Hardware Trojan Development and Detection using ATPG tools

Presented by: Adam Zygmontowicz | Faculty Advisor: Jennifer Dworak | Department: Computer Science and Engineering

SMU | Bobby B. Lyle School of Engineering

SMU | HACNet High Assurance Computing And Networking Labs

What if a drone had a computer virus that could not be removed?

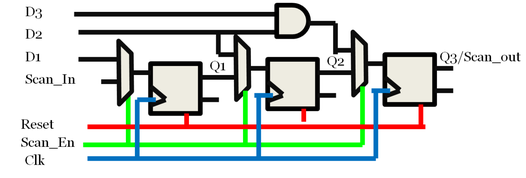
- Hardware Trojans are built-in permanent malicious additions to circuits
- Most integrated circuits are not made in the US including some the DoD's
- Trojans can be inserted at RTL, Manufacturing, 3rd Party IP…
- Trojan have two components: Trigger and Payload
  - Trigger – Is the action that causes the payload to occur
  - Payload – Malicious action, can be active or passive
- Passive Payload
  - Leak data
  - Increased resource usage
- Active Payload
  - DOS – Denial of Service
  - Provide backdoor
  - Change data

Abstract

- Hardware Trojans are malicious additions to computer hardware designs that can cause a lack of security
- Increased costs associated with owning an IC fab has resulted in more ICs being manufacture overseas
- As the result of some DoD ICs being produced overseas, a need to test for Hardware Trojans has grown
- Hardware Trojans have a trigger and a payload, triggers are hard to detect, payloads are attacks on circuits
- Using a "Stuck at" fault model with ATPG tools we are attempting to detect Hardware Trojans

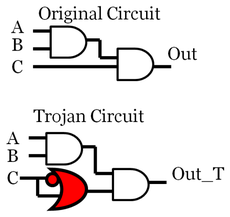
How Hardware is Tested for Defects

- Scan Chains makes for better testing with increased controllability and observability
- Stuck at Fault Model
  - Stuck at 1 – Wire stuck in on state (Ex. Shorted to power)
  - Stuck at 0 – Wire stuck in on state (Ex. Shorted to ground)
  - A good test set will have high fault coverage



What is a Hardware Trojan?

Trojan Hardware is in red



A	B	C	Out	Out_T
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	1
1	1	1	1	1

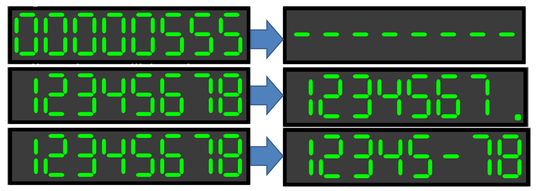
What is at risk with current practices?

- Additional Hardware Embedded Maliciously
- Secure Information
  - Provide attacker encryption keys or master keys
  - Leaking secure information or plaintext
- IP – Intellectual Property
  - Circuit Designs
  - 3rd Party IP
  - Licensed designs

BCD Trojans

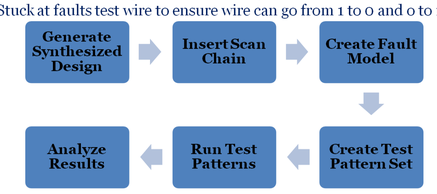
- BCDs are used to display decimal numbers with LEDs
- Example of uses: Alarm Clocks, Timers, Power regulator for a F16 Fighter Jet

Original Display **Trojan Display**



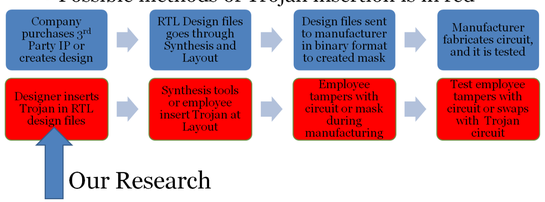
How to detect a Hardware Trojan

- Synopsys and Mentor Graphics Toolsets
  - Synopsys: Design Complier, DFT Complier, and TetraMax ATPG
  - Mentor: Leonardo Spectrum, DFT Advisor, and FastScan ATPG
- ATPG – Automated Test Pattern Generation
  - Toolsets examine design files and generate test patterns
  - ATPG patterns are used to test each fault in the circuit
  - If 100% fault coverage is obtained every wire in the circuit is tested
  - Stuck at faults test wire to ensure wire can go from 1 to 0 and 0 to 1



How are Hardware Trojans Inserted?

- Hardware Trojans can be inserted at many points in the supply chain
- Supply chain is shown in blue
- Possible methods of Trojan insertion is in red



MIPS Trojans

- MIPS processor is a general purpose computer processor
- Can be used to run MIPS assembly code
- Trojan1 – No more instructions
  - Trigger: Add $555_{16}+777_{16}$ (using calculator)
  - Payload: Disables instruction input (kills keyboard)
- Trojan2 – Memory Clearer
  - Trigger: Add $888_{16}+999_{16}$ (using calculator)
  - Payload: Clears all user data (deletes all files in memory)
- Trojan3 – Shadow Registers
  - Opcode = 111111_2 (part of input instruction)
  - Payload: Copies data in processor then saves in memory

Watchdog Processor Trojan

- Watchdog Processors are used to monitor timed-out processes
  - Checks 2 word registers to ensure equivalence
  - If counter times out and the words are not equal, reset occurs
- Trojan:
  - Trigger: Word_input =$AFAB_{16}$
  - Payload: Reset loop does not stop until Global Reset
- Bottom line: Trojan stops circuit from working
  - Ever had a computer stop working? Annoying, isn't it?

Conclusions and Future Work

- Research is ongoing
- 6 Trojans have been created for 3 hardware designs
- ATPG tools have been able to detect Trojans
- Currently developing data analysis scripts
- Mentor Graphics toolsets have been primary focus
- Project will eventually move to Synopsys tools
- How do we identify patterns that detected Trojans?
- Are the patterns functionally valid?