Week 13: Lecture B

Hardware Testing

Wednesday, April 9, 2025

How are semester projects going?

Making progress?

Stuck?

The Next Few Weeks

Part 3: New Frontiers in Fuzzing

Monday Meeting	Wednesday Meeting
Mar. 31 **Kernel Fuzzing** ▶ Readings:	Apr. 02 **LLM-assisted Fuzzing** ▶ Readings:
Apr. 07 **Compiler Fuzzing** ▶ Readings:	Apr. 09 **Hardware Fuzzing** ▶ Readings:
Apr. 14 **Fuzzing Configurable Software** ▶ Readings:	Apr. 16 ⚠ **Final Presentations (Day 1)**
Apr. 21 ⚠ **Final Presentations (Day 2)** ⚠ **Final Reports due Tuesday by 11:59pm via Canvas**	Apr. 23 **No Class (Reading Day)**

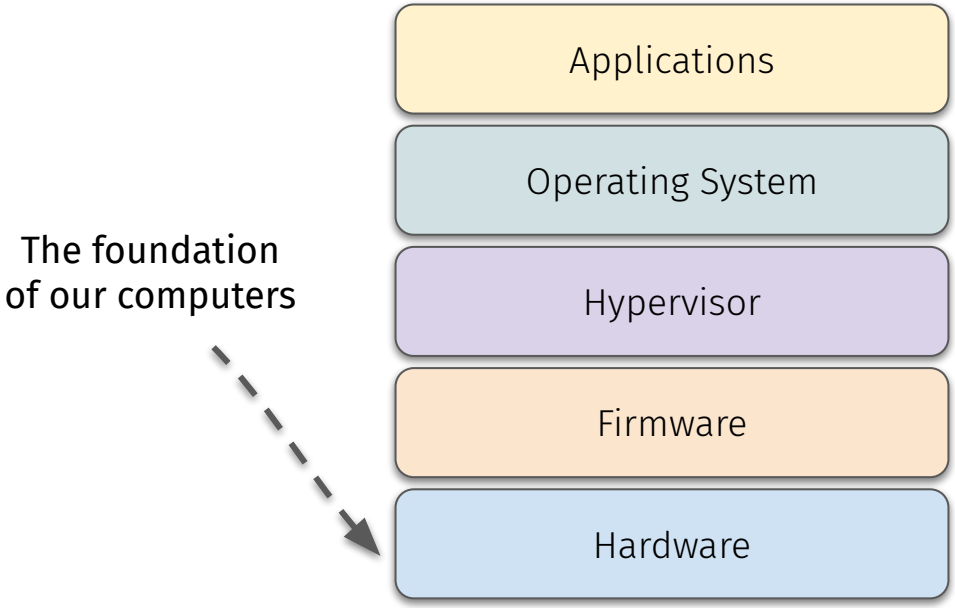
Recap: Project Schedule

- **Apr. 16th & 21st:** final presentations
  - **5–8 minute** slide deck and discussion
  - What you did, and why, and what results
  - **Report any bugs found** (and show you did so!)
- What's most important:
  - High-level technique
  - Challenges and workarounds
  - Key results (bugs found, other successes, etc.)
- **Project report** due by midnight last day of class
  - 3–5 pages describing your work and results
  - Reports of any bugs found

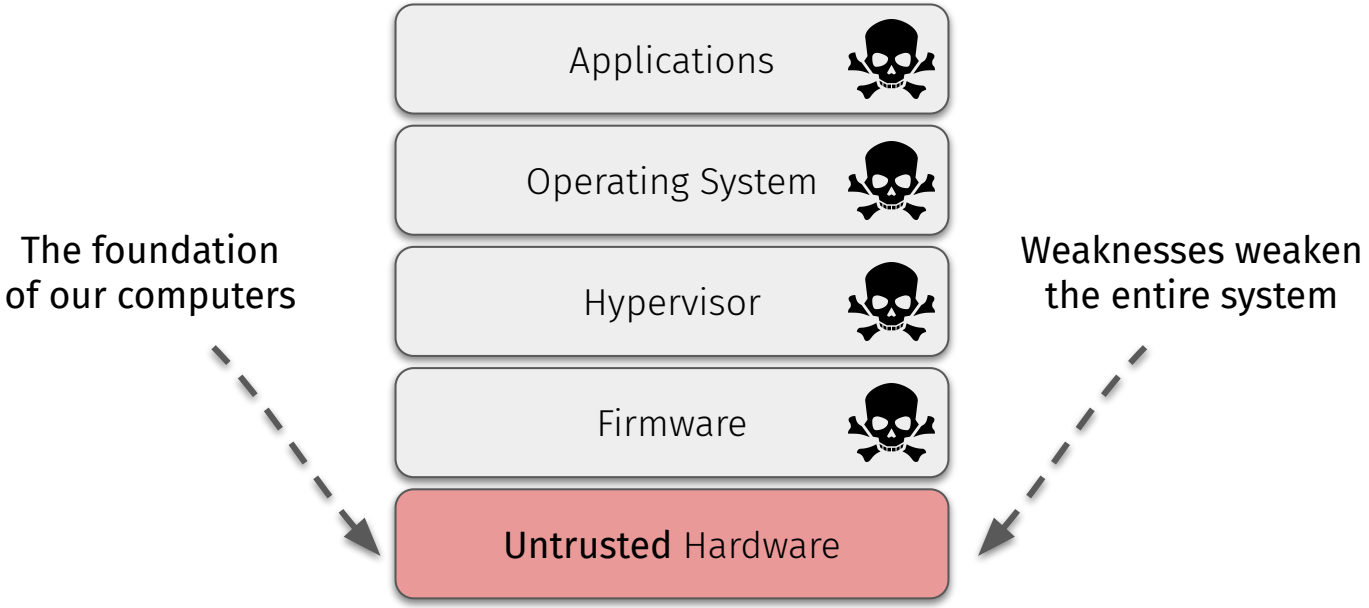
Questions?

Hardware Security and Testing

Hardware

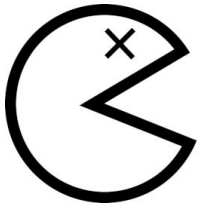

Applications
Operating System
Hypervisor
Firmware
Hardware
The foundation
of our computers

Hardware

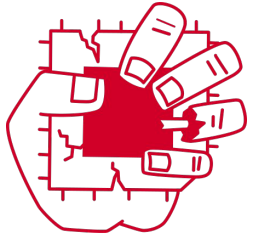

Applications
Operating System
Hypervisor
Firmware
Untrusted Hardware
The foundation of our computers
Weaknesses weaken the entire system

Hardware

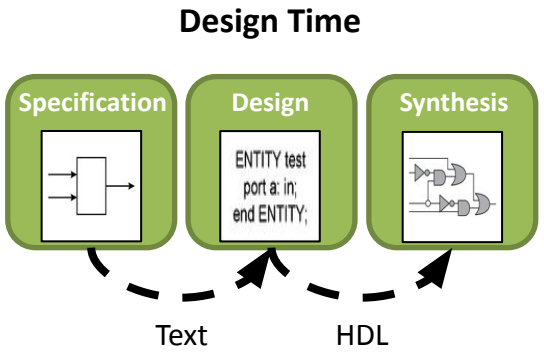
The foundation
of our computers



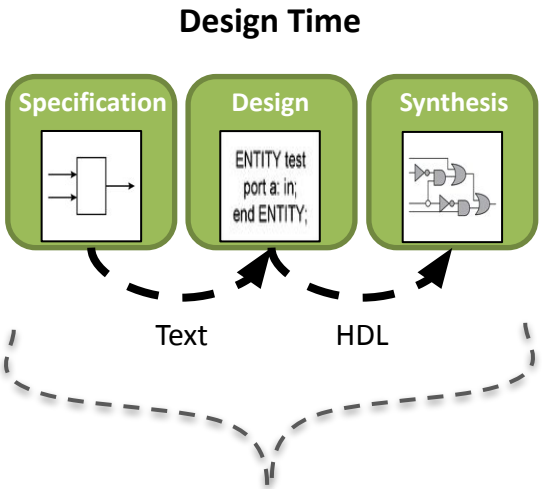
Weaknesses weaken
the entire system



Creating Hardware

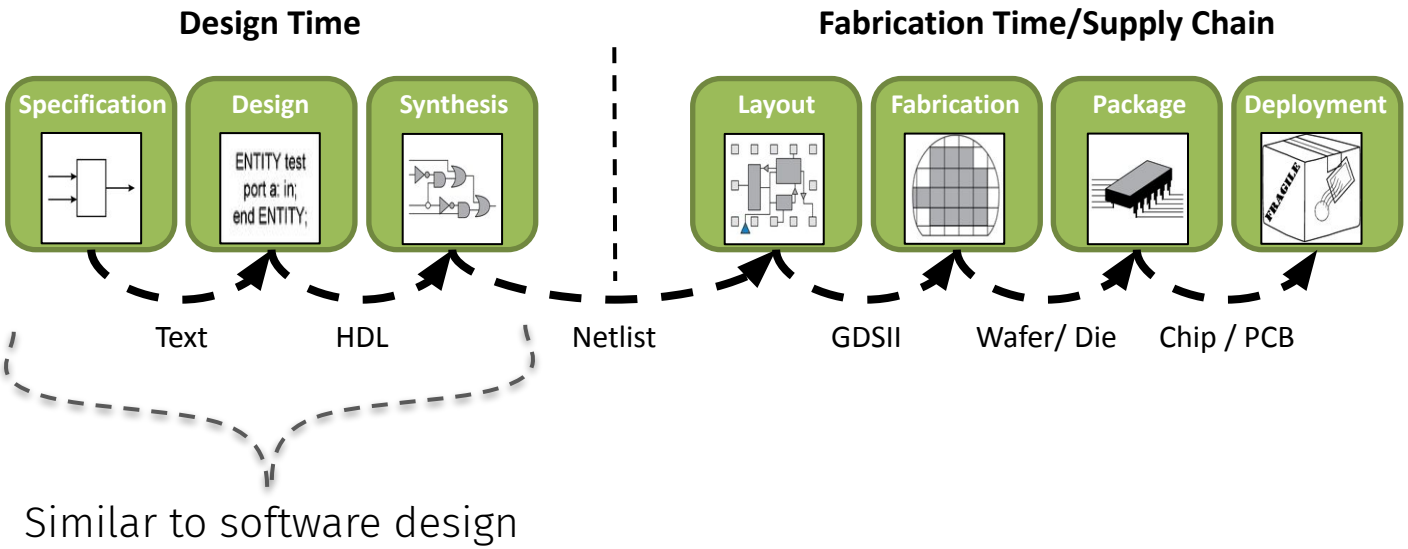


Creating Hardware

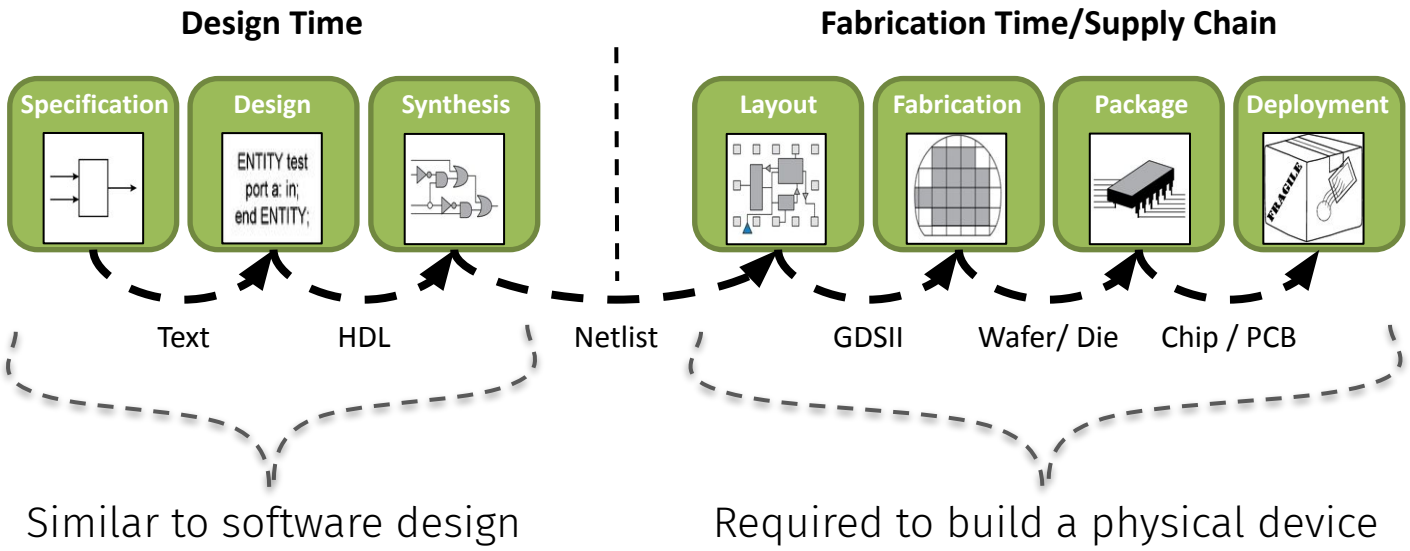


Similar to software design

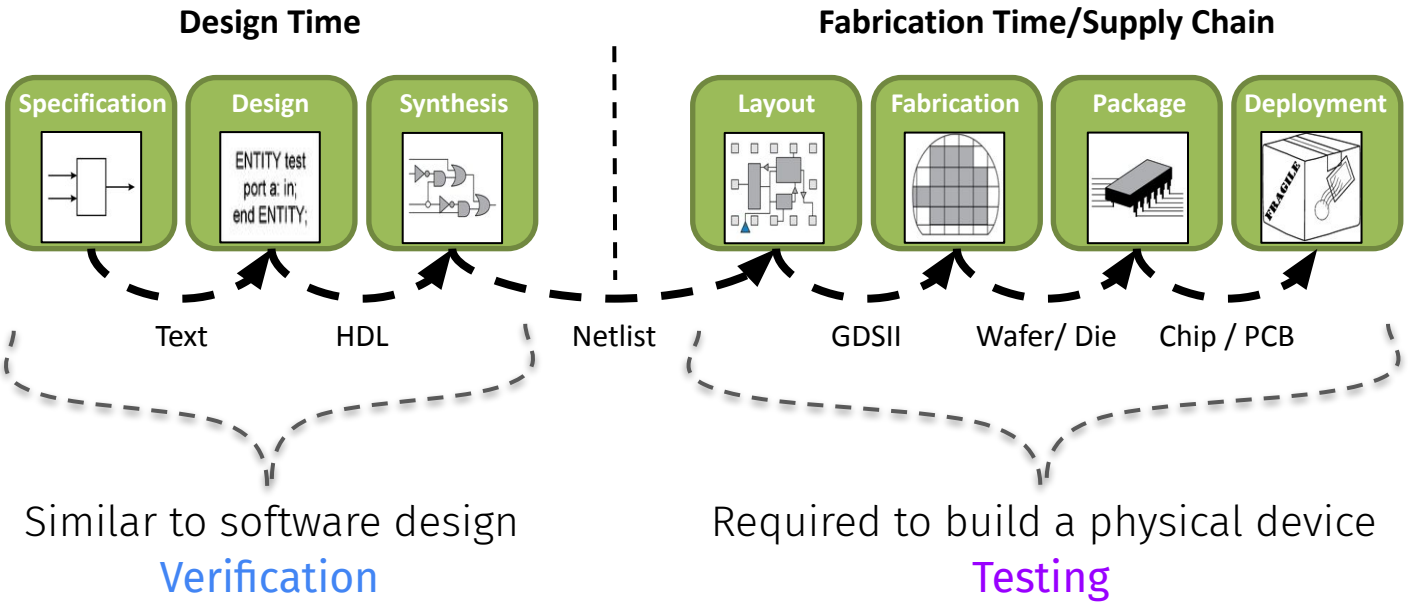
Creating Hardware



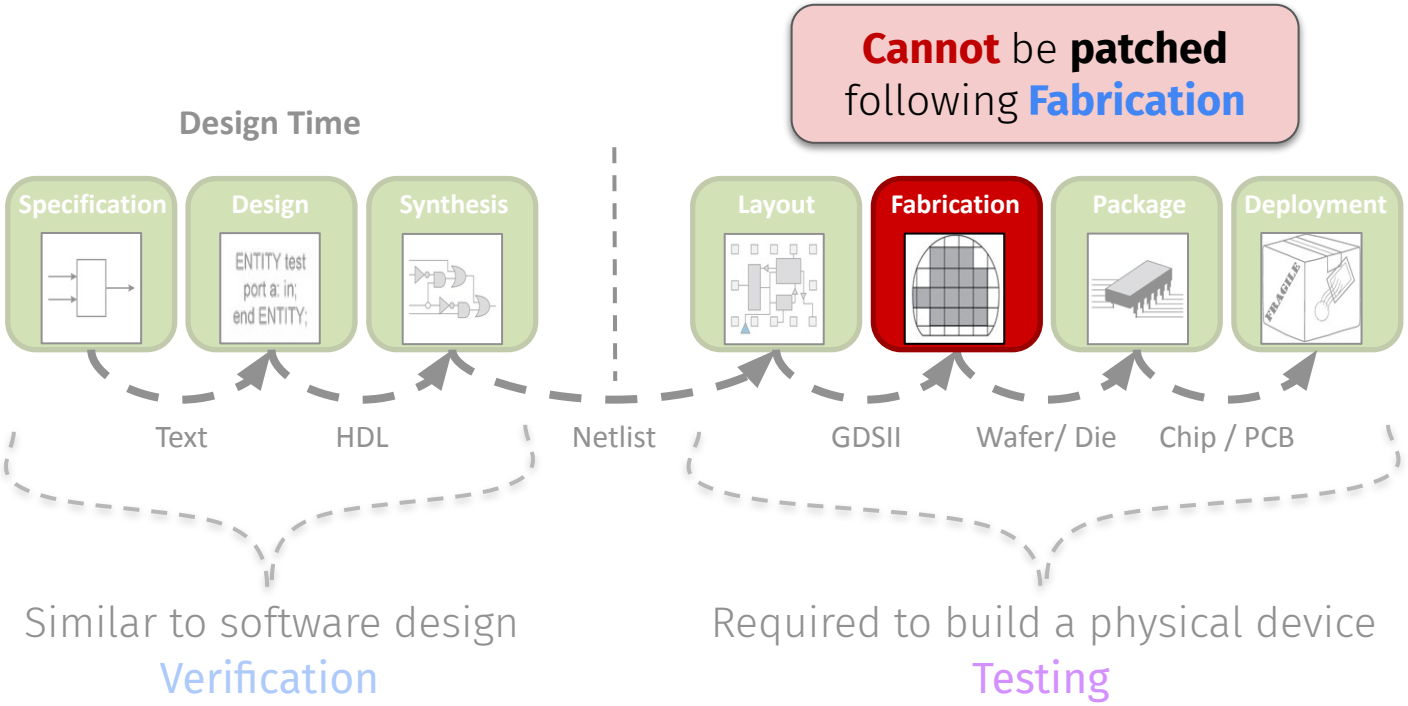
Creating Hardware



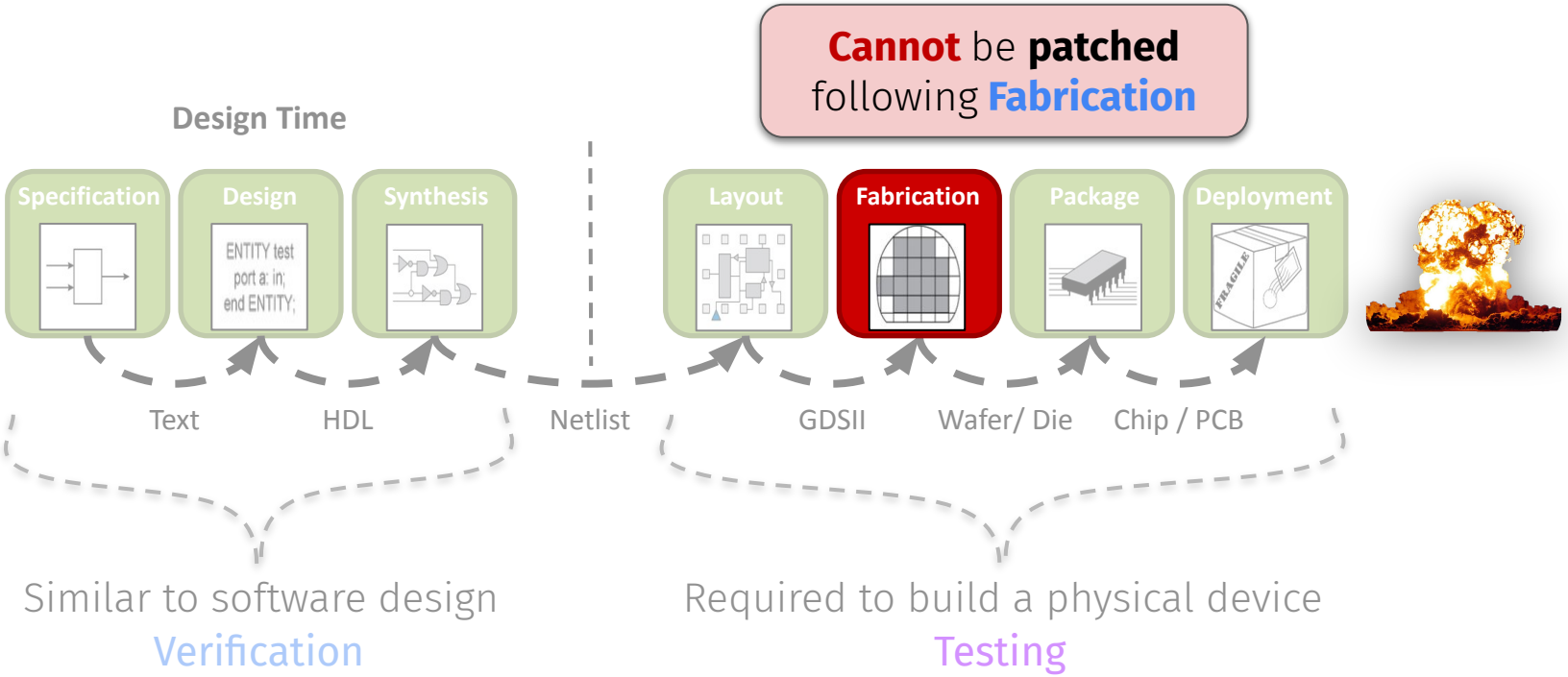
Creating Hardware



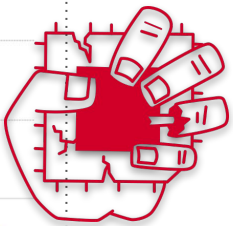
Hardware Bugs



Hardware Bugs



Hardware Bugs

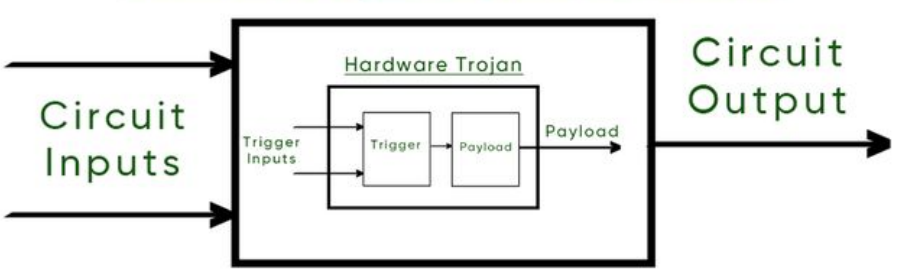


Hardware Trojans

- **Trojan Horse:**
  - **???**

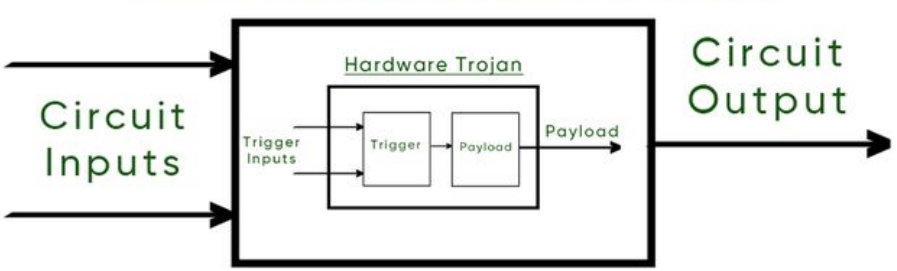
Hardware Trojans

- **Trojan Horse:**
  - Attack pre-inserted into chip
  - Will be **exploited** at **run time**
  - **Remotely triggered** by attacker



Hardware Trojans

- **Trojan Horse:**
  - Attack pre-inserted into chip
  - Will be **exploited** at **run time**
  - **Remotely triggered** by attacker
- **Ideal characteristics:**
  - Small
  - Stealthy
  - Controllable



Hardware Trojans

- **Trojan Horse:**
  - Attack pre-inserted into chip
  - Will be **exploited** at **run time**
  - **Remotely triggered** by attacker
- **Ideal characteristics:**
  - Small
  - Stealthy
  - Controllable
- **Engineering a trigger**

```
1  void attack_signed_c() {
2      volatile int a, b, c = 0;
3
4      while(1) {
5          int c1 = c;
6          int b1 = b;
7
8          int i1 = ((b1 / c1) + 1);
9          int i2 = ((i1 / c1) + 1);
10         int i3 = ((i2 / c1) + 1);
11         int i4 = ((i3 / c1) + 1);
12         int i5 = ((i4 / c1) + 1);
13         int i6 = ((i5 / c1) + 1);
14         int i7 = ((i6 / c1) + 1);
15         int i8 = ((i7 / c1) + 1);
16         int i9 = ((i8 / c1) + 1);
17
18         a = ((i9 / c1) + 1);
19     }
20 }
```

Division sets div-by-zero flag

Addition resets div-by-zero flag

Software state will affect **analog state**!

Hardware Trojans

Israeli sky-hack switched off Syrian radars countrywide

Backdoors penetrated without violence

Lewis Page

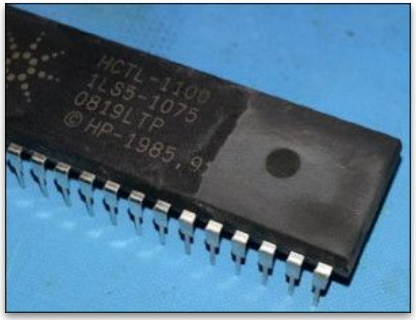
Thu 22 Nov 2007 // 13:57 UTC

More rumours are starting to leak out regarding the mysterious Israeli air raid against Syria in September. It is now suggested that "computer to computer" techniques and "air-to-ground network penetration" took place.

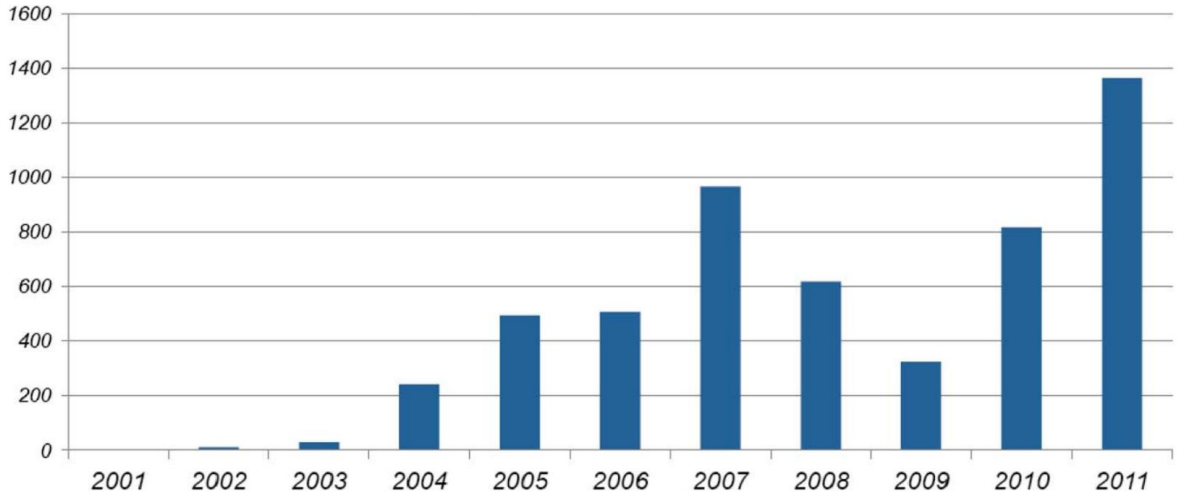
The latest revelations are made by well-connected *Aviation Week* journalists. Electronic-warfare correspondent David Fulghum says that US intelligence and military personnel "provided advice" to the Israelis regarding methods of breaking into the Syrian air-defence network.

Recycled and Counterfeit Hardware

Guin *et al.*: Counterfeit Integrated Circuits: A Rising Threat in the Global Semiconductor Supply Chain

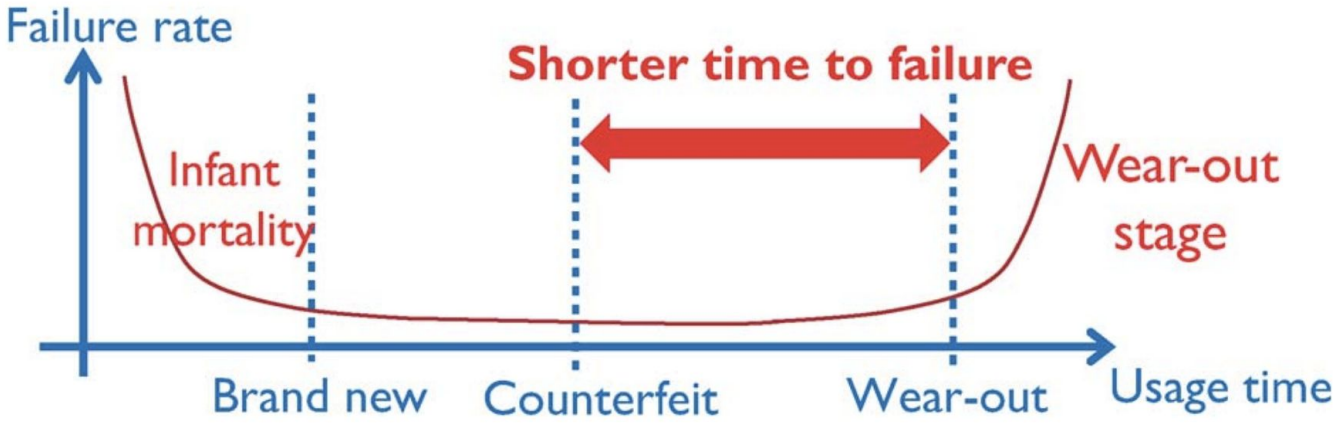


Russia is resorting to putting computer chips from dishwashers and refrigerators in tanks due to US sanctions, official says



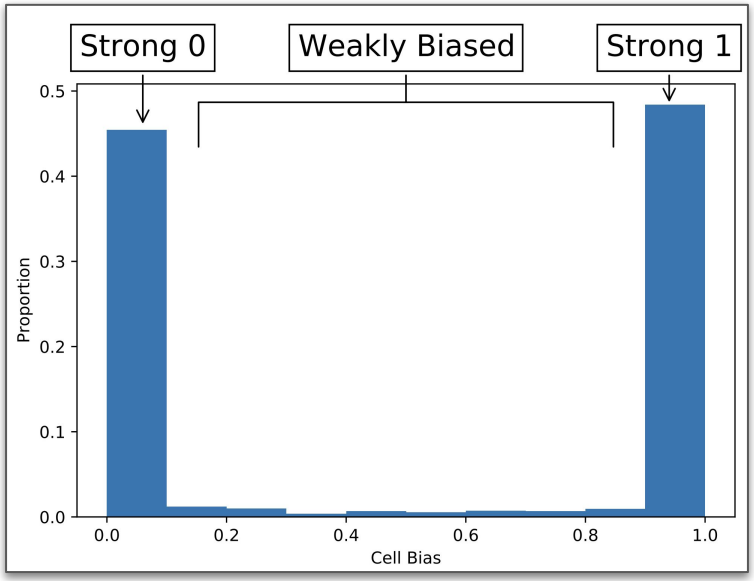
Recycled and Counterfeit Hardware

- **Counterfeit** and **recycled chips** have a **shorter lifespan**
  - Absolutely dangerous for security-critical use cases



Recycled and Counterfeit Hardware

- **Counterfeit** and **recycled chips** have a **shorter lifespan**
  - Absolutely dangerous for security-critical use cases

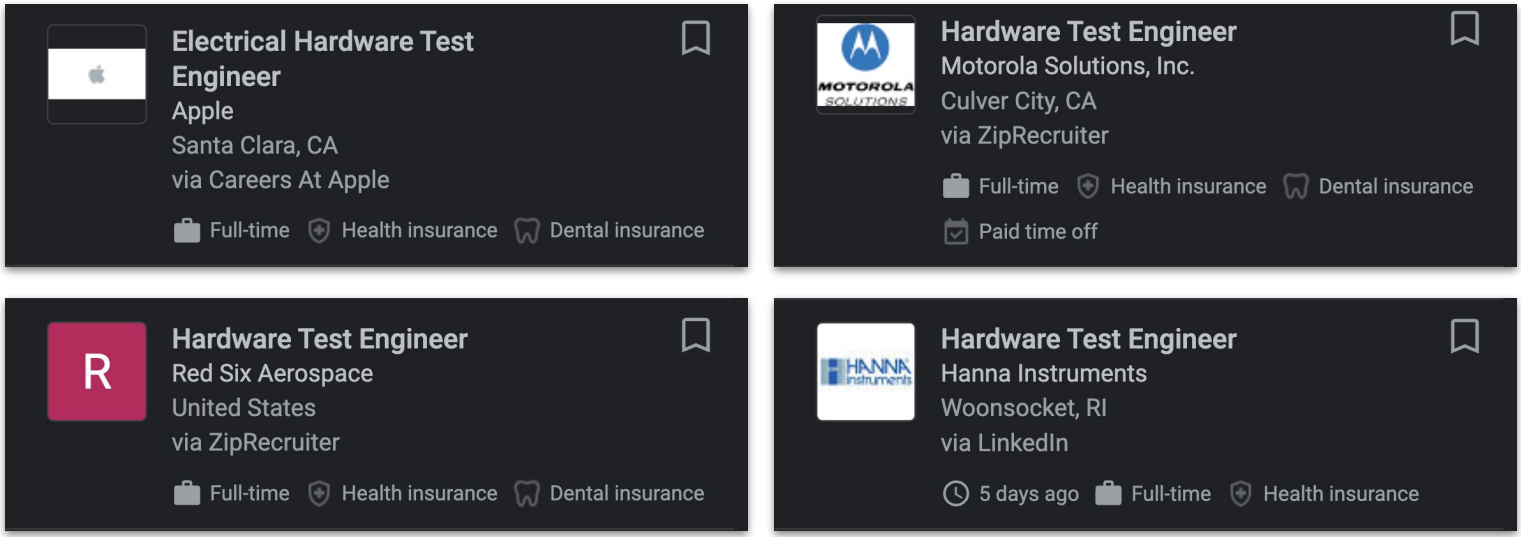


Secure Hardware

- **Can we ever know for sure that a chip is secure?**

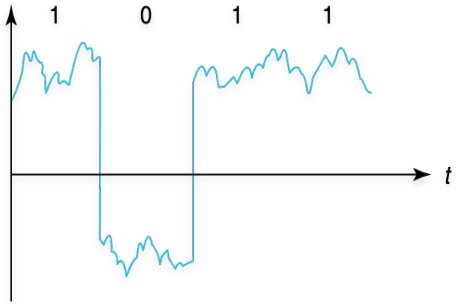
Hardware Testing

- One of the highest-paid (and steep-learning-curve) careers in testing
  - **Spoiler:** it's even harder than testing software

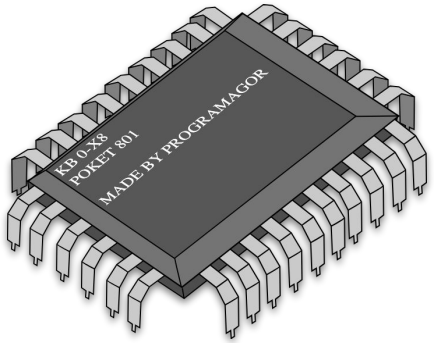


Testing Hardware Physically

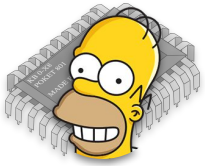
- **How could we even do this?**



Signal Generator

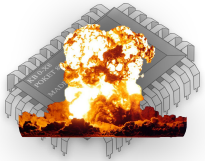


Hardware Chip



Success

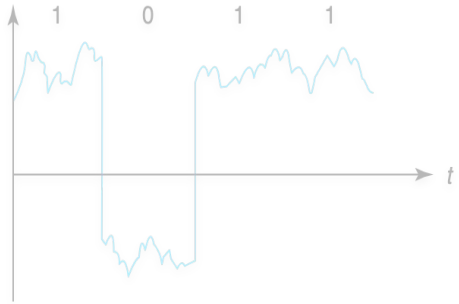
?



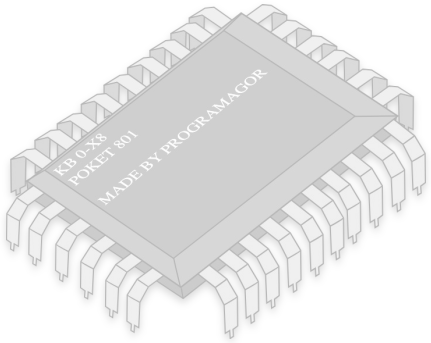
Failure

Testing Hardware Physically

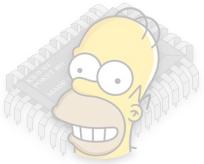
- **How could we even do this?**
  - **Downsides?**



Signal Generator



Hardware Chip



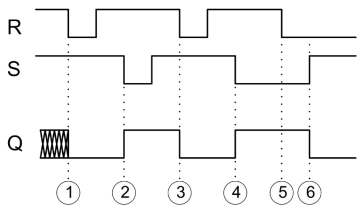
Success

?

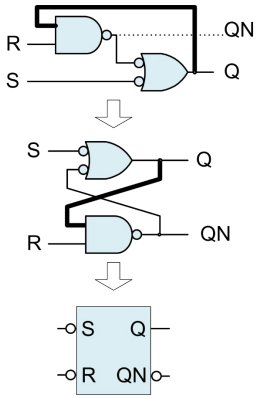
Failure

Testing Hardware Pre-Silicon

- **Idea:** apply testing to the HDL (Hardware Description Lang.)
  - E.g., Verilog
- Benefits over physical testing?



1. Q is undefined until R is asserted
2. Q → ‘1’ when S is asserted
3. Q → '0' when R is asserted
4. Q → '1' when S is asserted
5. Q stays at '1' when S & R asserted
6. Q → '0' when R asserted, S de-asserted



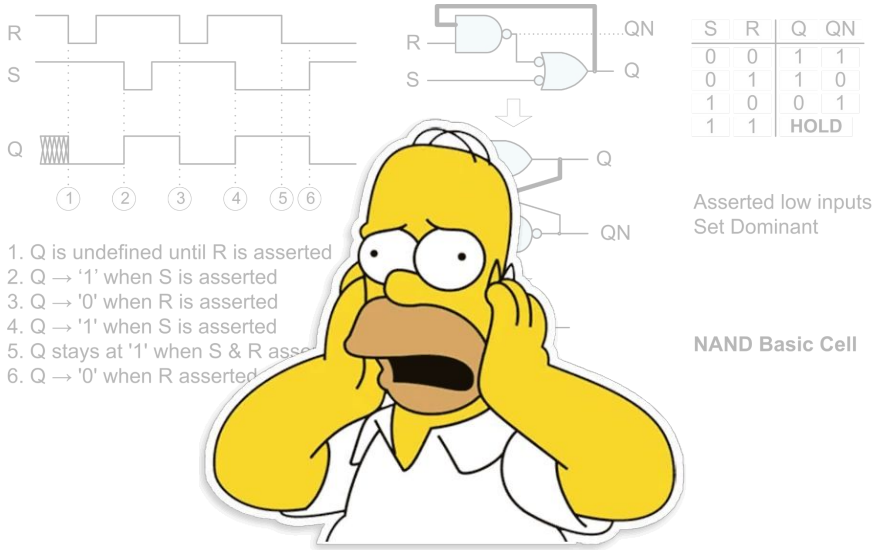
S	R	Q	QN
0	0	1	1
0	1	1	0
1	0	0	1
1	1	**HOLD**	

Asserted low inputs
Set Dominant

NAND Basic Cell

Testing Hardware Pre-Silicon

- **Idea:** apply testing to the HDL (Hardware Description Lang.)
  - E.g., Verilog
- Benefits over physical testing?
  - **Downsides?**



Enter the Simulation

- **Idea:** “translate” HDL to a more workable representation (e.g., C++)

- Benefits?
  - **Downsides?**

Questions?