Hardware Trojan Attack II

on HaHa v3.0 Board

We describe an experiment on the Hardware Trojan attack which requires you to insert your own Trojan triggered by temperature sensor measurement signal

Instructor: Dr. Swarup Bhunia
Co-Instructors/TAs: Reiner Dizon-Paradis and Shuo Yang

Theory Background

The taxonomy of Trojan circuits has been presented in various forms, and it continues to evolve as newer attacks and Trojan types are discovered. Figure 1 shows a high-level classification based on variations in activation mechanism and Trojan effect.

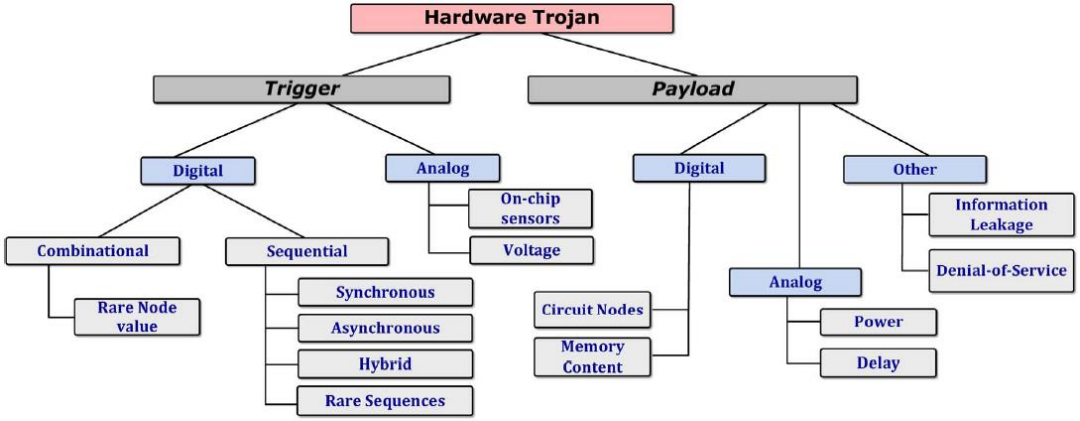

Figure 1 Trojan taxonomy based on trigger and payload mechanisms.

Based on the trigger condition, the hardware Trojans can be classified into analog or digital Trojans. The former is activated by analog conditions such as temperature, delay, or device aging effect, whereas the latter are triggered by some Boolean logic function. Digitally triggered Trojans can again be classified into combinational and sequential types.

In terms of the payload, the Trojan can cause functional failure upon triggering or have a passive effect such as heating of the die or leaking of information. A Trojan can cause an ‘‘information leakage’’ attack, where secret information is leaked by a Trojan via a transmitted radio signal or serial data port. It could also involve a side-channel attack where the information is leaked through the power trace or through thermal radiation or through optical modulation of an output LED. Another type of Trojan payload would be an unauthorized alteration in system behavior.

Experiment Set-up: Configuration

The instruments needed for this experiment are the:

- HAHA V3.0 Board
- Computer
- Heater, e.g., a hairdryer, or cooler, eg. compressed air

The software needed for this experiment are the:

- GOWIN FPGA Designer
- avrdude

Experiment Set-up: Instructions

Part I: Use the Temperature Sensor

In this part, you will use the temperature sensor in the Microcontroller on the HAHA V3.0 Board. Refer to haha.h for the functions that will allow you to read the ADC values from the temperature sensor. After production, the microcontroller is programmed with a calibration value that indicates the ADC value at 85C. You can use this to convert the ADC value you read to a temperature in Celsius. Send this calculated Celsius value to the FPGA over the 8-bit interconnection. The send over the interconnect use the **sendDataToFPGA()** function, making sure to call the **interBegin()** function once during setup.

You can heat the board to be higher than 40℃, but strictly below 80℃ with a heater, i.e., a hairdryer, and see how the value changes. Alternatively, you can use a can of compressed air to cool it.

Part II: Design a system of your own

Use the FPGA to design a circuit of your own. It can have any functions as you want. The only requirement is that it should receive the temperature data from the chip interconnection. Your design can be a counter which shows its value through LEDs or 7 segment display, or your design can show the temperature on the LEDs in binary, or your design can be an encryption machine that encrypts information, etc.

Part III: Insert a temperature triggered Trojan

Modify your design by inserting a Hardware Trojan into it. The Trojan should be activated by the temperature signal: The Trojan will be triggered when the temperature becomes 40℃ or higher. Alternatively, your Trojan can trigger when the board becomes lower than 26℃

Its payload could be any kinds: it could totally halt the original function; or it could induce delays to the design; or it could make the design consume more power, or it could cause information leakage; or it could make the design have other unexpected functions. Whatever the Trojan does, you should be able to observe the malicious effects the Trojan caused.

Your design should come back to normal functions when the devices cool down to be under 40℃. (or above 26℃ if you are using cooling)

Alternative Arrangements:

For those who do not have access to a hairdryer or compressed air can (I will place some air cans in the lab):

1. You can try triggering the Trojan at a slightly higher temperature than room temperature, i.e., lowering the triggering temperature. The board should automatically heat-up as it runs, and you can try safely touching the FPGA chip to alter the temperature slightly.
2. You can also try to trigger the Trojan at a lower temperature with respect to room temperature, i.e., reverse the trigger condition (lower temperature triggers the Trojan). You can try using a moisture-isolated ice bag or a cold object to decrease the temperature.

Measurement, Calculation, and Question

Part I: Use the Temperature Sensor

1) Commit your code (main.c) for using the ADC to read the microcontroller's internal temperature and send it to the FPGA. Try to display the value by creating an FPGA project that reads the chip interconnection signals and setting up a GOWIN Analyzer Oscilloscope (GAO) file. Otherwise, show the value in the LEDs. Your code should be able to update the temperature measurement result, i.e., repeating outputting values time after time.
2) What is the output value of the temperature sensor under room temperature? Attach a screenshot showing the content of the GAO tool for the chip interconnection (CM) signal or the value in the LEDs.
3) What is the output value of the temperature sensor when you heat or cool the microcontroller? Attach a screenshot showing the content of the GAO tool for the chip interconnection (CM) signal or the value in the LEDs.

Part II: Design a system of your own

1) Describe your design. What is the function? Commit your Verilog code and Gowin project.
2) How does your design receive the temperature data?
3) How many Logic Elements (LEs) does your circuit use?

Part III: Insert a temperature triggered Trojan

1) Describe your Trojan. How is it triggered?
2) What is the payload? How do you observe the phenomenon when the Trojan is activated? Attach pictures as needed.
3) After inserting the Trojan, how many LEs does your design use now? How much is the hardware overhead?
4) Commit your Verilog with the trojan included.
5) What classification does your Trojan belong to? Refer to Figure 1 to answer this question.

References and Further Reading

[1] http://securityaffairs.co/wordpress/17875/hacking/undetectable-hardware-trojan-reality.html

[2] Bhunia, Swarup, et al. "Hardware Trojan attacks: threat analysis and countermeasures." Proceedings of the IEEE 102.8 (2014): 1229-1247.

[3] Roy, Debapriya Basu, et al. "Reconfigurable LUT: A Double Edged Sword for Security-Critical Applications." International Conference on Security, Privacy, and Applied Cryptography Engineering. Springer International Publishing, 2015.

[4] Majzoub, Sohaib, and Hassan Diab. "Mapping and performance analysis of lookup table implementations on reconfigurable platform." 2007 IEEE/ACS International Conference on Computer Systems and Applications. IEEE, 2007.

[5] http://ww1.microchip.com/downloads/en/Appnotes/00002535A.pdf