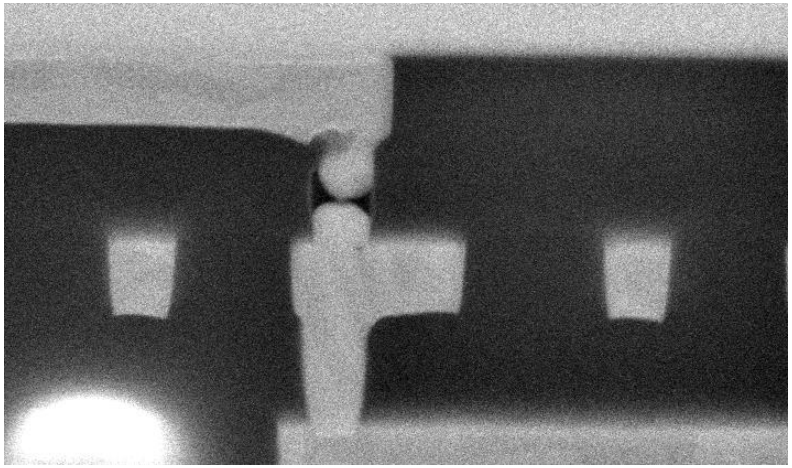
Manufacturing defects: examples

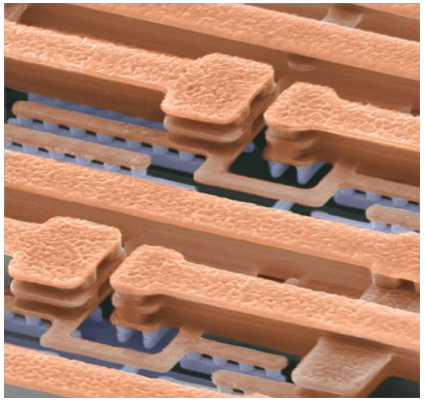


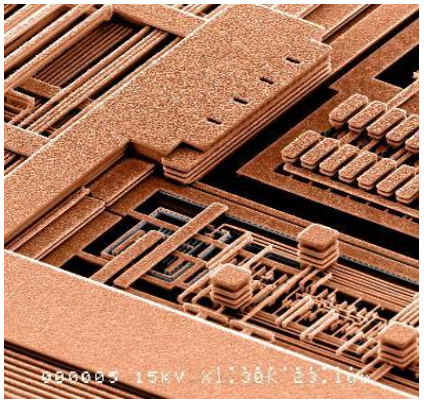
Unexpected connection
=> short circuit



Interconnection break
=> open circuit

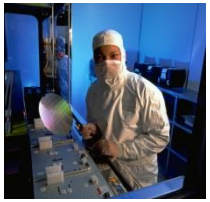






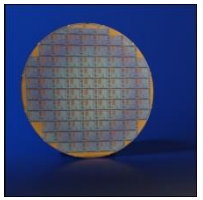
Source: IBM

After manufacturing: Testing!



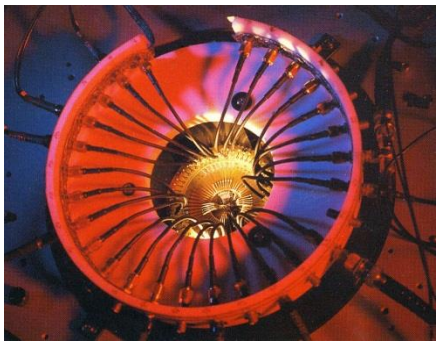
Manufacturing (process)

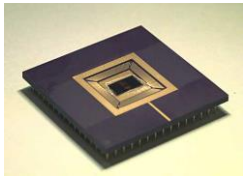
Visual controls (options)



Processed wafers

Probing-based tests





Die slicing/ assembling

Test in package
- **parametric**
- **consumption**
- **functional (nominal and extrem conditions)**
- **dynamic (performances)**

Accelerated aging

Burn-in (option)

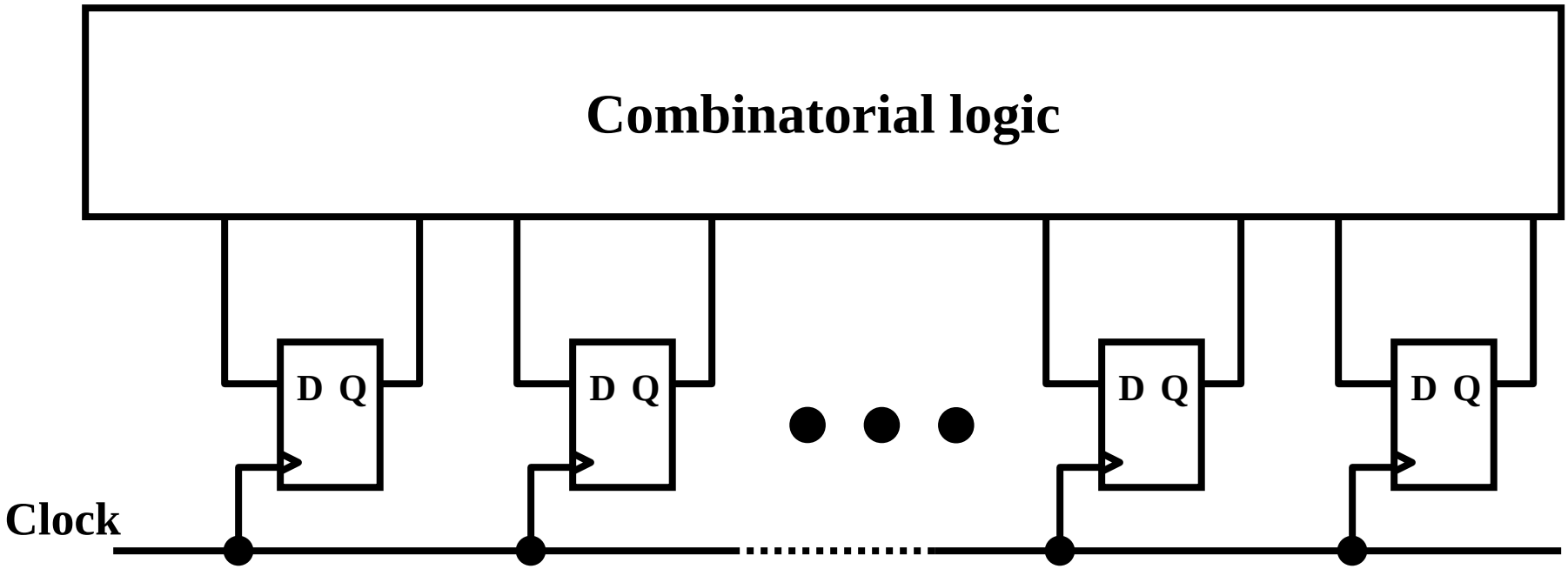
Distribution

Tests: need for DfT (Design for Testability)

- Test must be prepared at each design step
- Various techniques
  - Helping external testing with ATE
  - or performing (partial) self-test
- Classical approach: scan test (Boundary scan at board level)
  - Internal memory elements (flip-flops) connected in serial chain(s) or shift register(s) + I/Os if boundary scan
  - Simple external control signals for complete or partial memory controllability and observability
- Security constraints opposed to testability contraints !!
  - Scan chain = backdoor for hackers
  - Maximum (test) vs. minimum (security) controllability/observability

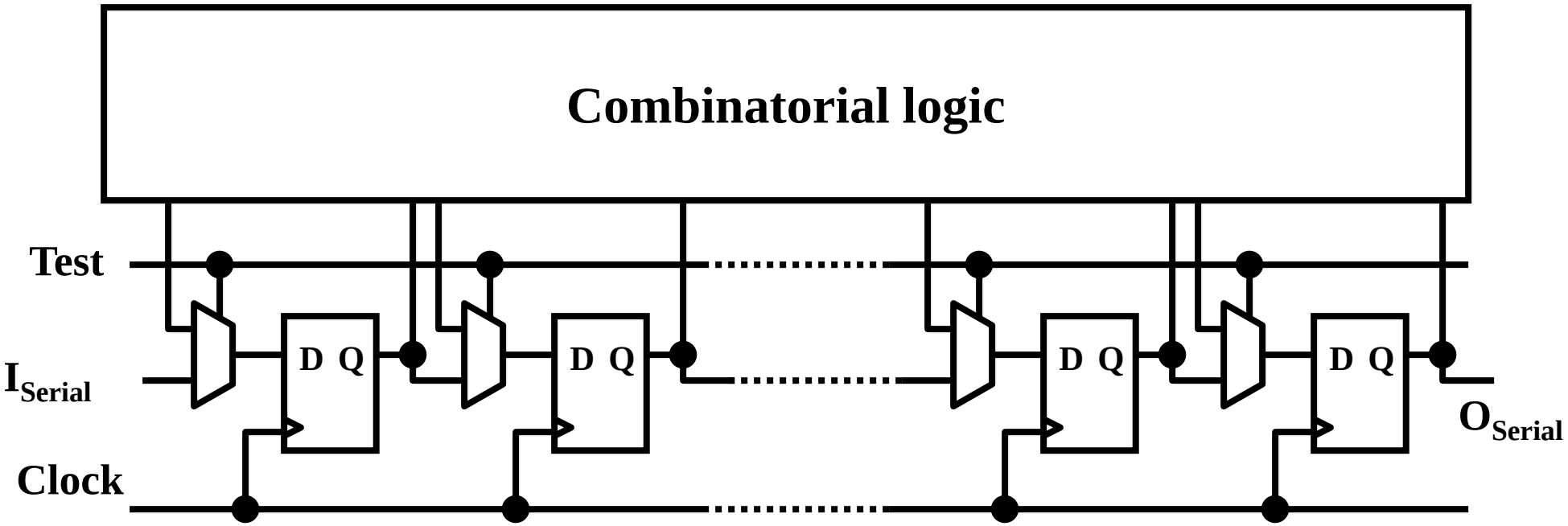
"Scanpath" implementation – basics (1)

Initial circuit:

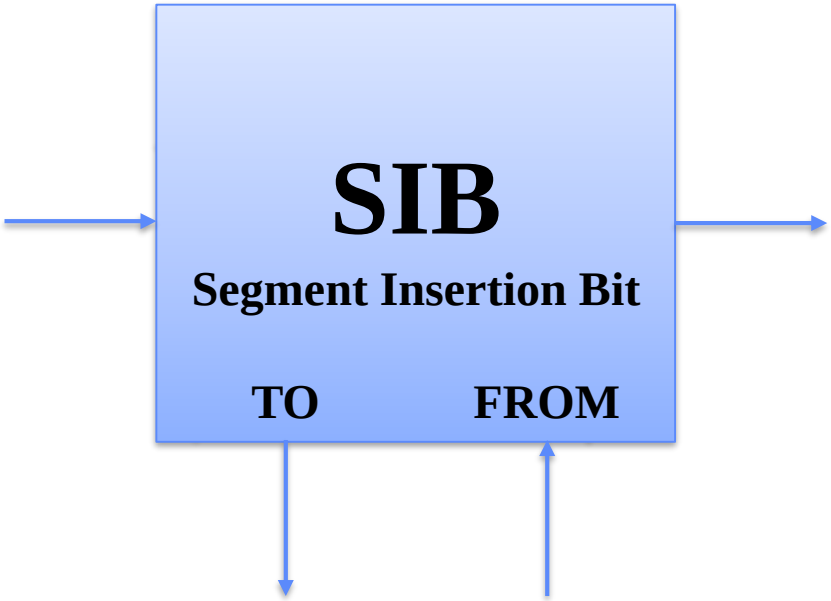


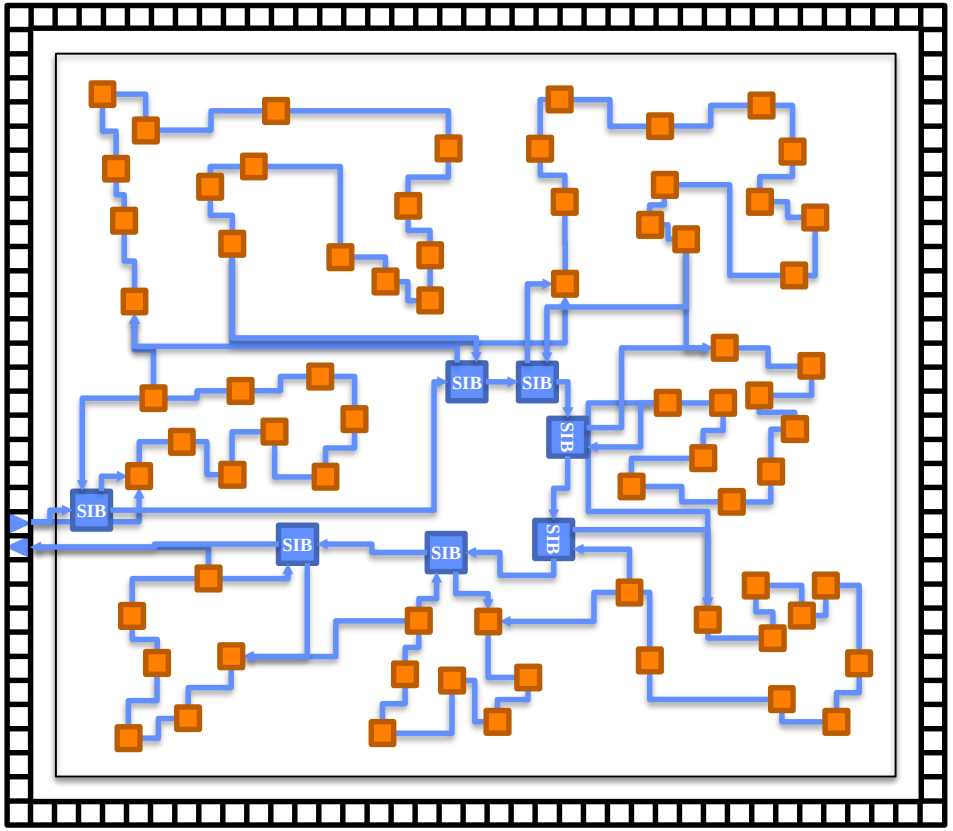
"Scanpath" implementation – basics (2)

Circuit with a single scan chain:



Reconfigurable Scan Networks





Main circuit design constraints …

- Area and yield
- Speed (clock frequency / computing power)
- Power/energy consumption and heat dissipation
- Pin number (off-chip interconnections)
- On-chip interconnections
- Testability
- Electromagnetic characteristics (emission, susceptibility)
- Robustness / dependability (including security)

- + Time-to-market, cost, …

Constraints: link with security ??

- Area and yield ? Yes, limits protections
- Speed (clock frequency / computing power) ? Yes, limits protections and/or increases susceptibility to faults
- Power/energy consumption and heat dissipation ?
  - Dynamic consumption: Yes, side channel
  - Temperature: Partially, temperature-induced faults
- Pin number (off-chip interconnections) ? Not directly
- On-chip interconnections ? Yes, in particular dummies
- Testability ? Yes, potential backdoor
- Electromagnetic characteristics (emission, susceptibility) ? Yes, side channel
- Robustness / dependability ? Yes, fault-based attack detection

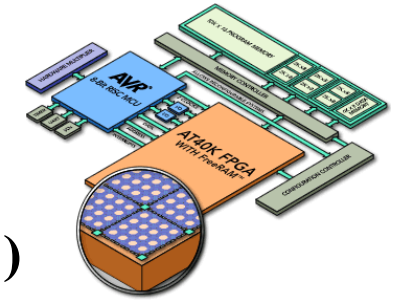
"Specific" Circuits

- **Dedicated to a given application, or a small application category**
  - **ASIC: Application Specific Integrated Circuit**
  - **Opposed to "general usage" circuits, available "on the shelf" (COTS)**
- **Specificity possible at different levels**
  - **Specific manufacturing (mask-level configuration)**
  - **Generic manufacturing, user configuration**
    => "programmable arrays": PLD, CPLD, FPGA
- **Many architectural/design common notions**
- **Focus on Programmable Arrays in the sequel (due to increasing ASIC NRE costs + lab work feasibility), in spite of limitations**

Programmable arrays

- A few thousands … (www.ednmag.com)

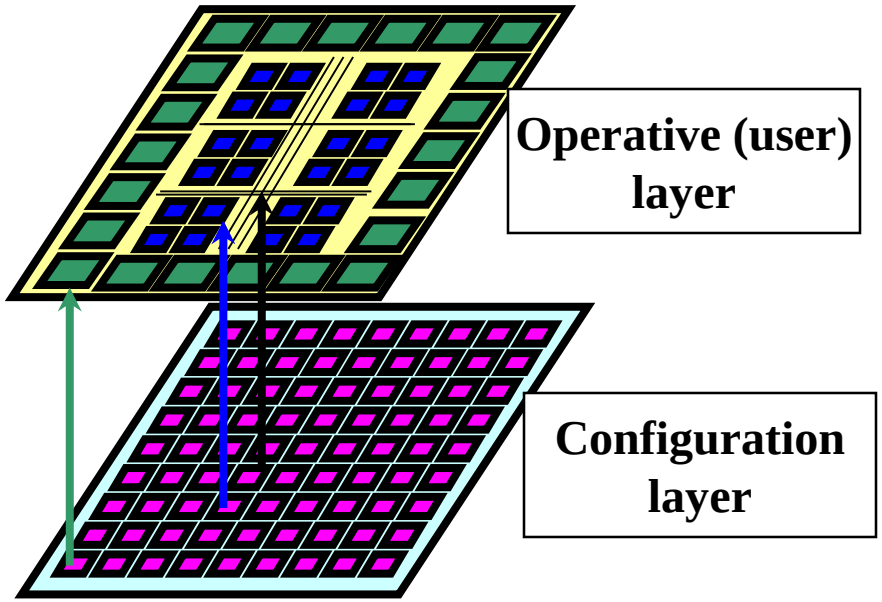
- Several classification criteria
  - Elementary cell architecture type
    - AND/OR arrays: sum of products => PLD
    - Elemental functions (Mux-based or LUT-based) => FPGA
  - Logic complexity
    - SPLD (one block: PAL, PLA, PLD, …) vs. CPLD (multiblocks)
    - FPGA (elemental cell array) vs. SoPC or "FPSLIC" (FPGA+optimized hard processor core)
  - Programmation technology
    - Fuses / anti-fuses
    - PROM, EPROM, EEPROM (PLD, EPLD, EEPLD, …)
    - SRAM, Flash



FPGAs vs. ASICs

- **Permanent configuration: basically equivalent to ASICs (user layer only)**

- **SRAM-based configuration: huge number of memory cells, errors in the configuration layer cannot be neglected**



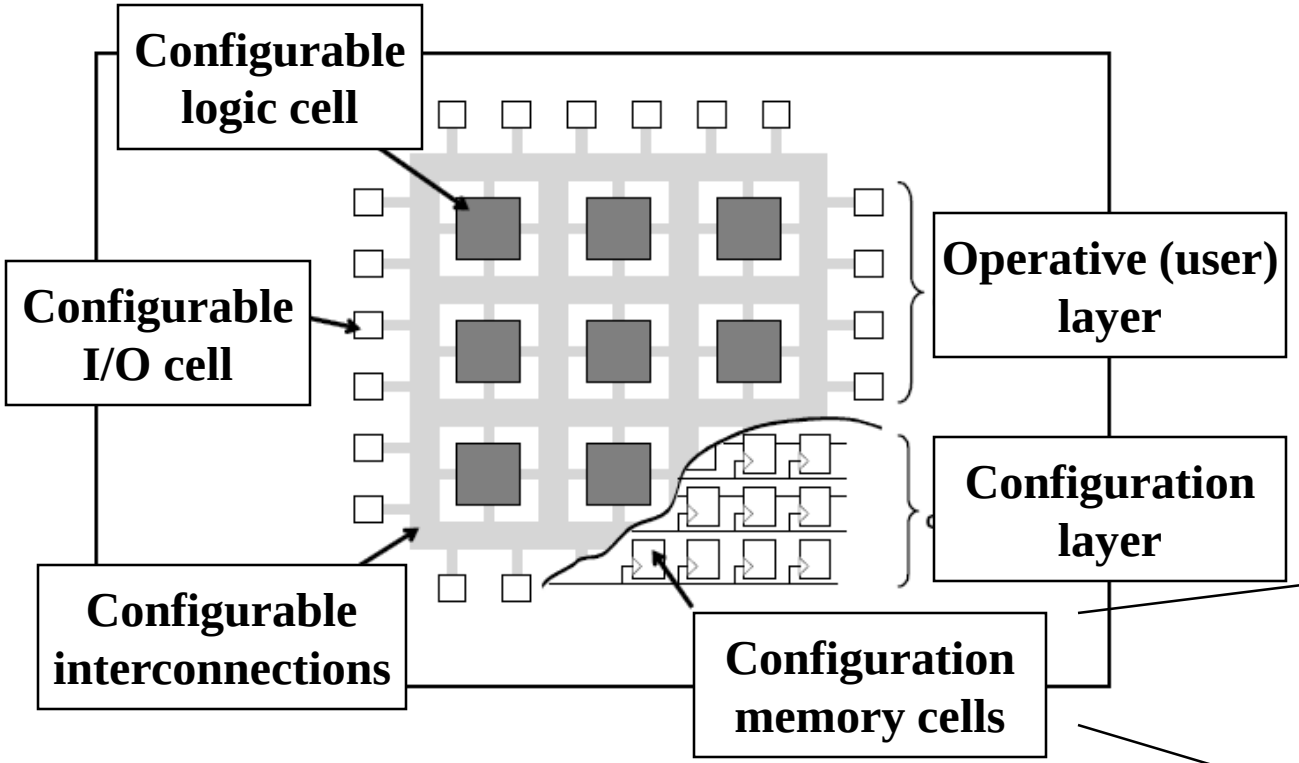
I/O blocks

Logic blocks

Interconnections and switches

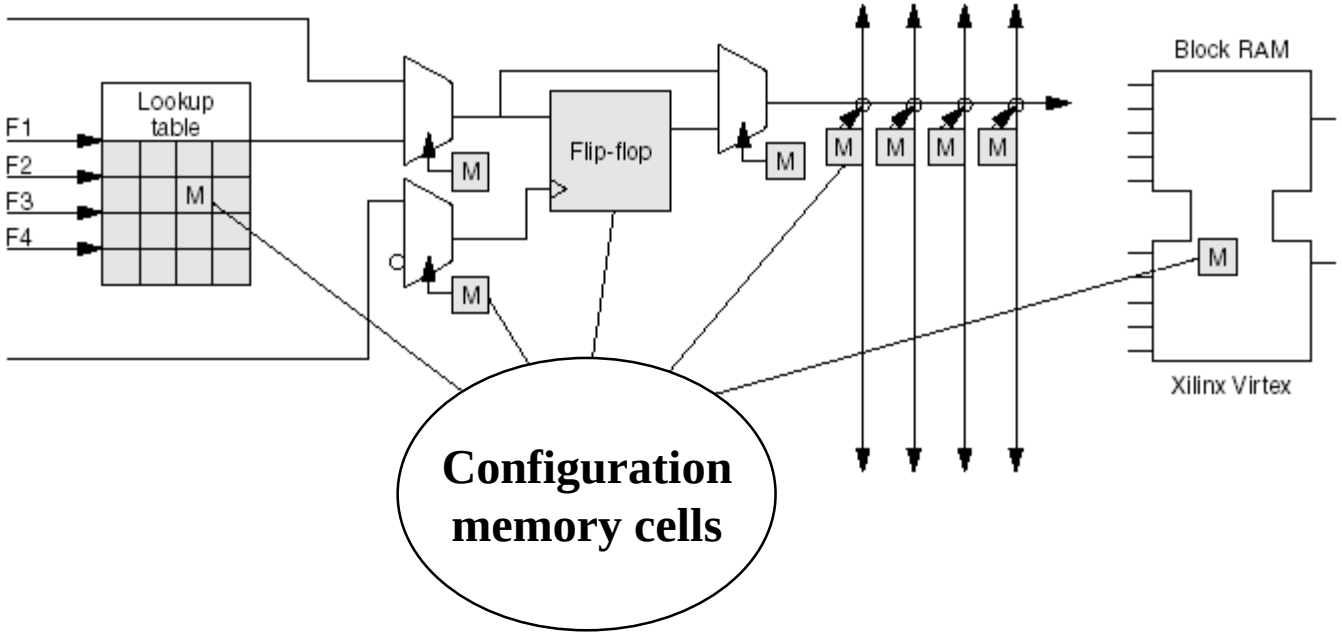
Configuration memory points

Generic organization of a SRAM-based FPGA



=> Types of tiles (CLB, BRAM, DLL, …)
=> User view (FFs in CLBs)
=> Configuration view (configuration memory)
=> May also include hardwired processors, multipliers, …

SRAM-based FPGA CLB: simplified view



From: F. Gusmao de Lima Kastensmidt, G. Neuberger, R. F. Hentschke, L. Carro, R. Reis
Designing fault tolerant techniques for SRAM-based FPGAs
IEEE Design & Test of Computers, vol. 21, no. 6, November-Décember 2004, pp. 552-562

Main characteristics of a device

- Digital, analog or mixte …
- In most cases: digital
  - Number of User I/Os
  - Number of flip-flops (+ memory blocks)
  - Array capacity (equivalent gates) and usage rate (depends on P&R and implemented function => influence of the basic architecture, based on sum of products or elemental cell)
  - Predictive evaluation and control of critical paths
  - Programmation type (retention, confidentiality, programmation speed, total or partial re-configurability)
- Security: SRAM-based is a concern
  - With respect to cloning (=> encrypted bitstreams)
  - With respect to configuration errors (modified function and/or structure)