SECT Security IN TELECOMMUNICATIONS

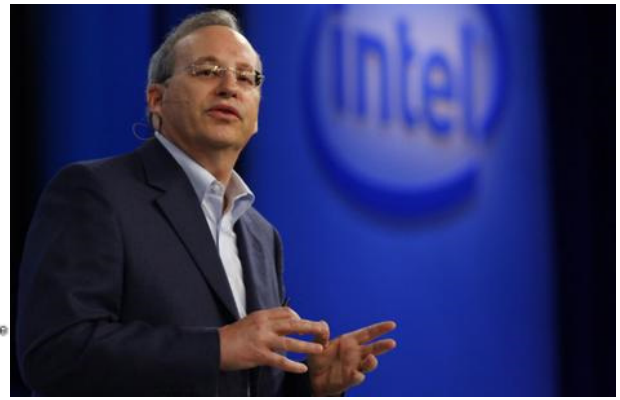
Trojan resilient ICs

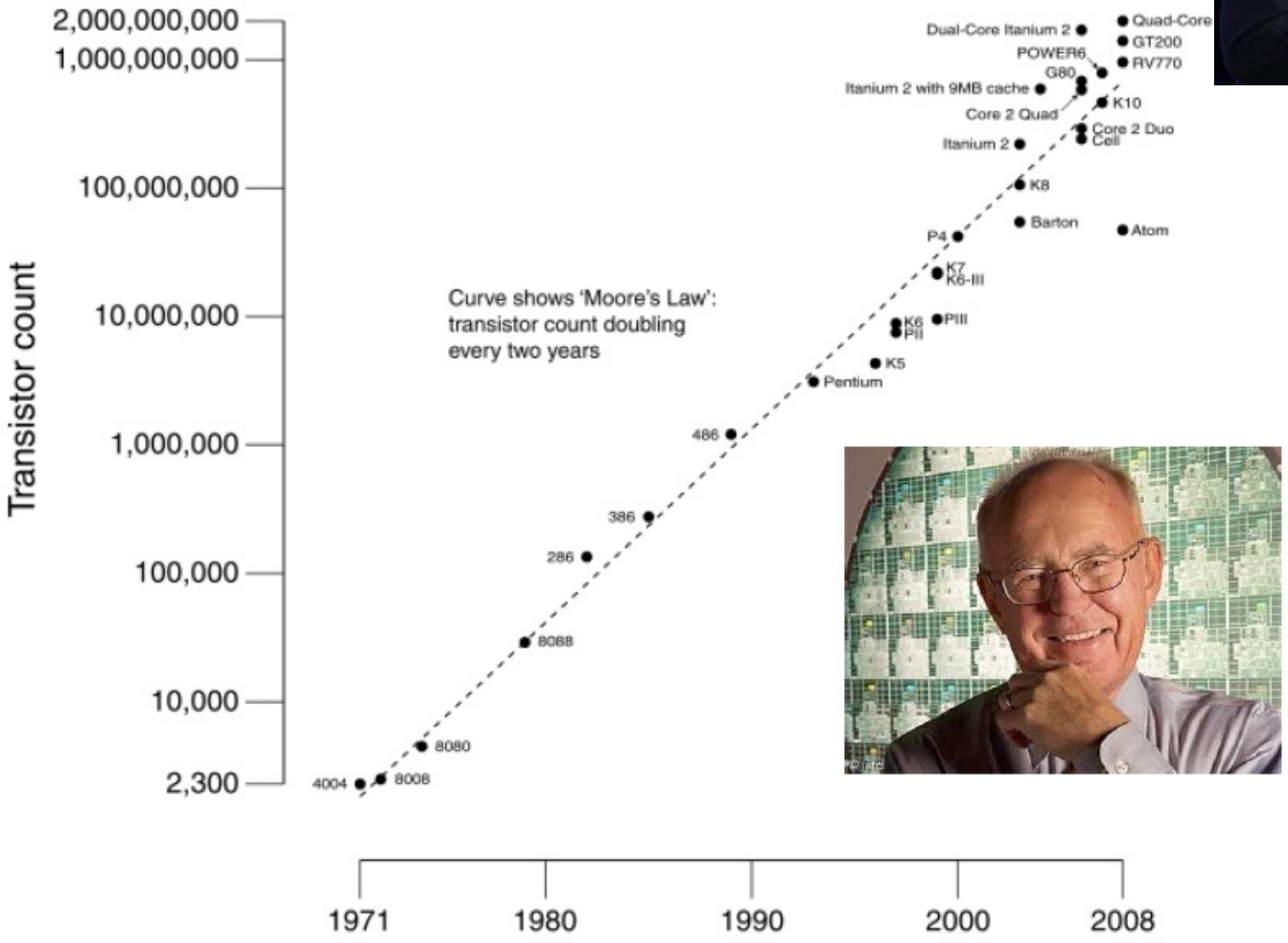
Christoph Bayer *&* Jean-Pierre Seifert

TU Berlin & Deutsche Telekom Laboratories, Berlin (Germany)

jpseifert@sec.t-labs.tu-berlin.de

“Transistors are free”

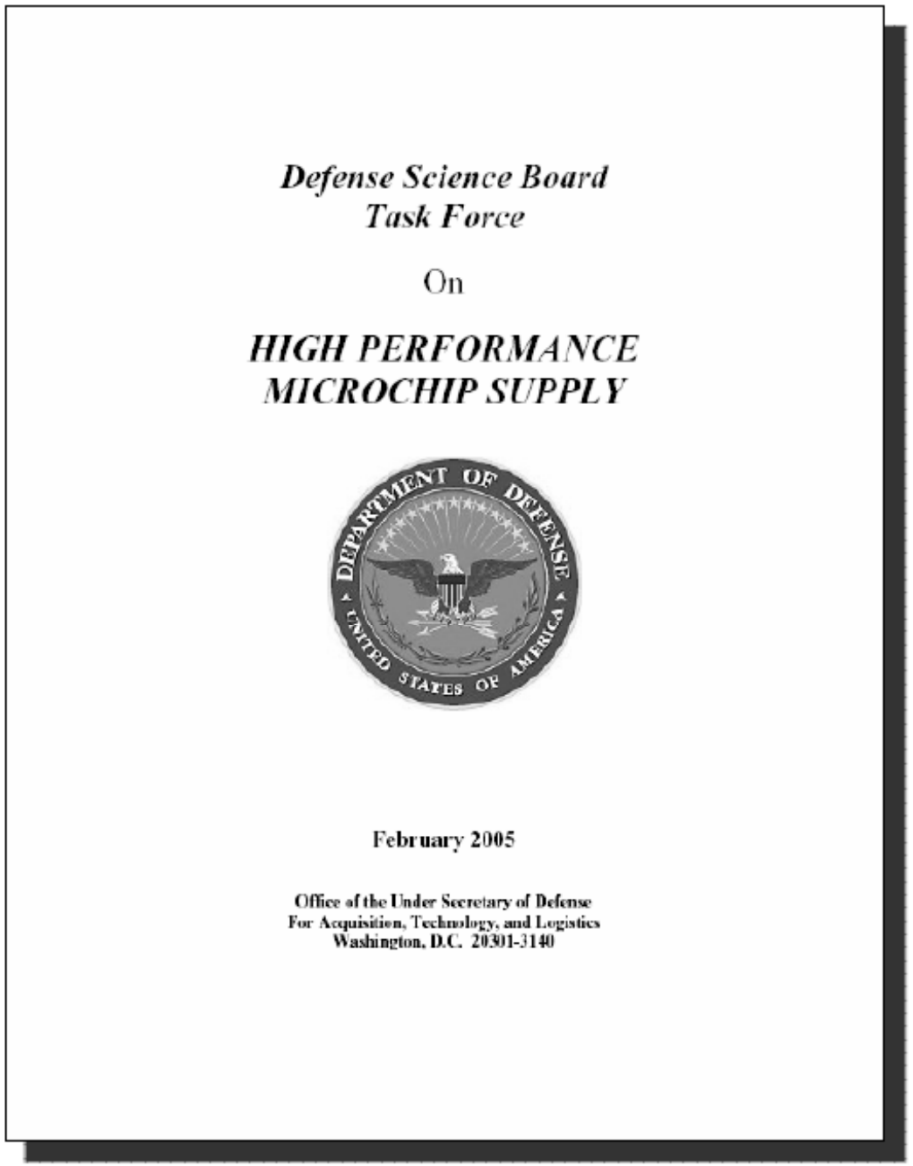




Agenda

1. Problem motivation
2. Trojan resilience
   - What is required?
   - Definitions?
3. A new idea to tackle the problem
4. Lots of definitions
5. A first provable result

Problem motivation



Defense Science Board
Task Force

On

HIGH PERFORMANCE MICROCHIP SUPPLY

February 2005

Office of the Under Secretary of Defense
For Acquisition, Technology, and Logistics
Washington, D.C. 20301-3140

http://www.acq.osd.mil/dsb/reports/2005-02-HPMS_Report_Final.pdf

The Economist

High-tech warfare

Something wrong with our **** chips today

Kill switches are changing the conduct and politics of war

Apr 7th 2011 | from the print edition

0

InformationWeek
BUSINESS INNOVATION POWERED BY TECHNOLOGY

Pentagon, Darpa Fear Enemies Could Tamper With Chips

The U.S. military's heavy dependence on overseas-made chips has got the government thinking about how to prevent tampering prior to delivery.

"The shift from United States to foreign IC manufacture endangers the security of classified information embedded in chip designs; additionally, it opens the possibility that 'Trojan horses' and other unauthorized design inclusions may appear in unclassified integrated circuits used in military applications," the board's report said. It added, "Neither extensive electrical testing nor reverse engineering is capable of reliably detecting compromised microelectronics components."

Among the concerns are that ICs could be doctored crudely in design or manufacture to fail early—for example, by changing chemical composition, by reducing material thicknesses or placing wires too close together. Alternatively, chips could be engineered to misbehave under more specialized circumstances with functional blocks serving as embedded "Trojan horses." That raises the prospect of weapon systems that could appear to be in perfect working order during tests or deployment but which could "switch off" in combat.

ieee spectrum INSIDE TECHNOLOGY

MAGAZIN

AEROSPACE BIOMEDICAL COMPUTING CONSUMER ELECTRONICS ENERGY

SEMICONDUCTORS // DESIGN

FEATURE

The Hunt for the Kill Switch

Are chip makers building electronic trapdoors in key military hardware? The Pentagon is making its biggest effort yet to find out

EMAIL PRINT SHARE PAGE 1 2 3 // VIEW ALL

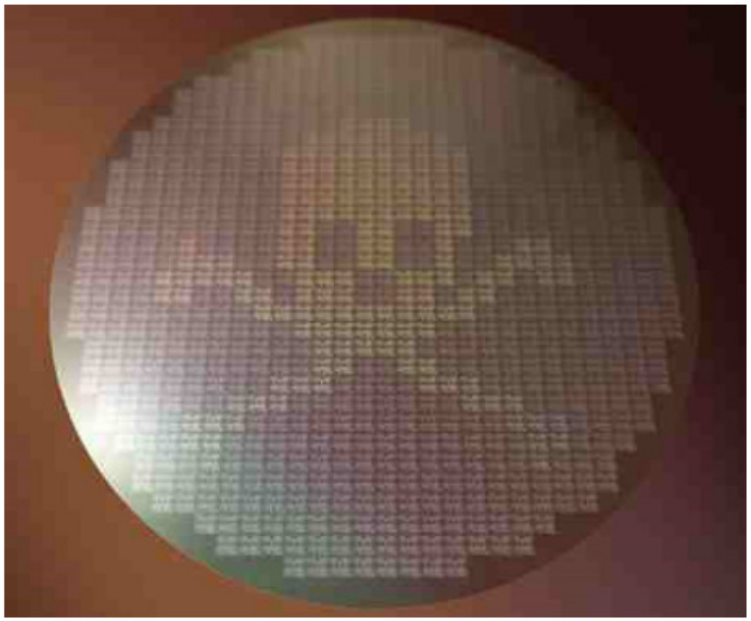
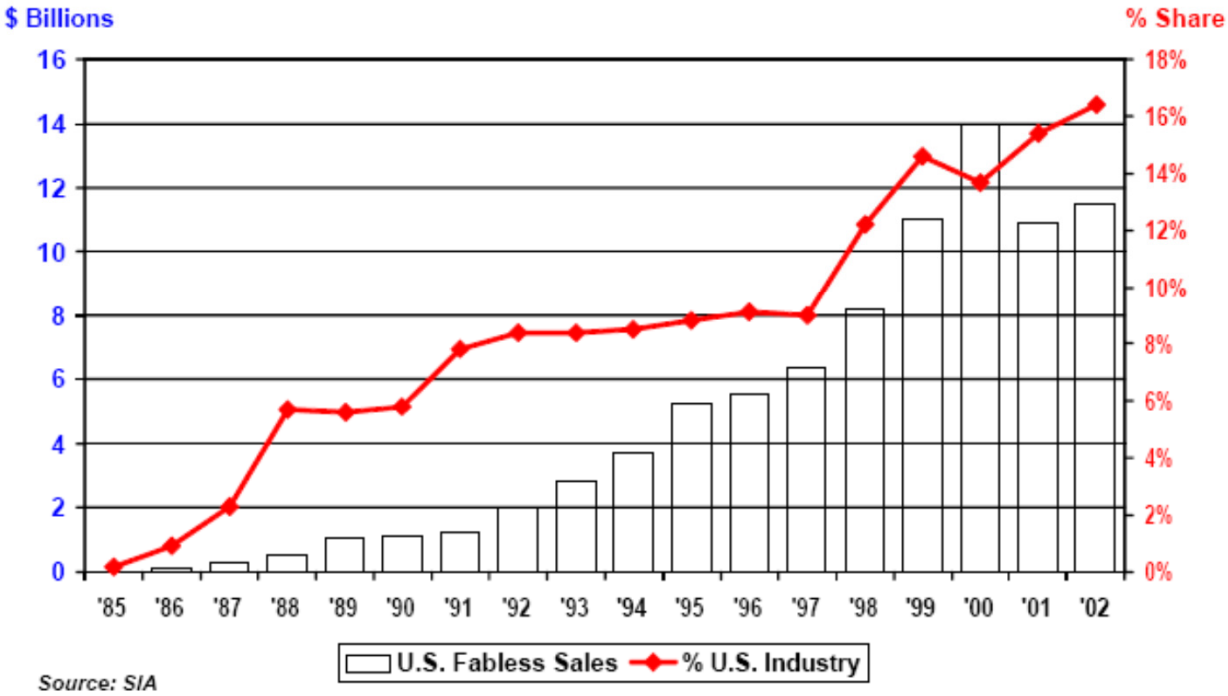


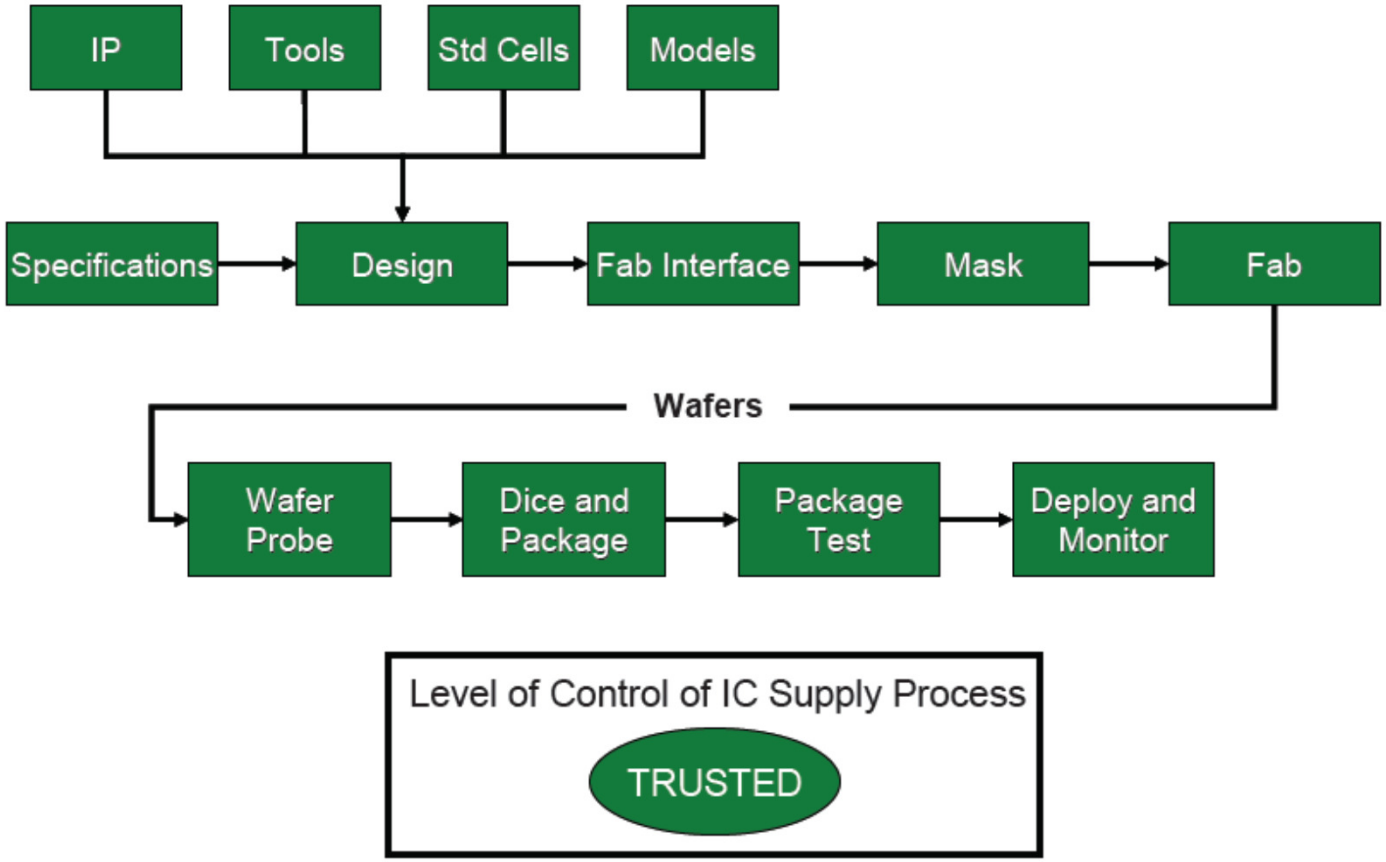
PHOTO:James Archer/AnatomyBlue

BY SALLY ADEE // MAY 2008

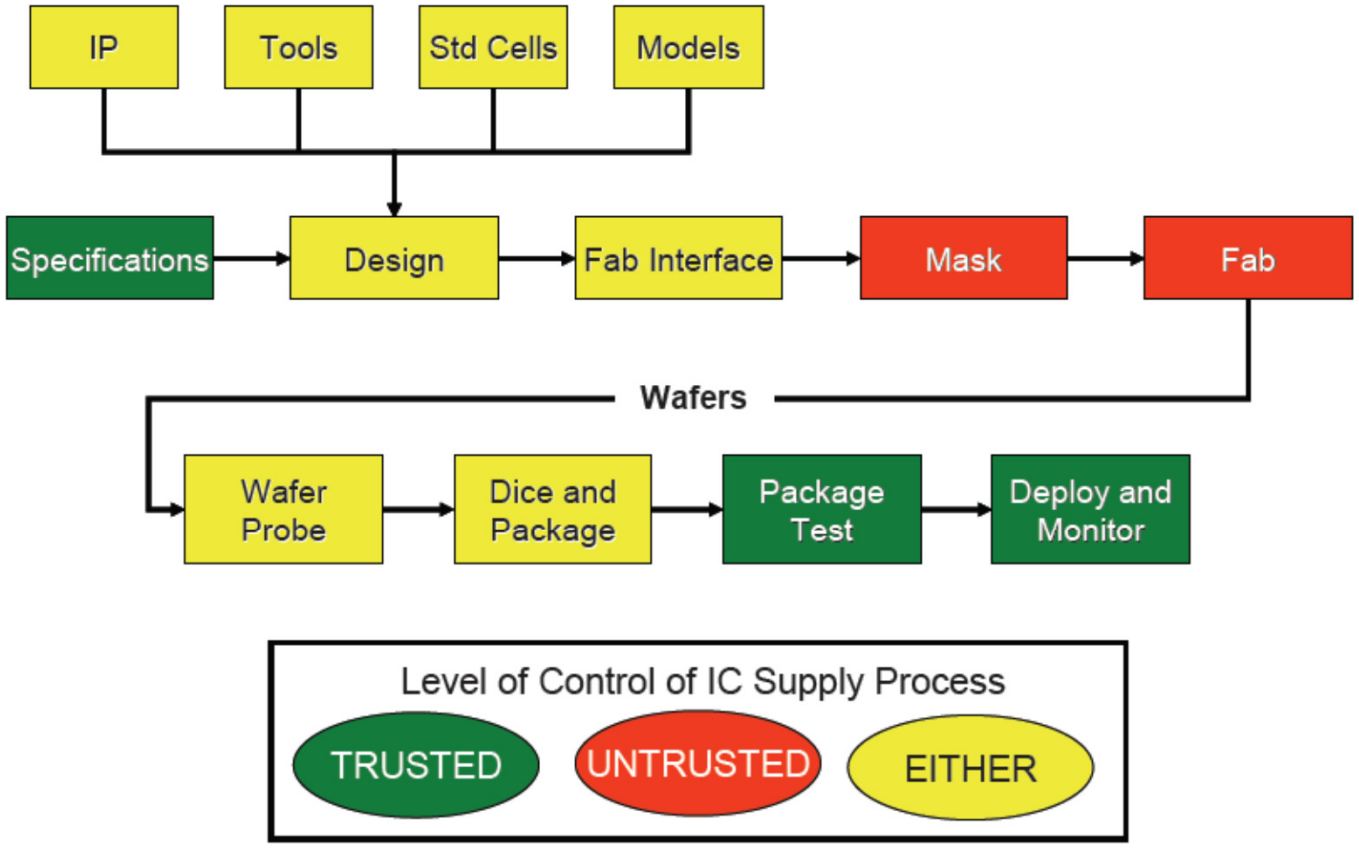
The fabless/foundry business model has grown to 16% of the U.S. chip industry. The trend is strongest in the leading process technology portion of the industry



Old Supply Chain Structure

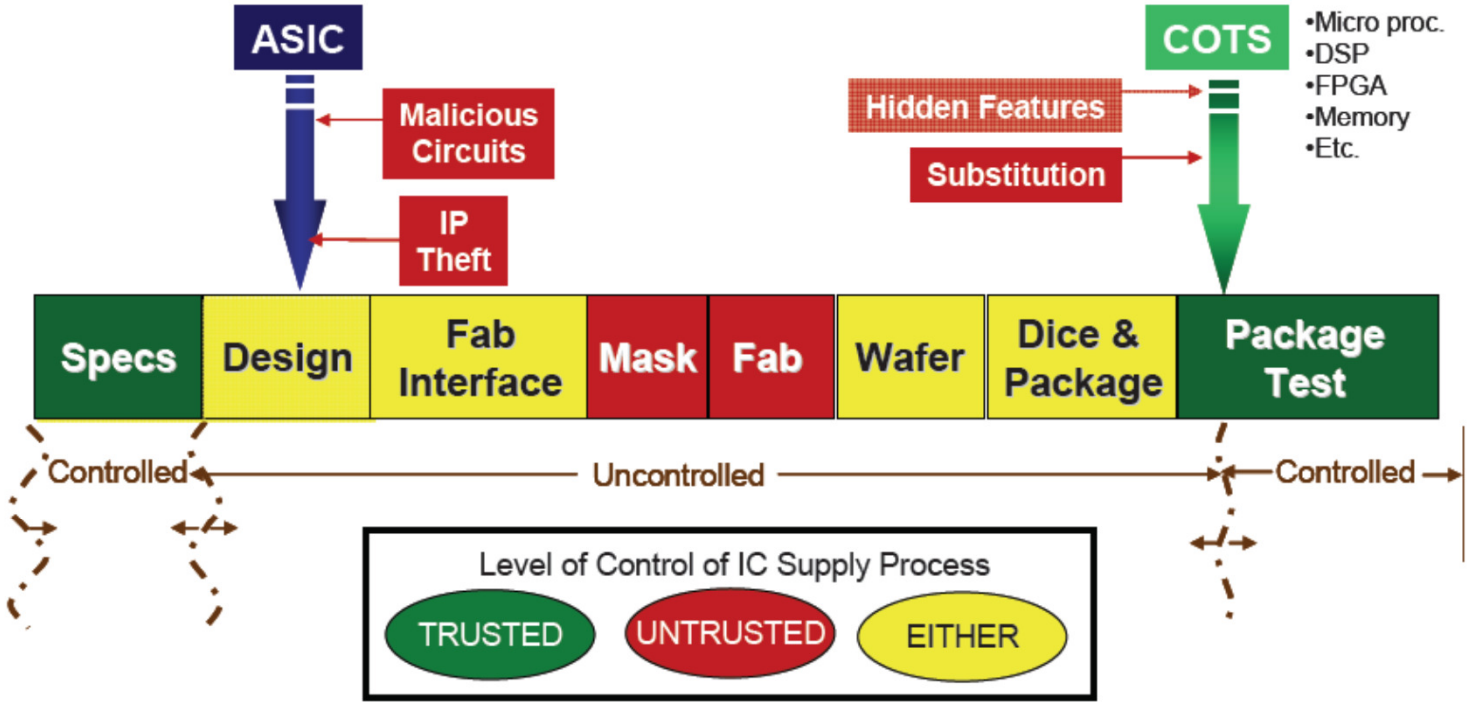


New Supply Chain Structure



APPROVED FOR PUBLIC RELEASE – Distribution Unlimited

Controlled and Uncontrolled Boundaries of the Chip Development Process



Malicous hardware?

Designing and implementing malicious hardware

Samuel T. King, Joseph Tucek, Anthony Cozzie, Chris Grier, Weihang Jiang, and Yuanyuan Zhou
University of Illinois at Urbana Champaign, Urbana, IL 61801

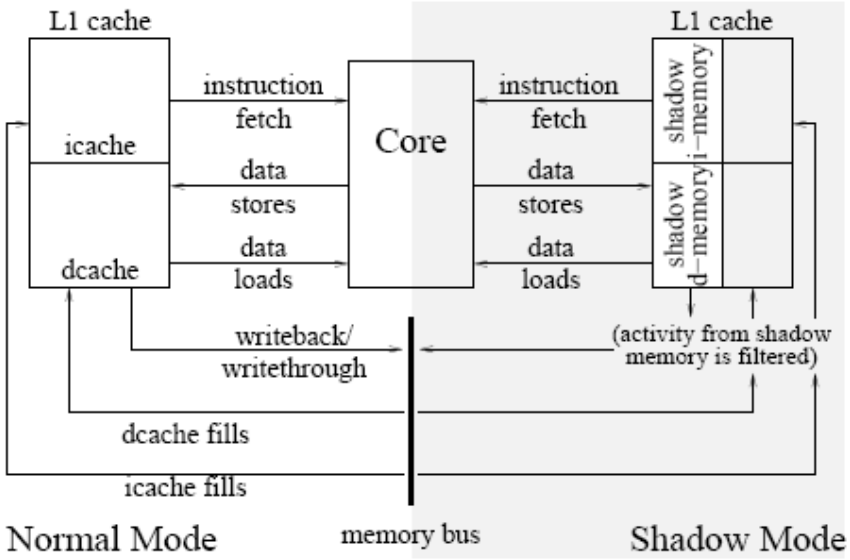


Figure 1: Hardware differences when shadow mode is active.

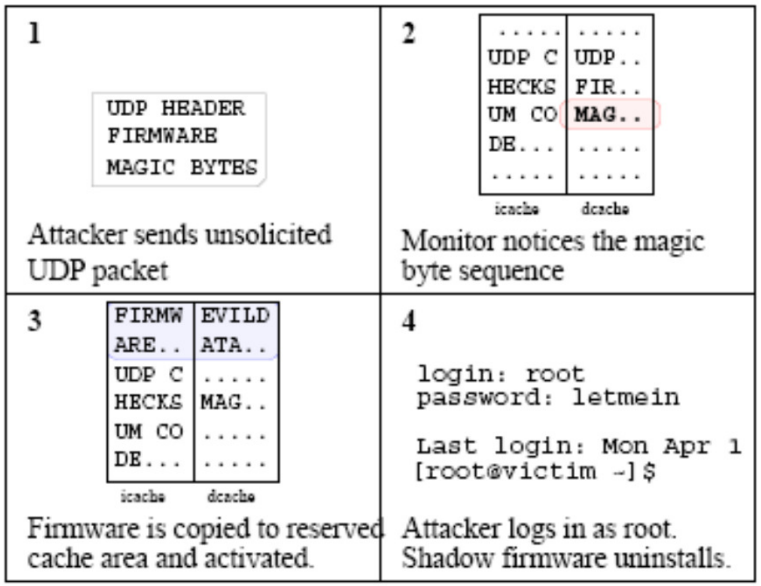


Figure 2: Overview of the login attack.

Malicous hardware?

Designing and implementing malicious hardware

Samuel T. King, Joseph Tucek, Anthony Cozzie, Chris Grier, Weihang Jiang, and Yuanyuan Zhou
University of Illinois at Urbana Champaign, Urbana, IL 61801

Processor	Logic gates	Lines of VHDL code
baseline CPU	1,787,958	11,195
CPU + memory access	1,788,917	11,263
CPU + shadow mode	1,789,299	11,312

Table 1: This table summarizes the circuit-level impact of our IMPs compared to a baseline (unmodified) Leon3 processor. We show the impact of an IMP that includes our memory access mechanism and an IMP that includes our shadow mode mechanism.

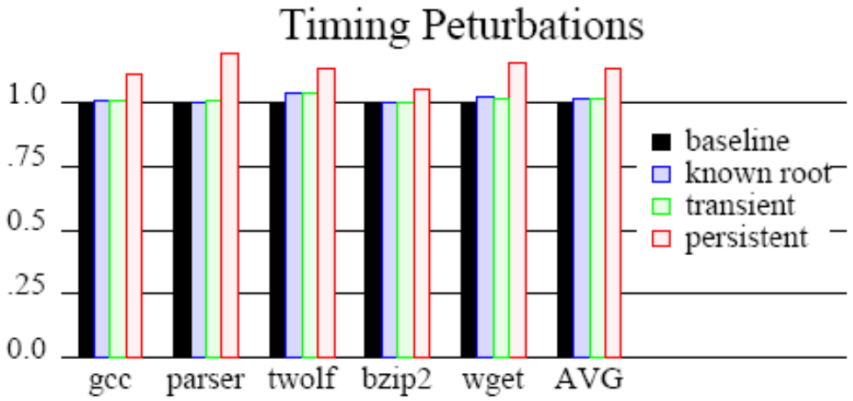
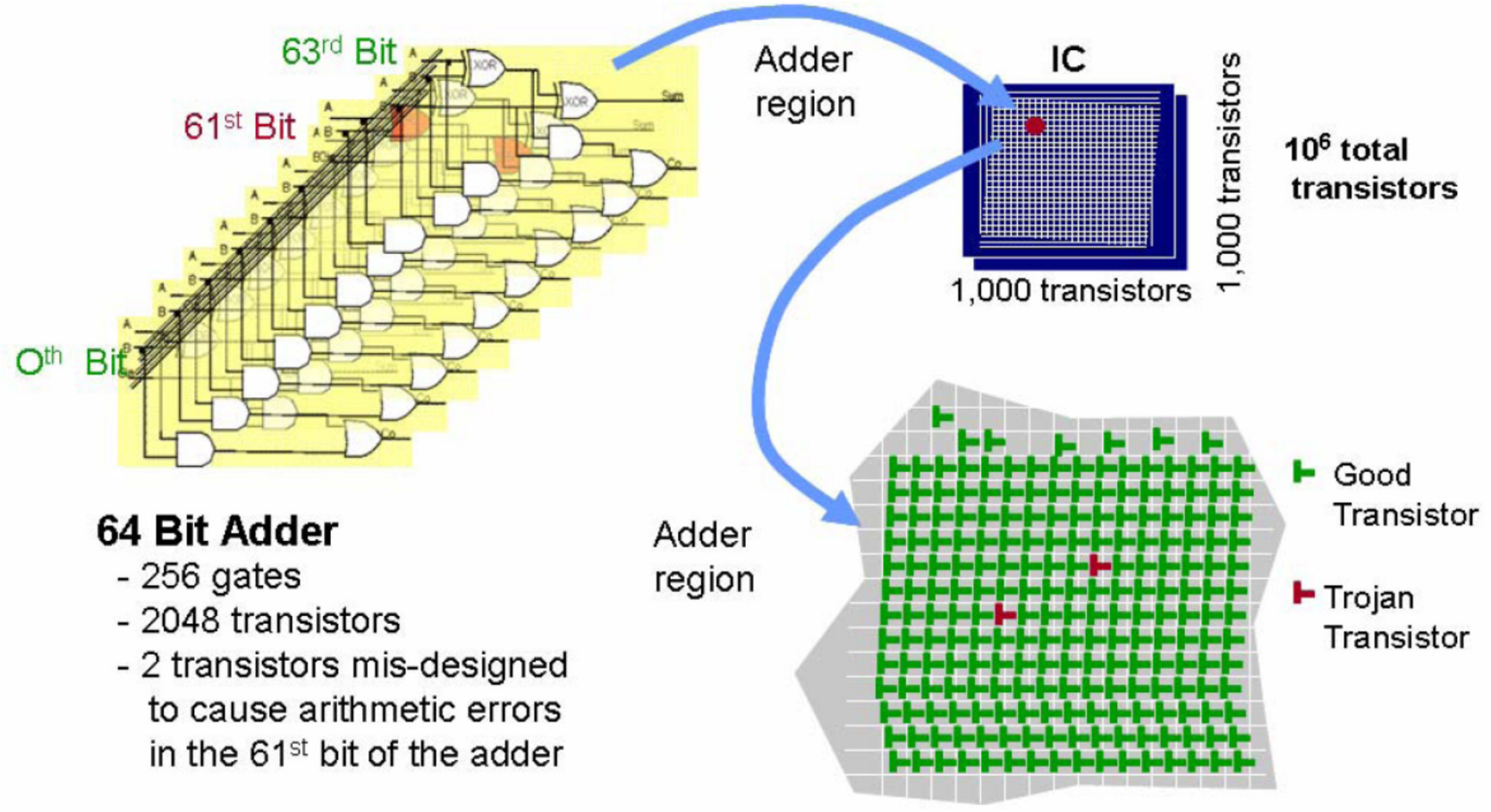


Figure 3: Time perturbations are measured relative to the baseline (non-attack) tests.

Areas of Interest

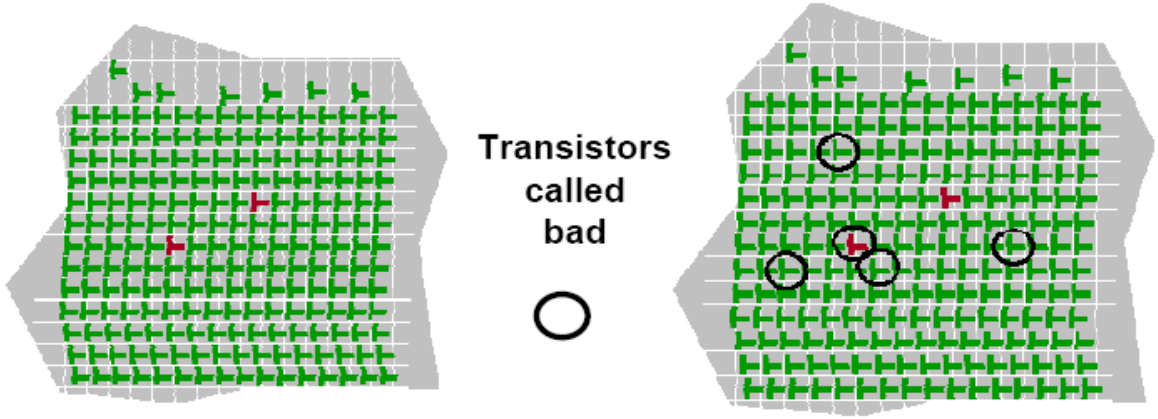
- **CASE1:** Given an IC corresponding to a known design, does the IC that is delivered do what it is supposed to do and nothing more? This is the case when the Fabrication facility is not trusted but the design process is. The problem is to determine whether the IC hardware received has been modified in order to determine that the fabrication can be trusted.
- **CASE2:** Given a specification and an IC design is the design true to the specification? In this case one assessing the trust of the design software and synthesis tools. The design itself must be validated.
- **CASE3:** Given a re-configurable IC, does the configurable data (bit stream) in the device accurately represent what was intended by the specification, design and VHDL synthesis?

What are the requirements?

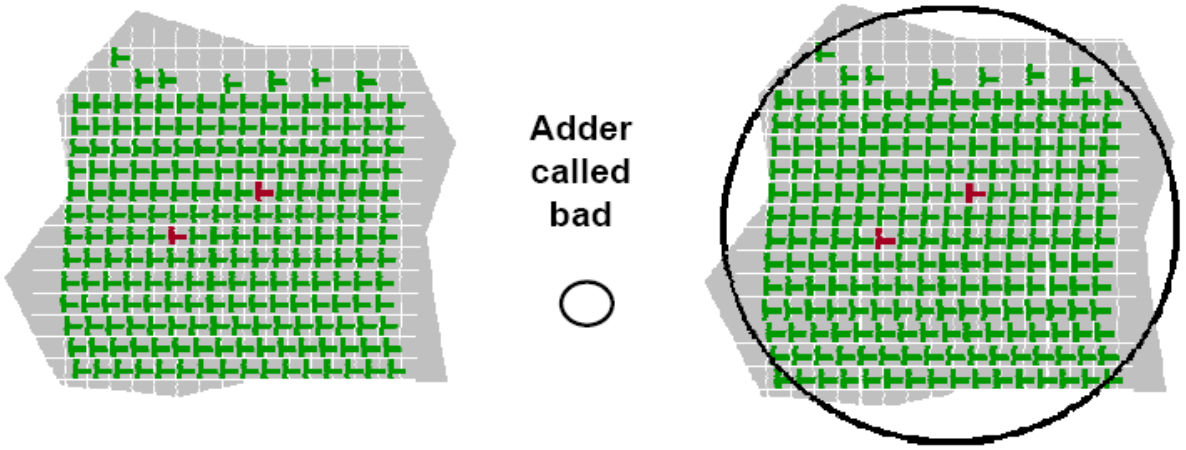


DARPA Metrics Challenge

Example 1 – Tests Performed at the Transistor Level



Example 2 – Tests Performed at the Functional Level



A new attempt to tackle the problem

- Idea:
  - Can we design ICs in such a way that no matter what the adversary (*in the untrusted fab*) does with our IC design, we can at least ensure equivalent functionality?
  - Tiny alterations don't change the IC functioning.
  - Huge alterations are detected are otherwise.

The good old world of a random adversary

- von Neumann (1956): The gates of a Boolean circuit fail independently with a probability bounded by a constant:

  There is a transformation that takes a circuit $\mathbf{C}$ into another circuit $\mathbf{C'}$ such that:

  1. The transformation takes polynomial time in the size of $\mathbf{C}$.
  2. The size of $\mathbf{C'}$ is $\mathbf{O(S(C)\ log\ (S))}$ where $\mathbf{S(C)}$ denotes the size of circuit $\mathbf{C}$.
  3. The depth of $\mathbf{C}$ is $\mathbf{O(D(C))}$ where $\mathbf{D(C)}$ is the depth of circuit $\mathbf{C}$.
  4. There exist $\boldsymbol{\varepsilon} > 0$ and $\mathbf{p < 1/2}$ such that if the gates of $\mathbf{C'}$ fail independently with probability bounded by $\boldsymbol{\varepsilon}$ then for every input $\mathbf{x}$ the probability that $\mathbf{C(x) \neq C'(x)}$ is at most $\mathbf{p}$.

But when the adversary is not random?

- If the adversary biases a few critical gates or the output bit then the circuit inadvertently outputs an incorrect value.

- So?

- We could inspect at least the last few transistors by hand, i.e., the output gate.

New model of adversary

- We would like to build *resilient circuits* that give the correct output even if at each level a small but maliciously choosen *constant fraction* of the gates are changed/malfunctioning.

- This can be thought of as using a constant number of absolutely reliable gates for the last few levels of the circuit.

Our Trojan model

Formal Attack Scenario Consider an integrated circuit IC with synchronous and combinational logic C. An adversary is allowed to choose at most a small constant fraction $\alpha < 1$ of the gates at each level of C to be faulty. Additionally, he is allowed to choose at most a fraction $\beta < 1$ of the wires between all levels of C to be faulty. That means the adversary may choose at most a $\gamma := \alpha + \beta$ fraction of the gates at each level to produce *incorrect outputs*. Observe that the last gate and the outgoing wire that reconnects the single output of C with the input of C via the clock must work correctly since the adversary is only allowed to destroy a $0 \leq \beta < 1$ fraction of this connection. The gates and wires of the last few levels are very likely to remain unchanged since alterations would be too obvious and detected during testing phase.

Our Trojan model

Definition 1. *We call an IC* infected by a hardware trojan *if an adversary has tampered with it for some* $\gamma > 0$ *according to our model. After the adversary tampered with an IC the way it is described for our hardware trojan, its combinational circuit* C *is called* γ-faulty.

Definition 2. *A circuit* C' *for a function* f *is said to be* γ-resilient *if* C' *computes the function* f *even if* C' *has been tampered by a hardware trojan for some* $\gamma > 0$. *Additionally, an IC that contains* C' *is also called* γ-resilient.

Coverage by our Trojan model

Insertion phase	Abstraction level	Activation mechanism	Effects	Location
specification	system level	always on	change the functionality	processor
design	development environment	triggered	downgrade performance	memory
fabrication	register-transfer level	▷ internally	leak information	I/O
testing	gate level	• time-based	denial of service	power supply
assembly and package	transistor level	• physical-condition-based		clock grid
	physical level	▷ externally		
		• user input		
		• component output		

Loose computation (Gal and Szegedy)

Definition:

For any computational device $\mathbf{M}$ we say that $\mathbf{M}$ δ-loosely computes $\mathbf{f}$ if

1. Whenever $\mathbf{f(x)=1}$ then $\mathbf{M(x)=1}$
2. If $\mathbf{f(z)=0}$ for every $\mathbf{z}$ with $\mathbf{d(x, z) \leq \delta * n}$ then $\mathbf{M(x)=0}$.

Here $\mathbf{d(x, z)}$ denotes the Hamming distance between $\mathbf{x}$ and $\mathbf{z}$, and $\mathbf{n}$ the number of input bits to $\mathbf{f}$.

$\mathbf{M}$ can output an arbitrary value or no value at all if input $\mathbf{x}$ does not belong to the above two categories.

Another definition

Definition:

- For an error correcting code $\mathbf{E_n}$ with codewords of length $\mathbf{q_n}$ and for a function $\mathbf{f}$ we define

  $\mathbf{f \bullet E_n : \{0,1\}^{qn} \rightarrow \{0,1\}}$ as follows

  1. $\mathbf{(f \bullet E_n)(z)=0}$ for all $\mathbf{z}$ where $\mathbf{z}$ is not a codeword of the code $\mathbf{E_n}$
  2. If $\mathbf{z=E_n(x)}$ then $\mathbf{(f \bullet E_n)(z) = f(x)}$

Proposition:

- If the Hamming distance of any two codewords in $\mathbf{E_n}$ is at least $\mathbf{q_n}$ and $\mathbf{M}$ is a computational device that computes $\mathbf{f \bullet E_n}$ in a δ-loose manner then

$$\mathbf{M(E_n(x)) = f(x)}$$

on any input $\mathbf{x}$.

It is known from coding theory that there exist linear binary codes $\mathbf{E_n}$ with the following properties:

- The matrix of $\mathbf{E_n}$ can be polynomially computed in $\mathbf{n}$.
- This also means that the length $\mathbf{q_n}$ of the codewords is also polynomial in n.
- The Hamming distance of any two codewords in $\mathbf{E_n}$ is at least $\boldsymbol{\delta * q_n}$ for some small constant $\boldsymbol{\delta > 0}$.

Main result

Theorem 4. *Let C be a Boolean circuit and let $f : \{0,1\}^n \to \{0,1\}$ be the corresponding Boolean function computed by C. There exists a code $E = E_C$ and a circuit C' such that C' computes $f \circ E$ in a δ-loose manner for every $\delta > 0$ even if an adversary destroys an α fraction of the gates at each level of C' and as well a β fraction of the wires between all levels of C'. Moreover, E and C' have the following properties:*

1. *$|E(x)| \leq q(|x|)$ for some polynomial q independent of C.*
2. *The Hamming distance $d(x, y)$ between any two codewords x and y of E is at least $\delta_0 |E|$ for some $0 < \delta_0 < 1$ independent of C (δ_0 is a function of δ).*
3. *$D(C') \leq O(\log S(C))$. This implies that $S(C')$ is polynomial in $S(C)$.*
4. *C' can be computed from C in probabilistic polynomial time and $E(x)$ can be computed from C and x in polynomial time.*

Thus

- We can design ICs in such a way that no matter what the adversary (*in the untrusted fab*) does with our IC design, we can at least ensure equivalent functionality,
  - provided that the adversary only changes a constant fraction of gates per circuit layer.

Questions?

Thank you!