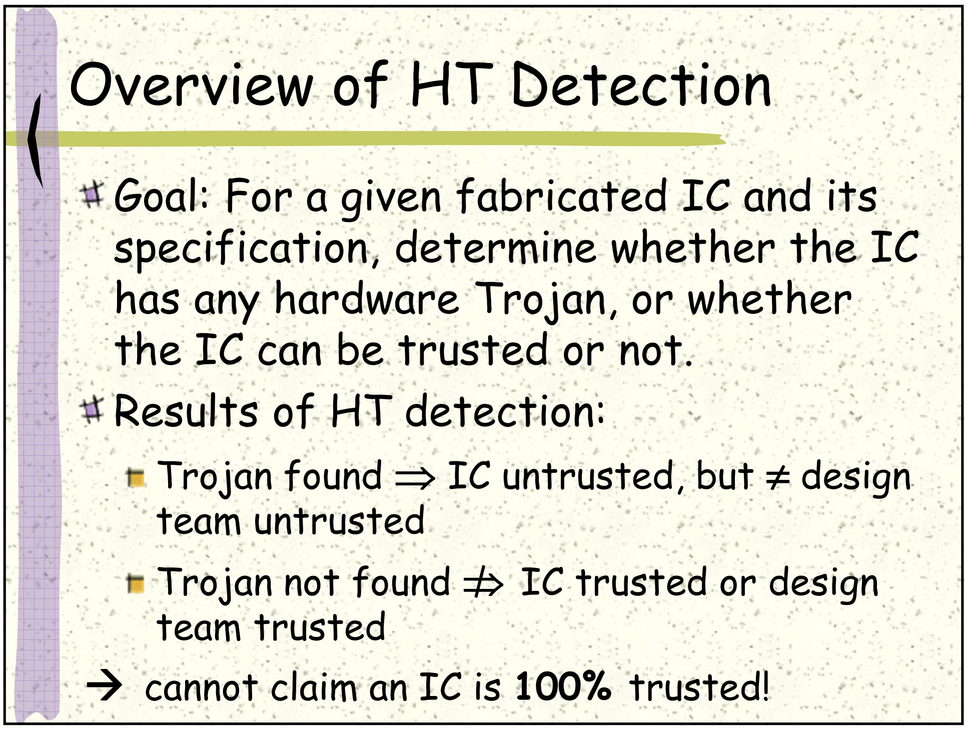
HT and Trusted IC
-- HT Detection Overview

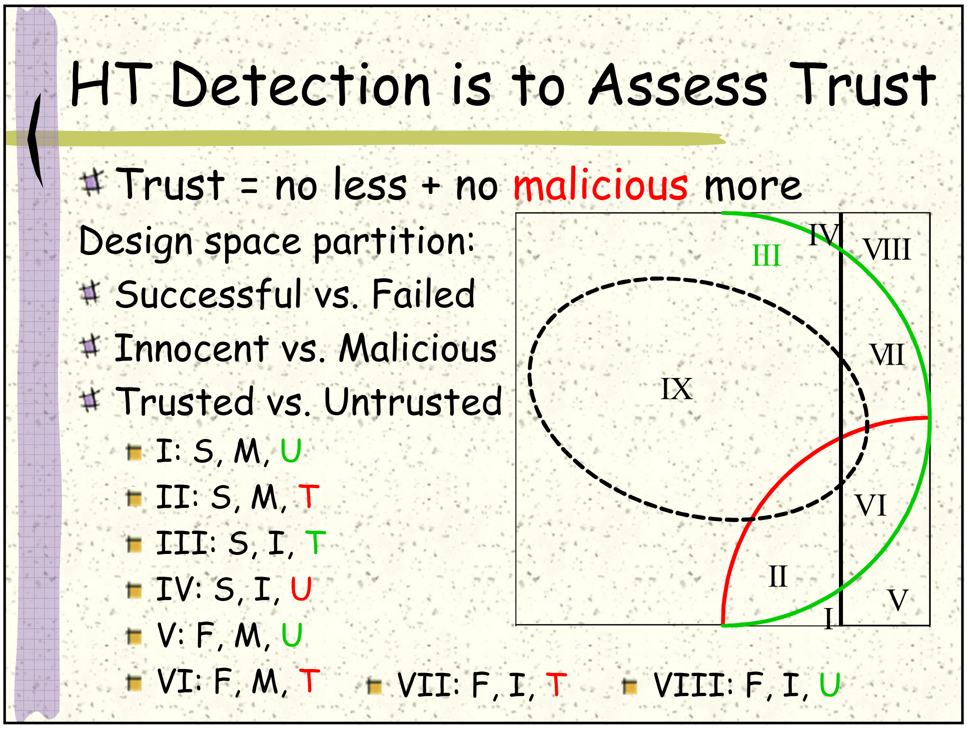
Cybersecurity Specialization
-- Hardware Security



Overview of HT Detection

- Goal: For a given fabricated IC and its specification, determine whether the IC has any hardware Trojan, or whether the IC can be trusted or not.
- Results of HT detection:
  - Trojan found ⇒ IC untrusted, but ≠ design team untrusted
  - Trojan not found ⇏ IC trusted or design team trusted

→ cannot claim an IC is **100%** trusted!



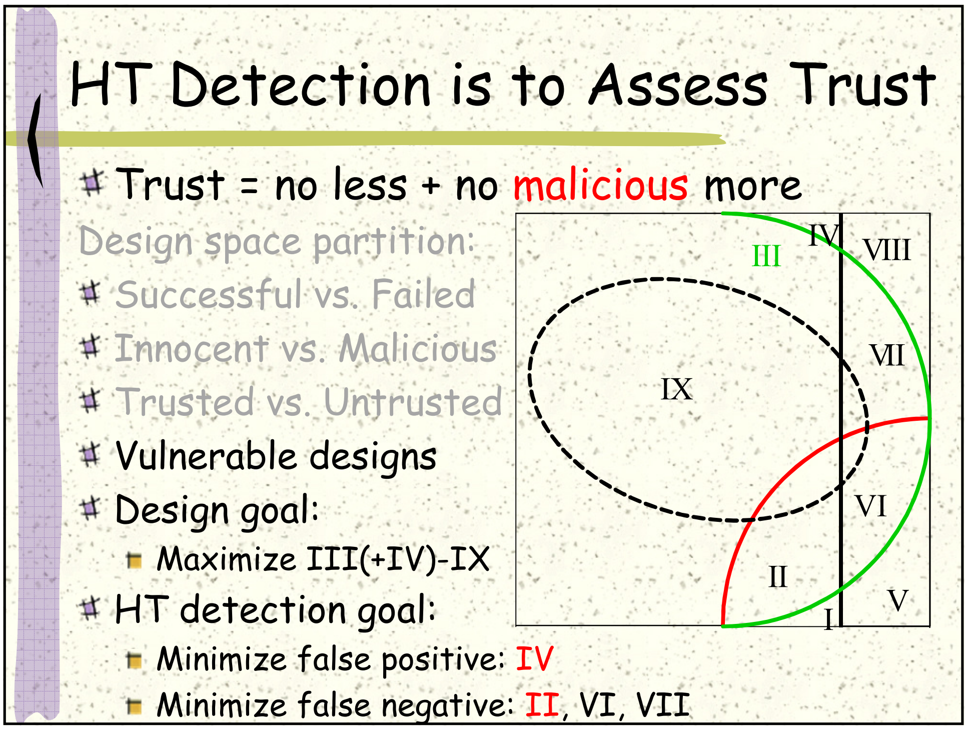
HT Detection is to Assess Trust

- Trust = no less + no malicious more

Design space partition:

- Successful vs. Failed
- Innocent vs. Malicious
- Trusted vs. Untrusted
  - I: S, M, U
  - II: S, M, T
  - III: S, I, T
  - IV: S, I, U
  - V: F, M, U
  - VI: F, M, T
  - VII: F, I, T
  - VIII: F, I, U

III
IV
VIII
VII
IX
VI
II
V
I



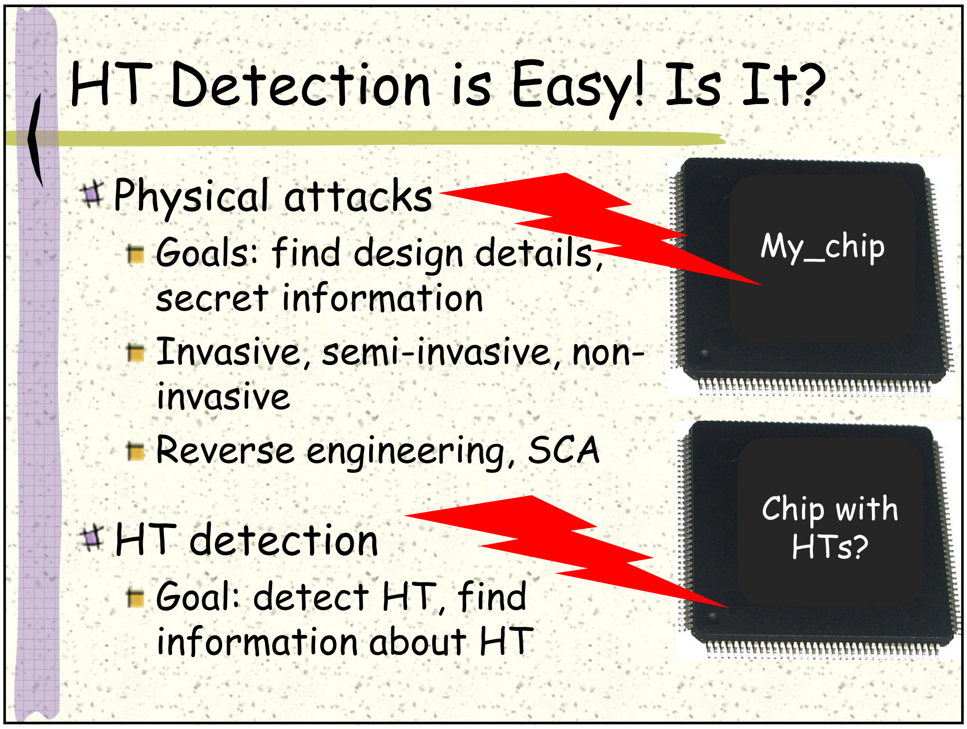
HT Detection is to Assess Trust

- Trust = no less + no malicious more

Design space partition:

- Successful vs. Failed
- Innocent vs. Malicious
- Trusted vs. Untrusted
- Vulnerable designs
- Design goal:
  - Maximize III(+IV)-IX
- HT detection goal:
  - Minimize false positive: IV
  - Minimize false negative: II, VI, VII

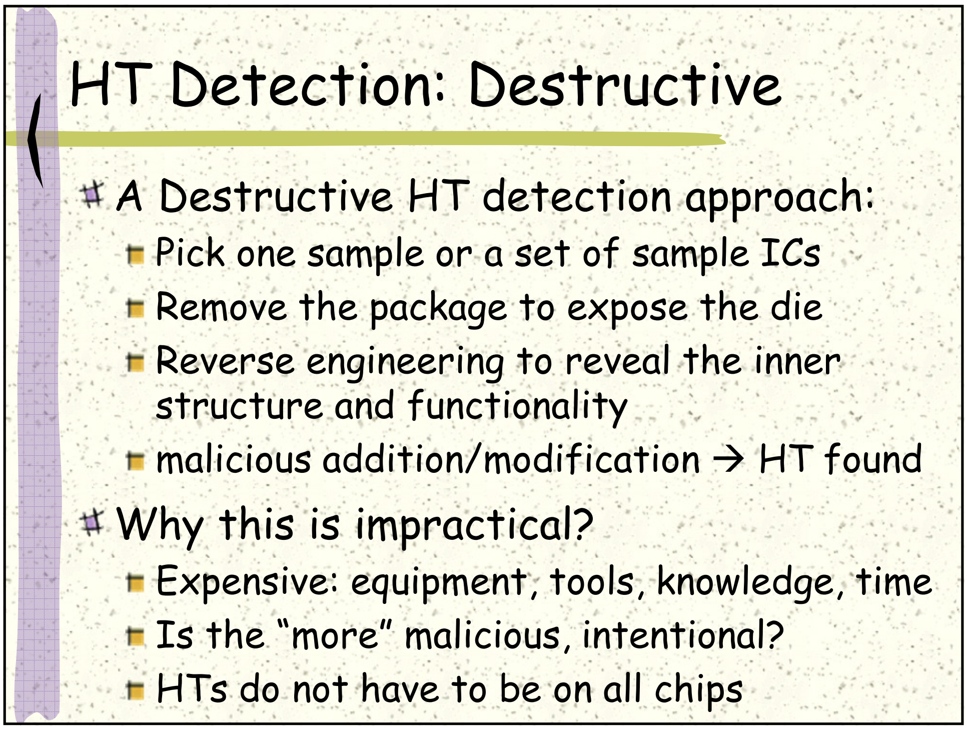
III
IV
VIII
VII
IX
VI
II
V
I



HT Detection is Easy! Is It?

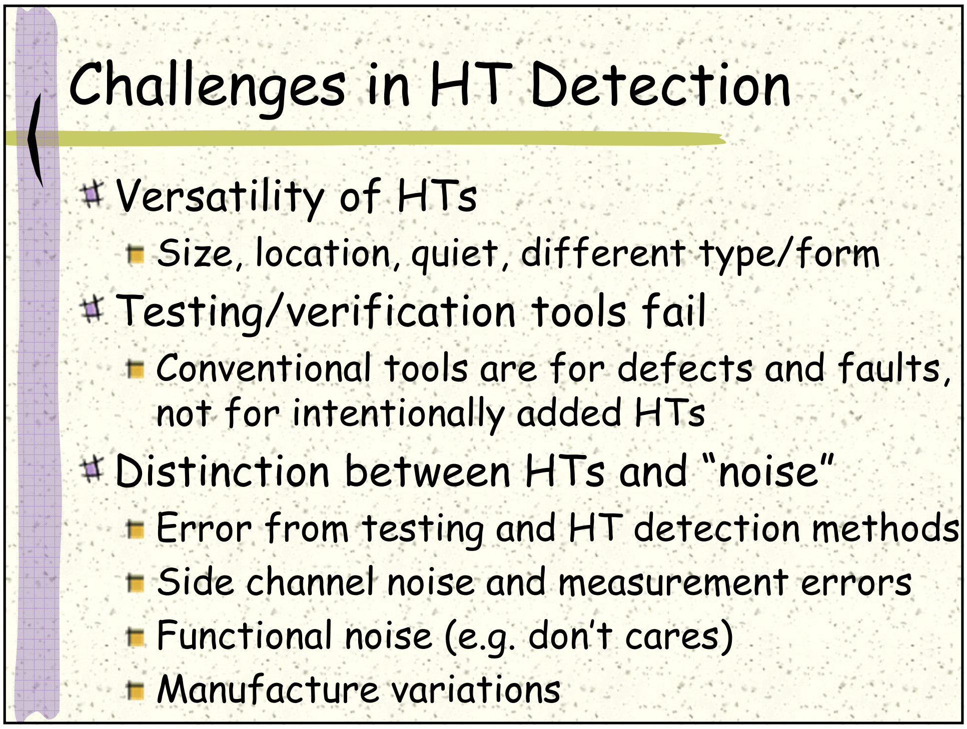
- Physical attacks
  - Goals: find design details, secret information
  - Invasive, semi-invasive, non-invasive
  - Reverse engineering, SCA
- HT detection
  - Goal: detect HT, find information about HT

My_chip
Chip with HTs?



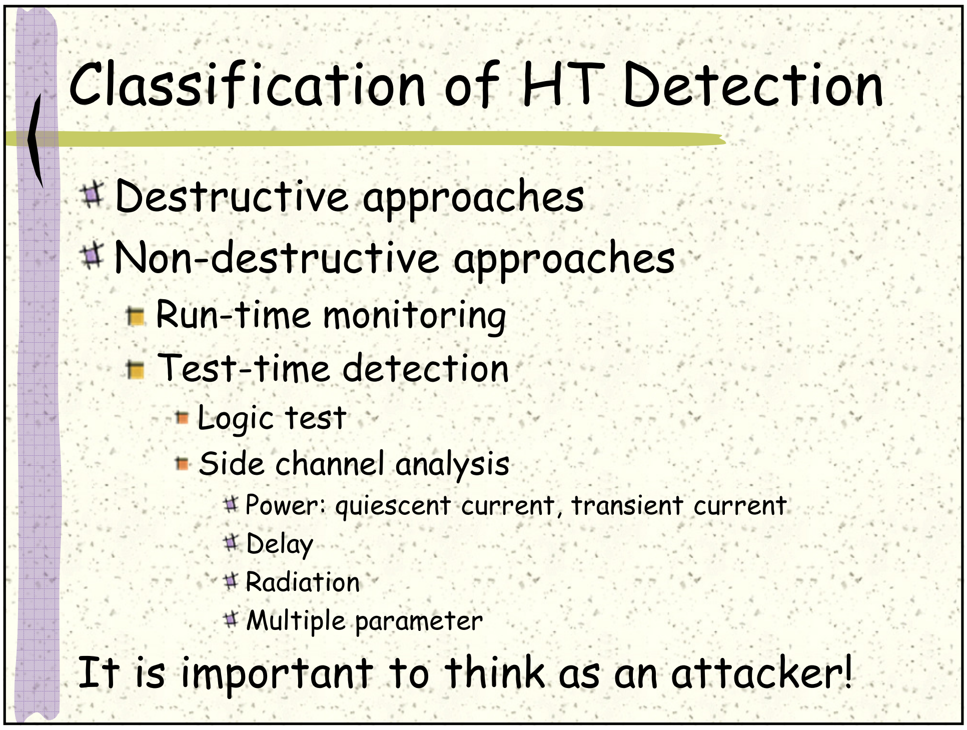
HT Detection: Destructive

- A Destructive HT detection approach:
  - Pick one sample or a set of sample ICs
  - Remove the package to expose the die
  - Reverse engineering to reveal the inner structure and functionality
  - malicious addition/modification → HT found
- Why this is impractical?
  - Expensive: equipment, tools, knowledge, time
  - Is the "more" malicious, intentional?
  - HTs do not have to be on all chips



Challenges in HT Detection

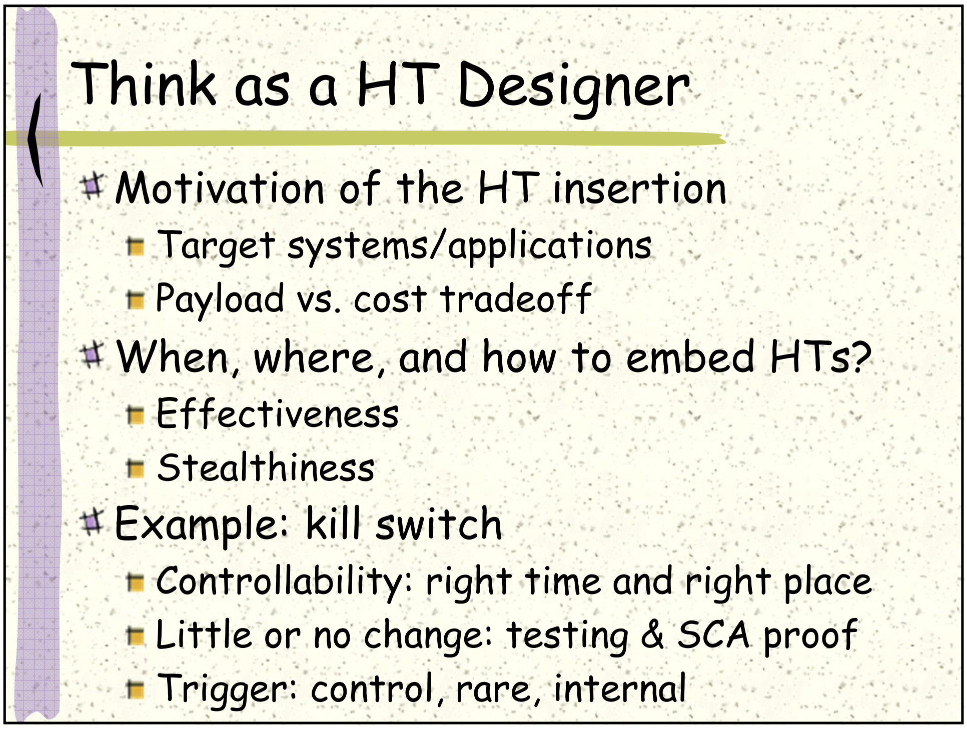
- Versatility of HTs
  - Size, location, quiet, different type/form
- Testing/verification tools fail
  - Conventional tools are for defects and faults, not for intentionally added HTs
- Distinction between HTs and “noise”
  - Error from testing and HT detection methods
  - Side channel noise and measurement errors
  - Functional noise (e.g. don’t cares)
  - Manufacture variations



Classification of HT Detection

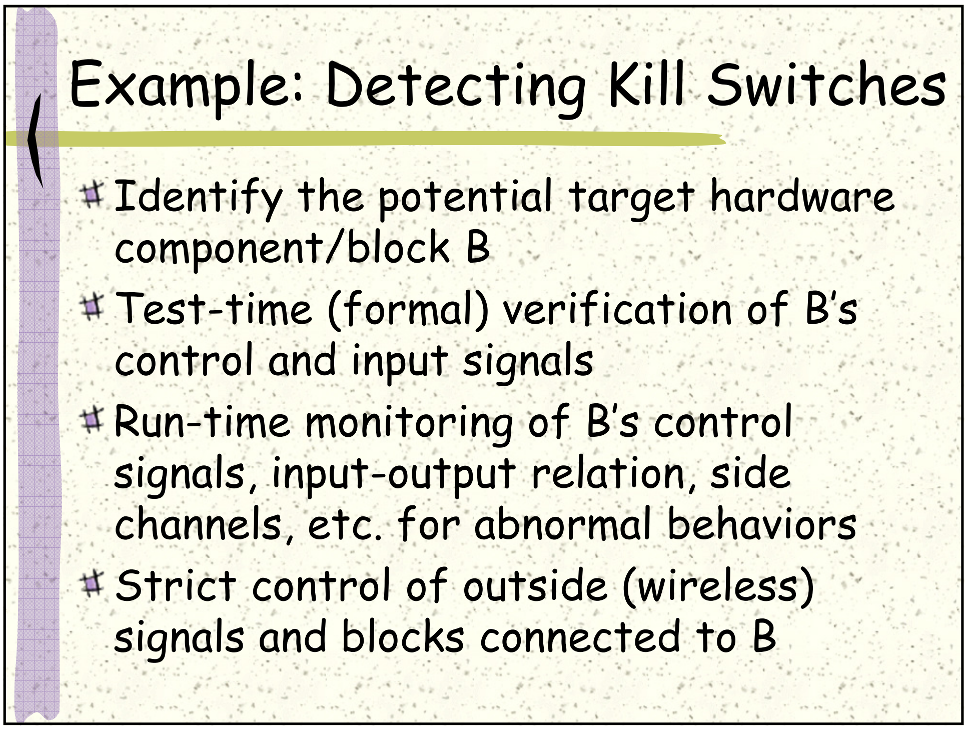
- Destructive approaches
- Non-destructive approaches
  - Run-time monitoring
  - Test-time detection
    - Logic test
    - Side channel analysis
      - Power: quiescent current, transient current
      - Delay
      - Radiation
      - Multiple parameter

It is important to think as an attacker!



Think as a HT Designer

- Motivation of the HT insertion
  - Target systems/applications
  - Payload vs. cost tradeoff
- When, where, and how to embed HTs?
  - Effectiveness
  - Stealthiness
- Example: kill switch
  - Controllability: right time and right place
  - Little or no change: testing & SCA proof
  - Trigger: control, rare, internal



Example: Detecting Kill Switches

- Identify the potential target hardware component/block B
- Test-time (formal) verification of B's control and input signals
- Run-time monitoring of B's control signals, input-output relation, side channels, etc. for abnormal behaviors
- Strict control of outside (wireless) signals and blocks connected to B