Hardware trojans

(can we trust anything?)

ISSES

Co-funded by the
Erasmus+ Programme
of the European Union

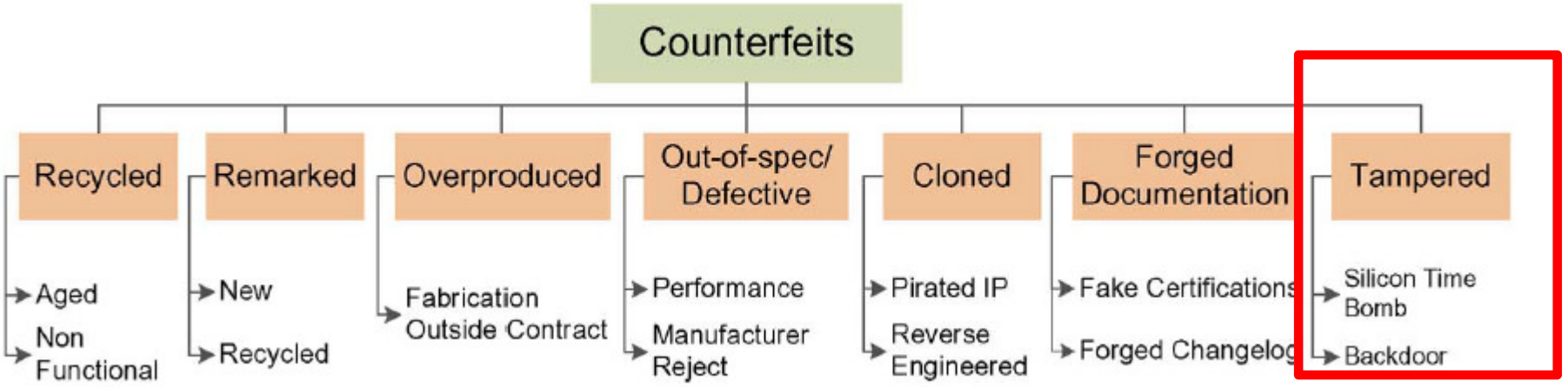
Can we trust hardware?

- Sen. Liberman 2003: “National security aspects of the Global migration of the US semiconductor industry”
- Chips are produced in foreign countries
  - No control over production
  - Complex IC with millions of gates
  - Difficult verification without destruction
  - Small changes can present threat to IC integrity and device operation

ISSES

Co-funded by the
Erasmus+ Programme
of the European Union

Integrated circuit counterfeit

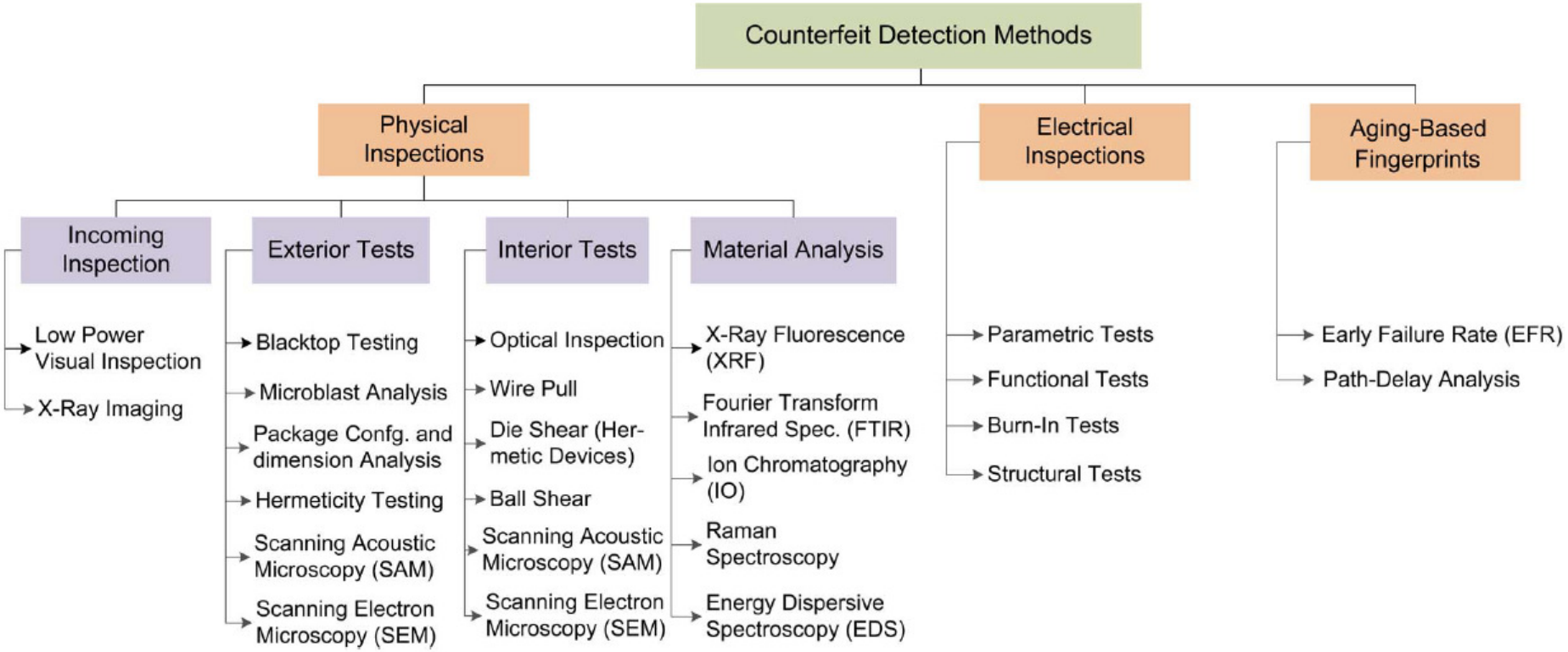


- Affects
  - IC and system operation (functionality)
  - IC and system reliability
  - IC and system security

ISSES

Co-funded by the Erasmus+ Programme of the European Union

IC inspection



ISSES
Co-funded by the Erasmus+ Programme of the European Union

Problems with the inspection

- Only destructive inspection ensures 100% verification
- Non destructive physical inspection methods not reliable
- Parametric and functional testing has limited set of test input/outputs (hardware trojans can avoid),
- Power consumption and path-delay analysis – new methods which can detect some types of trojans

ISSES

Co-funded by the
Erasmus+ Programme
of the European Union

Hardware trojan characteristics

- Small footprint (can be as small as 0.01% of the IC size)
- Can change digital or analog behaviour
- Can destroy the circuit
- Can be active all the time
- Can be triggered upon some special signal or with delayed start

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Hardware trojan payload

1. Broadcast to the attacker some internal signals, which are often sensitive data (e.g. the encryption key).
2. Compromise the function of the circuits (e.g. to replace the plaintext with other preset information).
3. Destroy the chip.

ISSES
Co-funded by the Erasmus+ Programme of the European Union

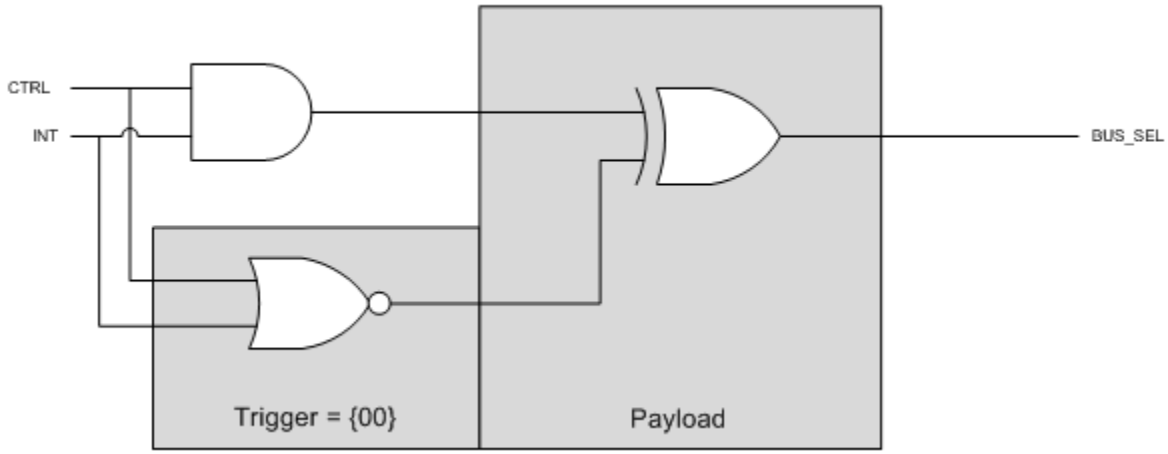
Hardware trojan triggers

1. The attacker can physically access the device and can give special input to trigger the Trojan directly.
2. The Trojan is triggered internally: by a specific input event, counter, or other signal change.
3. No trigger; the trojan is always activated.

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Hardware trojan types

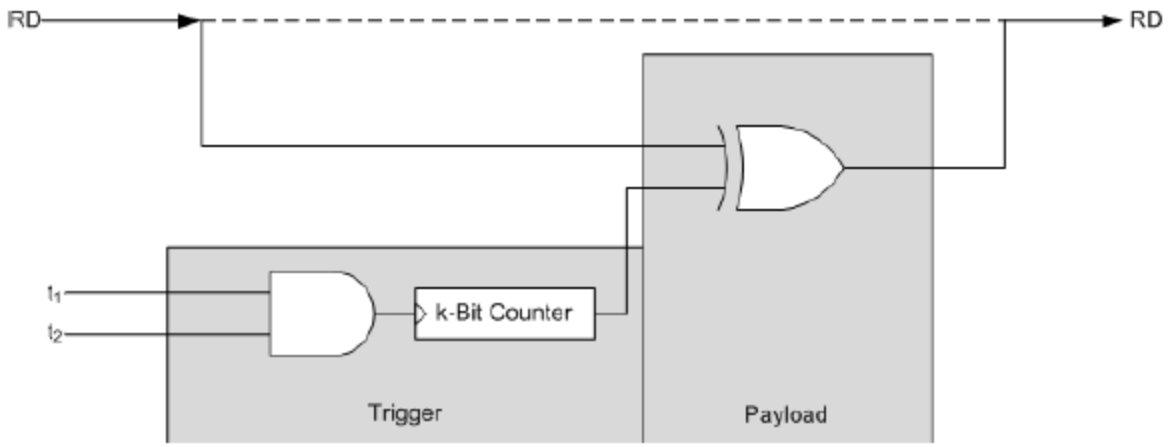
- Combinational trojan
- The trigger part monitors a control signal and an undefined instruction interrupt signal. When these two signals are both in low voltage levels, the payload part of the Trojan is activated and reverses the bus selection signal to select the wrong signals in the inner computation.



ISSES
Co-funded by the
Erasmus+ Programme
of the European Union

Hardware trojan types

- Sequential trojan
- Trigger part contains a k-bit counter. The Trojan won't do harm to the design even if the rarely occurrence event is accidently fulfilled. The same rare event should occur for 2k times until payload changes the read signal. Probably only attackers can construct the special input pattern sequence by purpose.



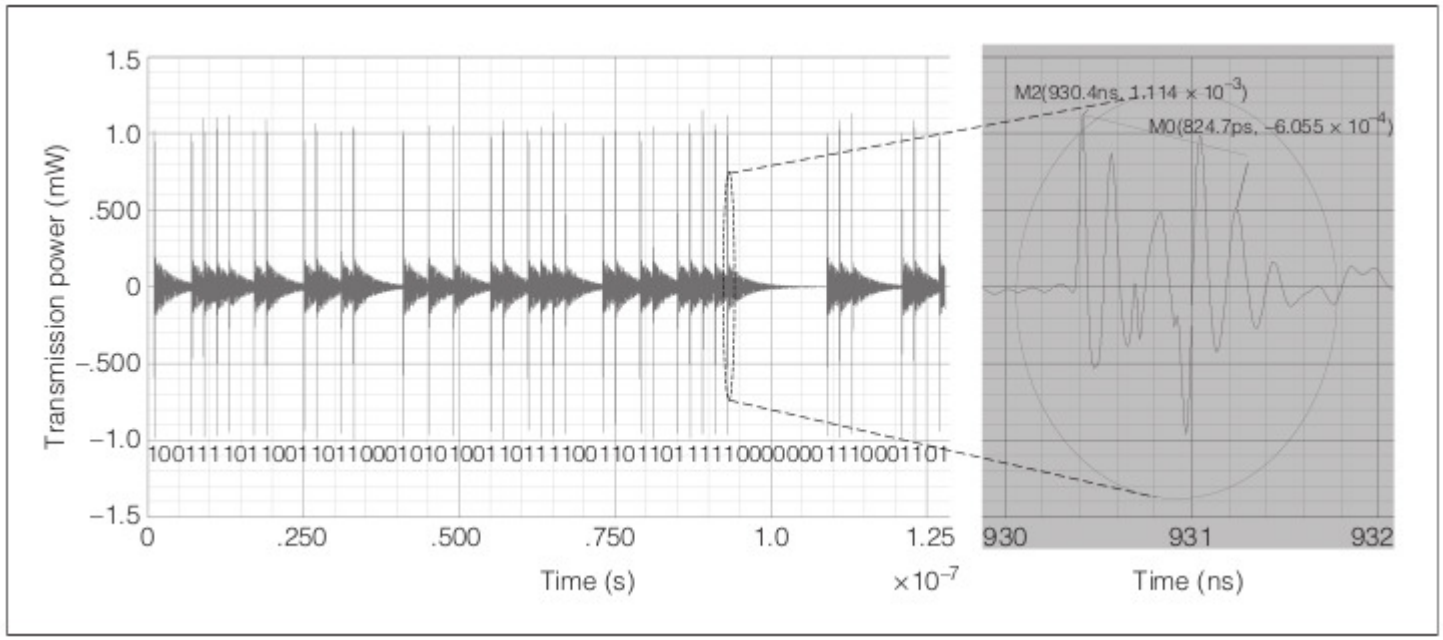
ISSES
Co-funded by the
Erasmus+ Programme
of the European Union

Hardware trojan example

- Idea: insert into the (encrypted) WiFi signal DES encryption key.
- Modification should be unnoticeable to the regular users.
- The attacker can extract the encryption key knowing the modification

ISSES
Co-funded by the Erasmus+ Programme of the European Union

WiFi analog signal

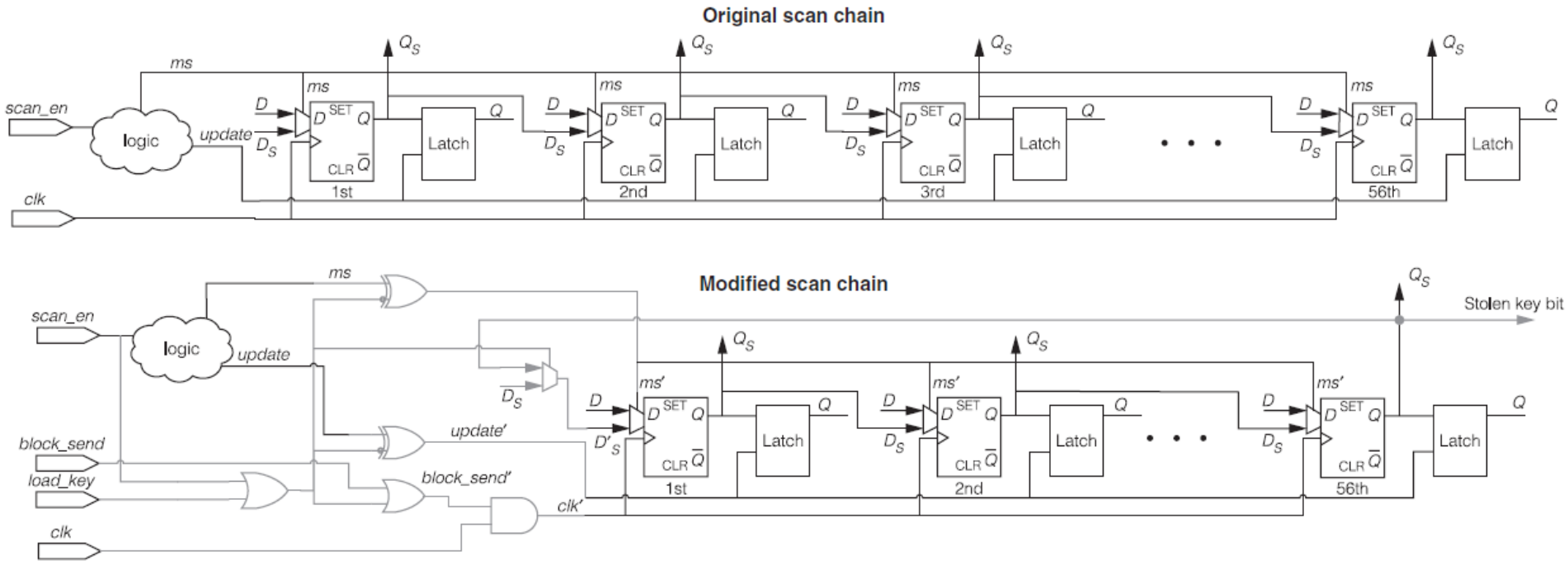


- Transmitting a logical 1 involves 5 to 7 peaks of amplitude over 300 uW with at least one of them over 900 uW.
- The actual performances of each individual chip will vary, depending on the fabrication process variations.

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Extracting the key

- Extracting the key bitwise, through a rotator made of the 56 enhanced scan flip-flops where it is stored

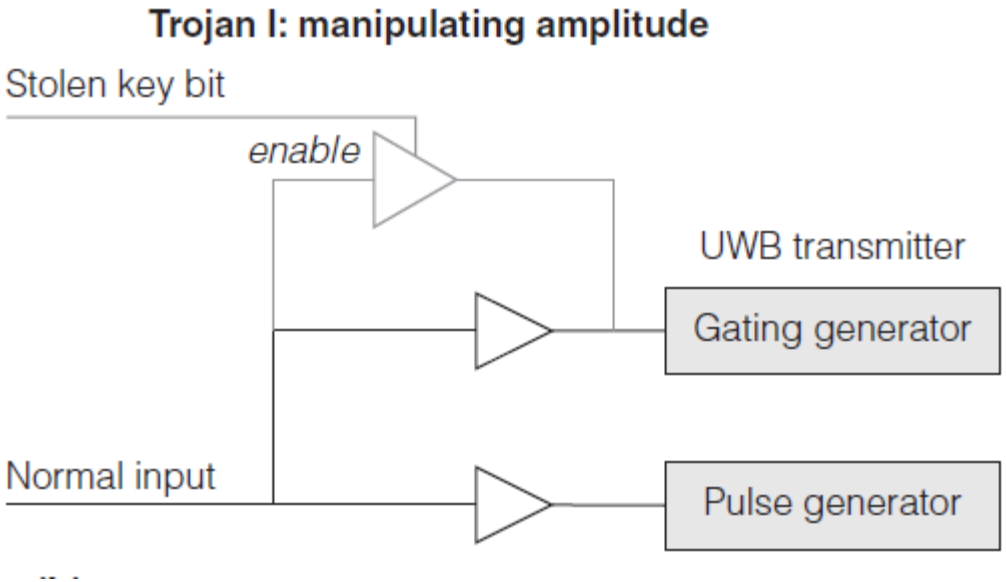


ISSES

Co-funded by the Erasmus+ Programme of the European Union

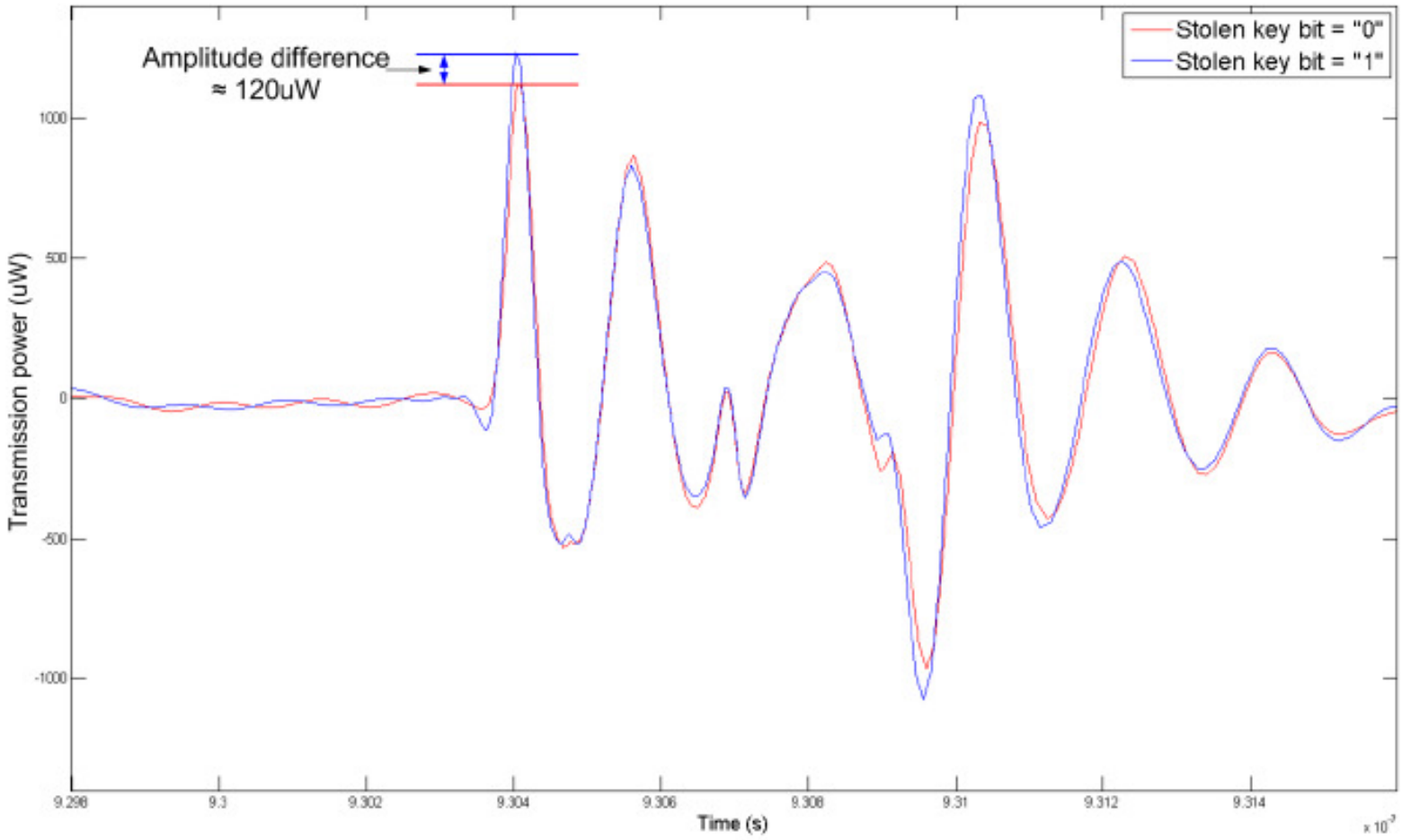
Encoding the key into the amplitude

- difference of 120 uW in the maximum amplitude.
- This difference is well within the margins allowed for process variations and operating-condition fluctuations and would not raise any suspicion.



ISSES
Co-funded by the Erasmus+ Programme of the European Union

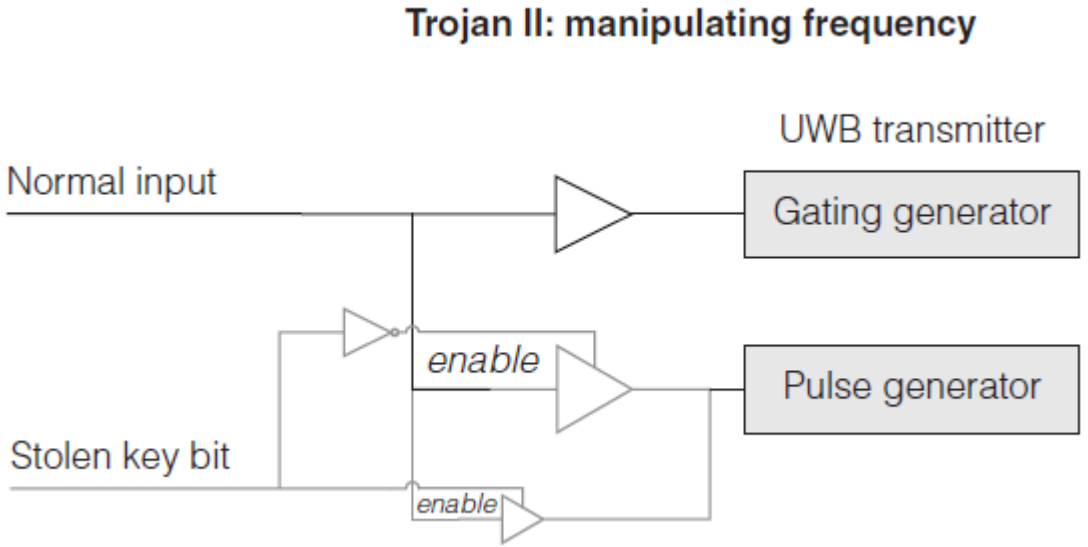
Effect – changed amplitude



ISSES
Co-funded by the
Erasmus+ Programme
of the European Union

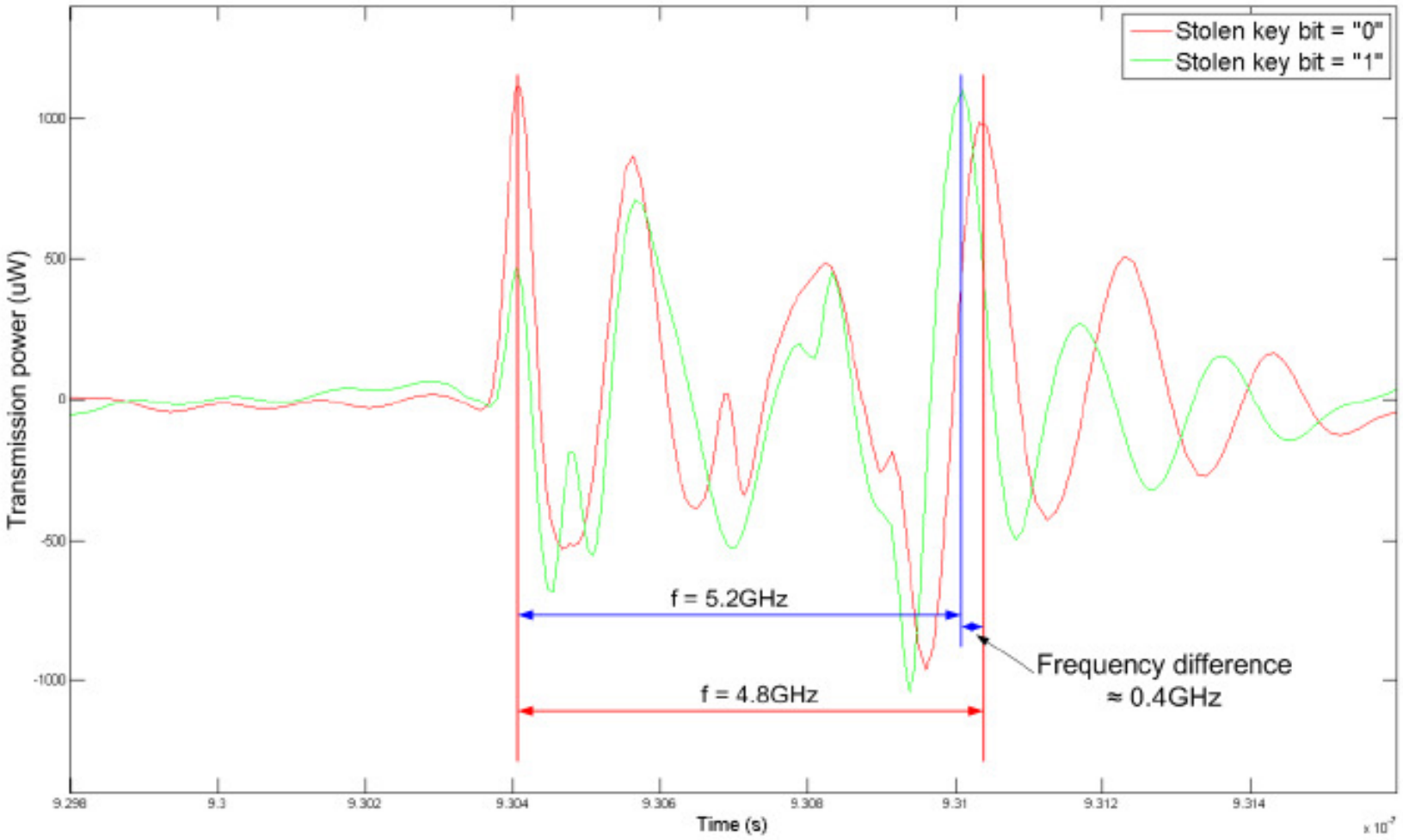
Encoding the key into the frequency

- The difference in the stolen key bit value is reflected as a 0.4-GHz difference in the frequency.
- This difference is within the margins allowed for process variations and operating-condition fluctuations and would not raise any suspicion.



ISSES
Co-funded by the Erasmus+ Programme of the European Union

Effect – changed frequency



ISSES
Co-funded by the
Erasmus+ Programme
of the European Union

Summary

- WiFi leaks DES encryption key while maintaining normal operation
- The overall area overhead incurred by each of these Trojans is around 0.02% of the digital part of the chip.
- Very small increase in power consumption (hard to detect, false alarms can appear)
- Statistical analysis

ISSES

Co-funded by the Erasmus+ Programme of the European Union

What about compilers?

- “We tested thirteen production-quality C compilers and, for each, found situations in which the compiler generated incorrect code for accessing volatile variables.”, Eide and Regehr in Volatiles Are Miscompiled, and What to Do about It, 2008
- “So far, we have reported more than 325 previously unknown bugs to compiler developers. Moreover, every compiler that we tested has been found to crash and also to silently generate wrong code when presented with valid inputs.”, Yang, Chen, Eide and Regehr in Finding and Understanding Bugs in C Compilers, 2010
- For non-critical, "everyday" software, miscompilation is an annoyance but not a major issue: bugs introduced by the compiler are negligible compared to those already present in the source program.
- The situation changes dramatically, however, for safety-critical or mission-critical software, where human lives, critical infrastructures, or highly-sensitive information are at stake.

ISSES
Co-funded by the Erasmus+ Programme of the European Union

What about the compilers?

- Miscompilation can be a way to plant a backdoor
- Is the compiled code logically the same as the code in high level programming language?
- Formal verification of the C compiler: CompCert - http://compcert.inria.fr/

ISSES
Co-funded by the Erasmus+ Programme of the European Union