Lecture 6: Side channel attacks

- Homework 3 has been posted, due Monday 2/10
- Textbook reading for this week: *Serious Cryptography*, ch. 4, pg. 13-23
- Note: will have a guest lecture next Tuesday

Crypto = Scientific field at intersection of many disciplines

Algorithms

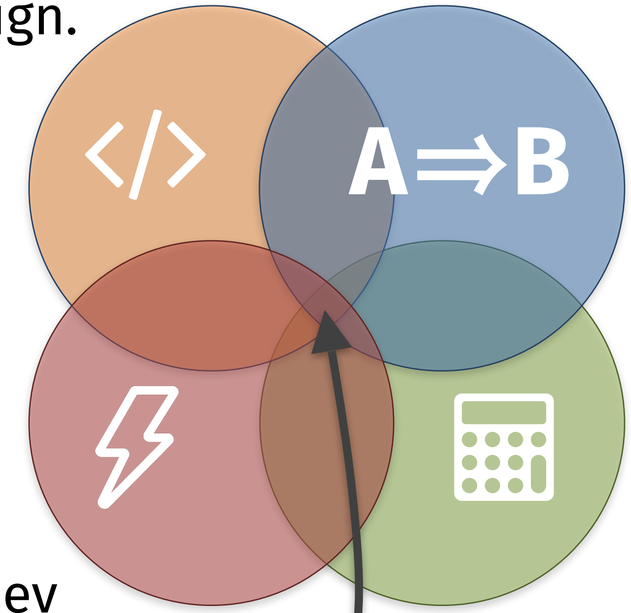
Known for cipher design.

Primarily found in European academia.

Complexity theory

Known for reductions.

Primarily found in American academia.



Engineering

Known for software dev and side channel attacks.

Primarily found in industry.

Mathematics

Known for cryptanalysis.

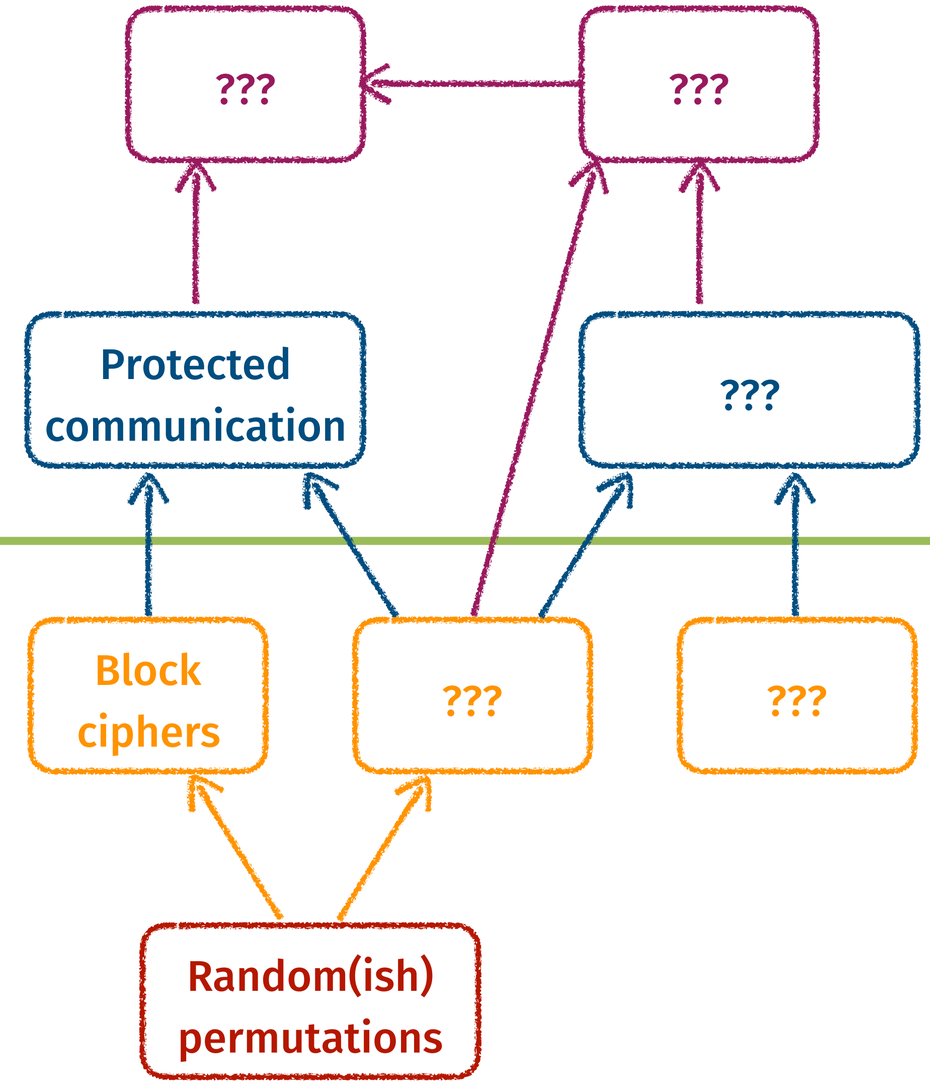
Primarily found in government.

This class

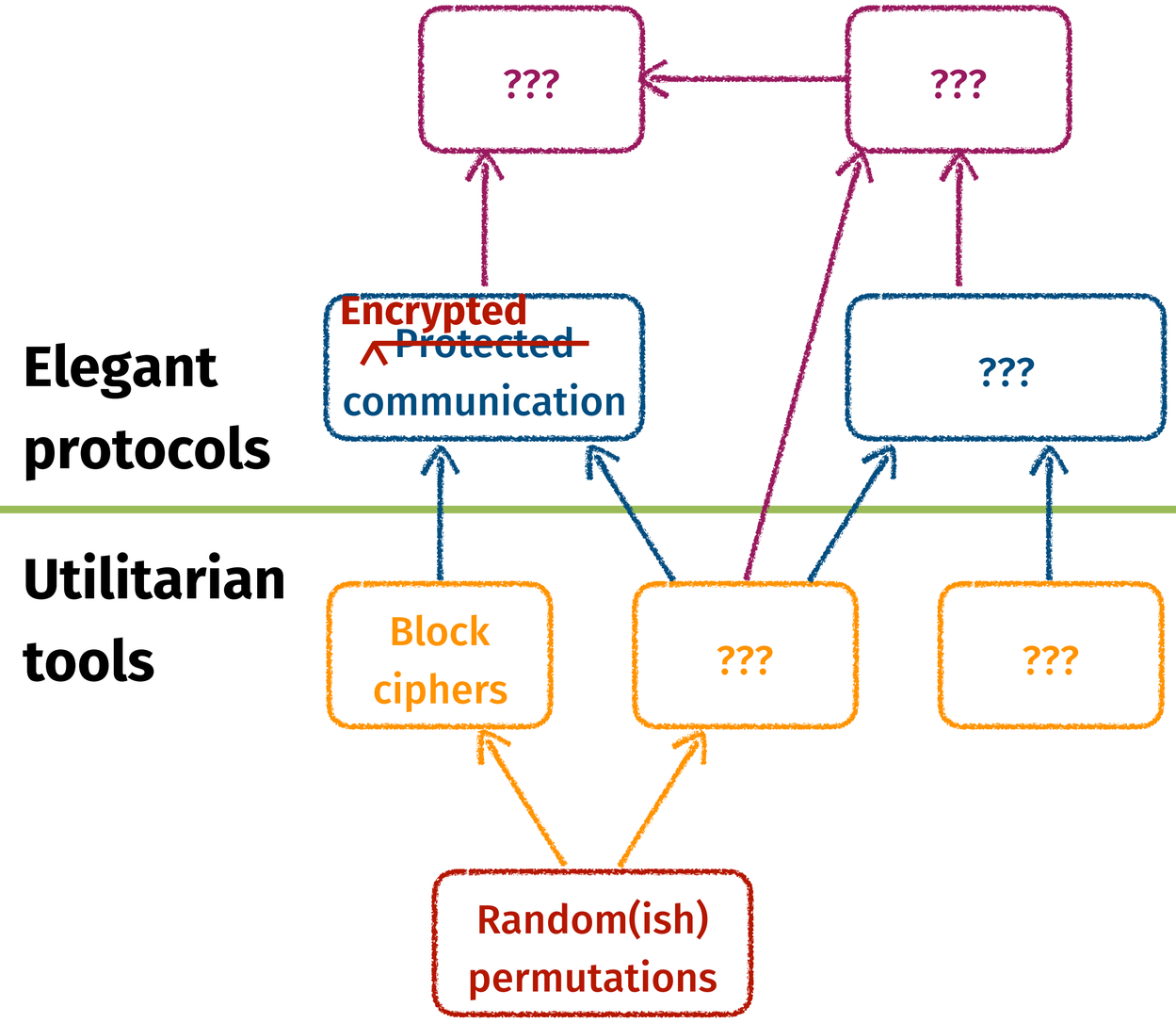
Crypto in this course

Elegant protocols

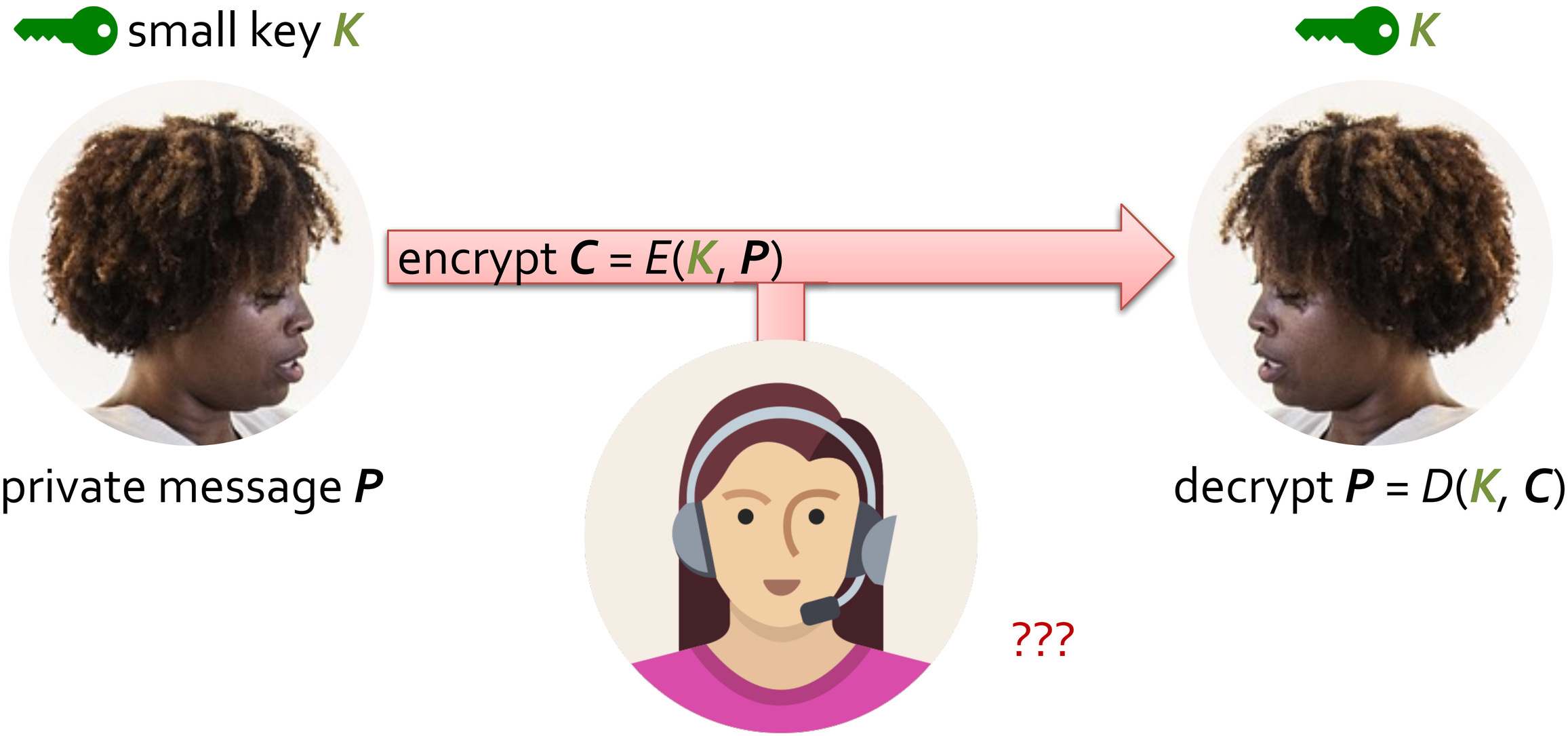
Utilitarian tools



Crypto in this course so far



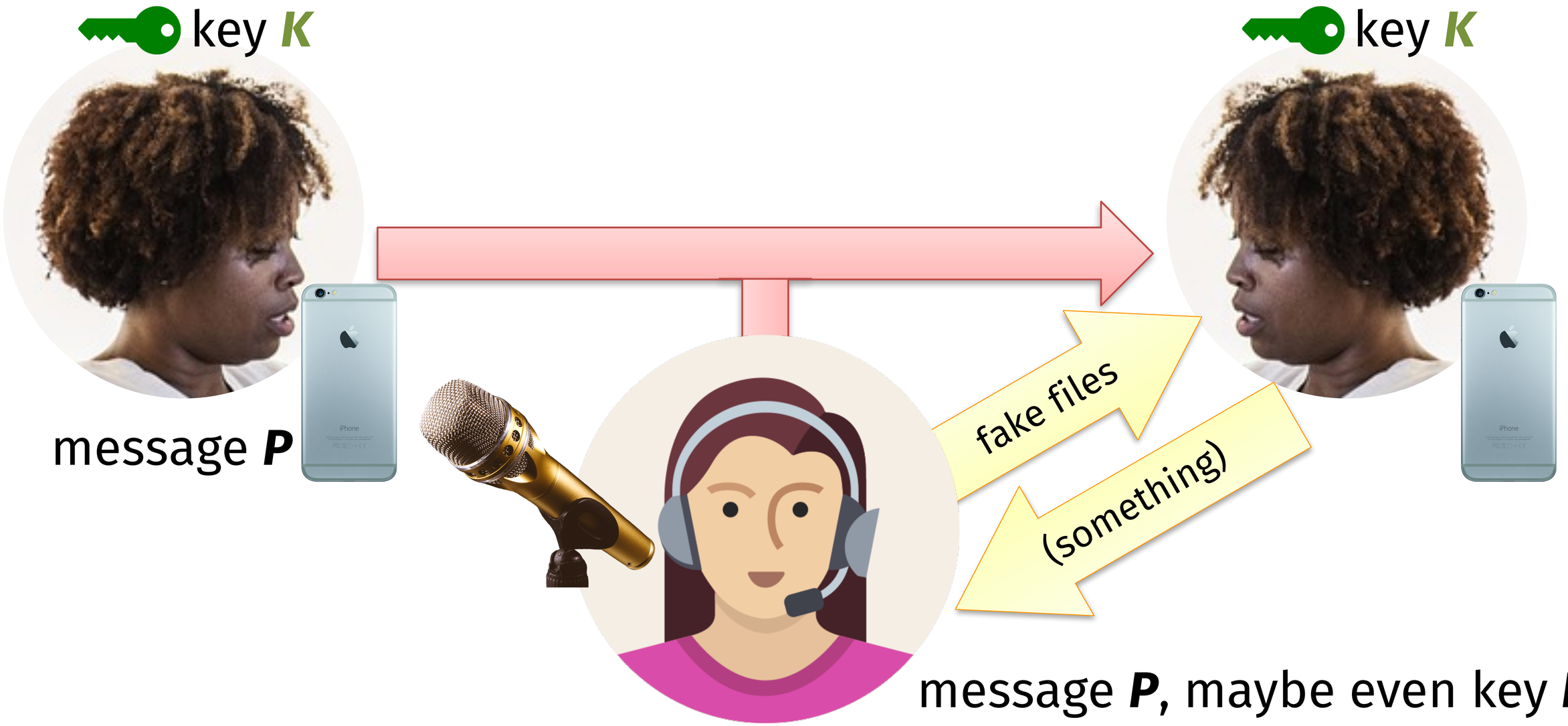
Recap: protecting data confidentiality (CBC or CTR)

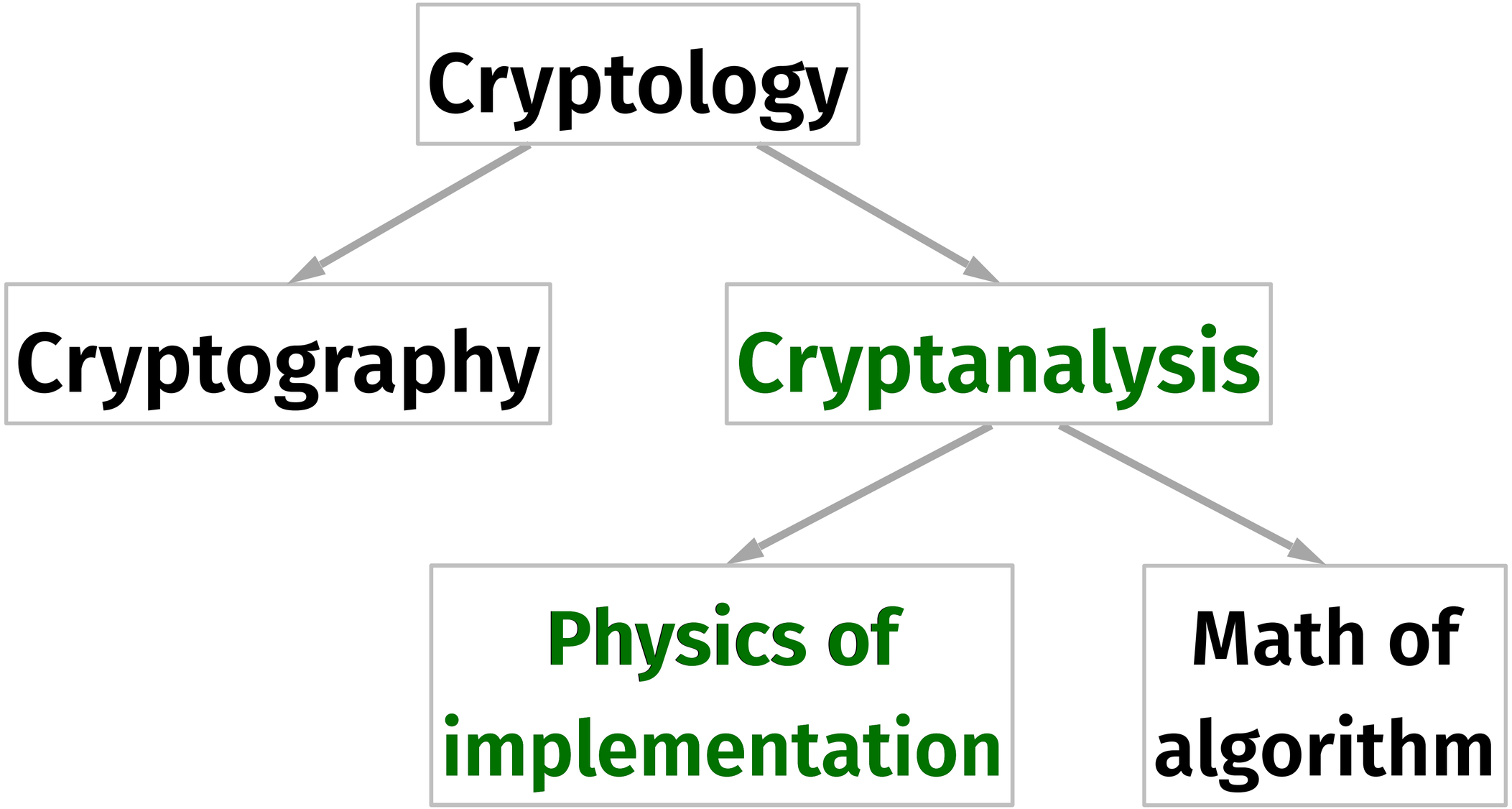


“Confidentiality xor authenticity is **not possible**.
If you don't have both, often you don't have either.”

–Prof. Matthew Green, Johns Hopkins

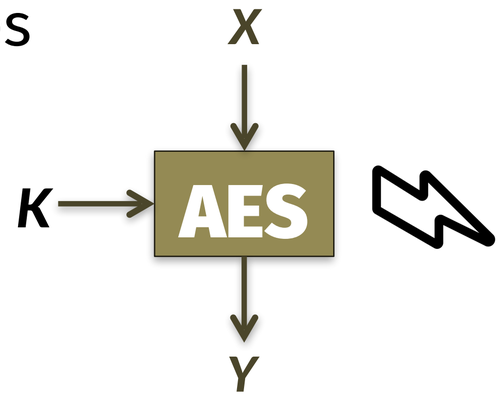
Today + next week: *breaking* data protections




Cryptology
Cryptography
Cryptanalysis
Physics of implementation
Math of algorithm

What happens when we implement crypto?

- So far, we have analyzed the security of cryptographic algorithms
- Security definitions ensure that the cryptosystem's output is "harmless" if the adversary Eve can only query the algorithms by providing inputs
  - Pseudorandomness: unpredictable outputs, even given part of the truth table
  - IND$-CPA: ciphertexts look random, even if Eve chooses messages
- But, **implementations of crypto** can reveal more than its desired outputs



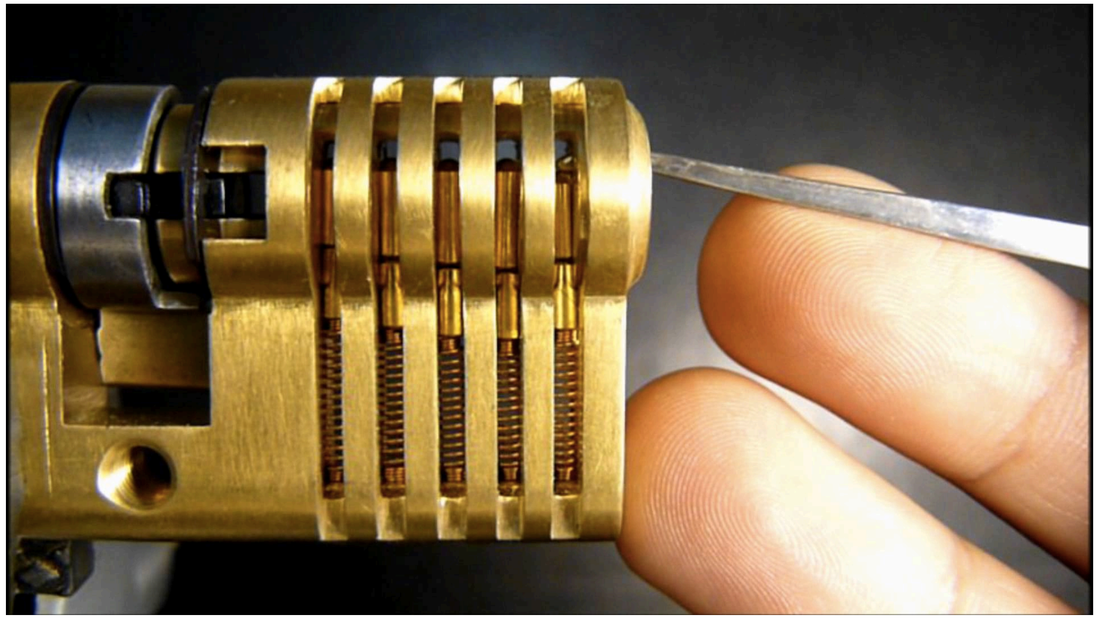
- Collectively we refer to these issues as ***side channels***: they're potential channels of information that are outside of our definitions

Side channel attacks on crypto

- *Issue:* Physical inspection of a device can reveal more than its outputs
- *Sources of extra information:* power, sound, optics, time, cache, error messages, …
- *Environments to attack:* PC software or dedicated hardware devices
- *Method of attack:* divide and conquer (like picking a lock)

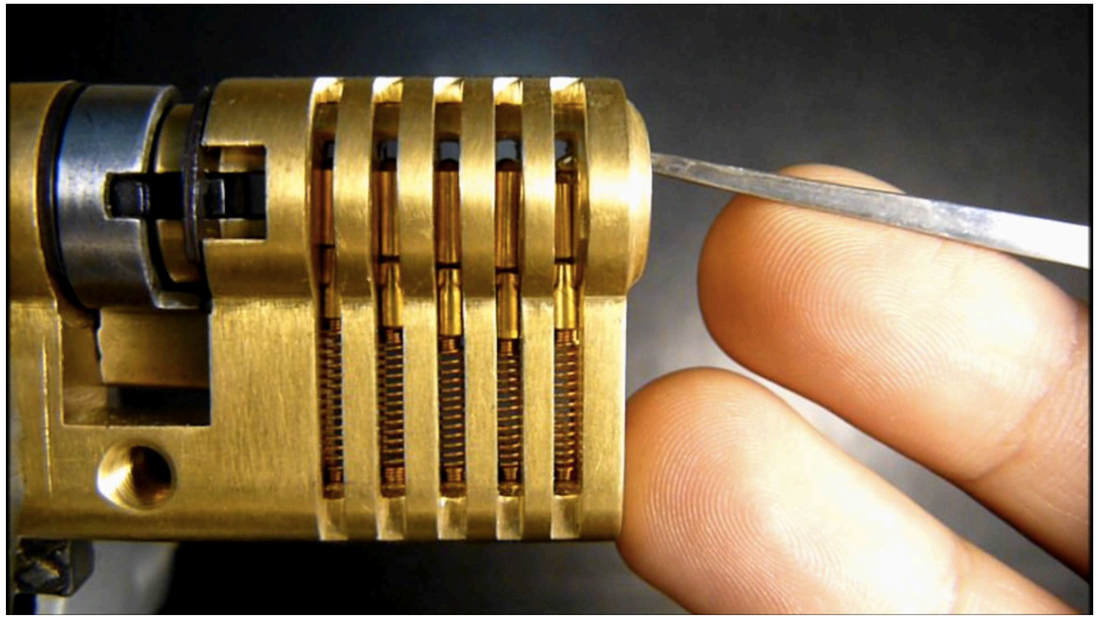
Divide and conquer (for locks)

- 5 pins, each with ~10 locations
- Lock only opens if all pins are in the right location (using the key)
  - Brute force search of the lock should take 10^5 attempts
- Lockpicking: somehow, check the pins separately
  - This attack only requires 5*10 work



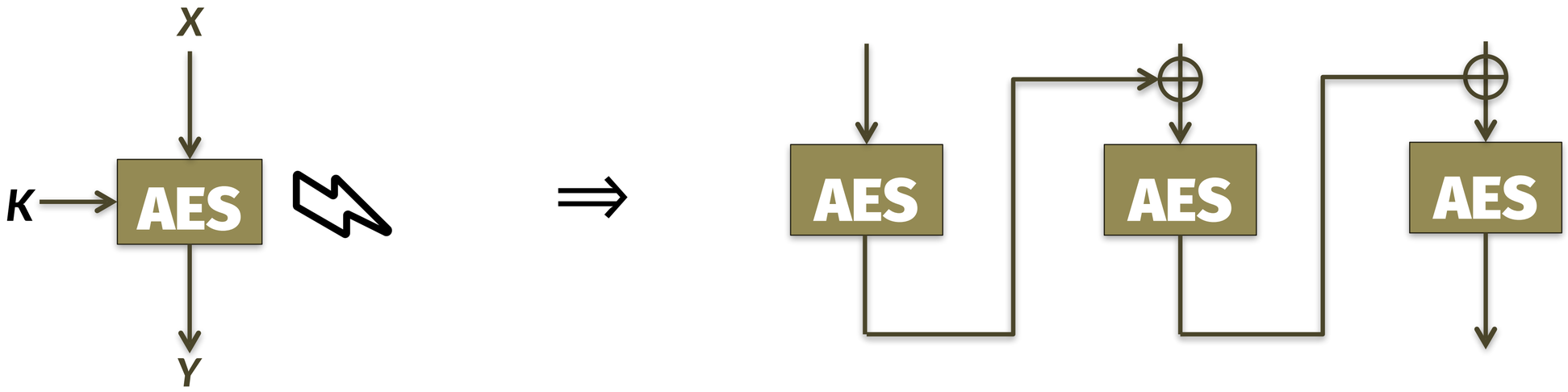
Divide and conquer (for crypto)

- Approach
  - Break 1 byte of the message or key at a time
  - For each byte: guess all 256 values and check which works
  - (Think: how you see crypto broken in any Hollywood movie)
- Effectiveness
  - Shouldn’t this fail? After all, cipher algorithms are diffusing
  - Implementations can betray this goal!



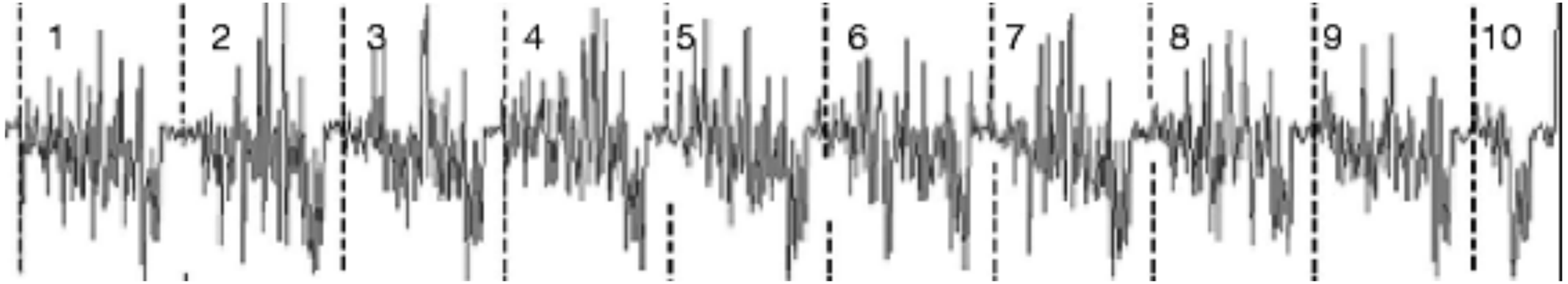
Today: power side channels on block ciphers like AES

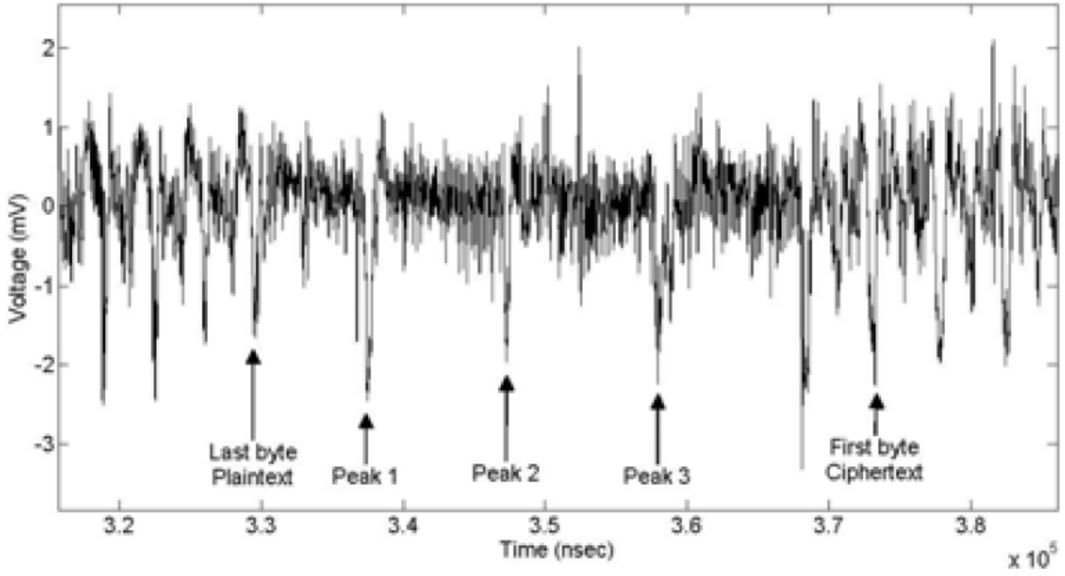
- Plan: use AES inputs + power traces to find the key
  - We must use the power traces in an ‘interesting’ way, since AES is pseudorandom when given inputs alone
- Breaking a block cipher ⇒ any mode of operation



Simple power analysis (SPA)

A single power trace can potentially reveal cryptographic information



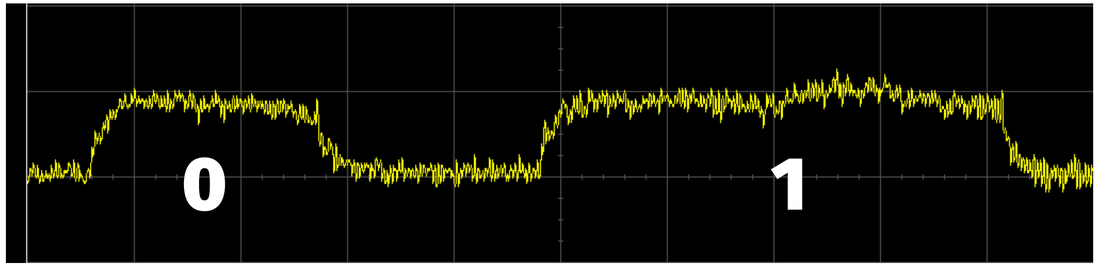


Simple power analysis (SPA)

Power consumption can depend on secret state!

RSA square and multiply (to compute x^k)

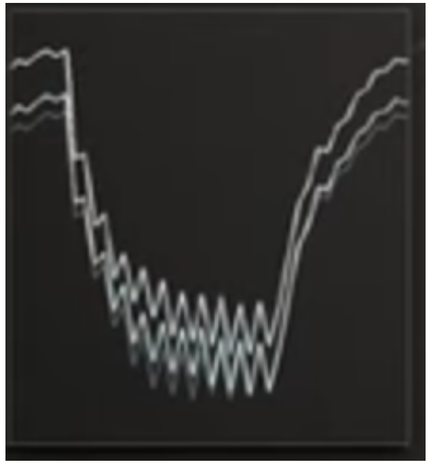
```
x = C
for i = 1 to n
    x = mod(x^2, N)
    if k_i == 1 then          ← Does work conditioned on 1 bit of secret key K
        x = mod(x · C, N)
return x
```

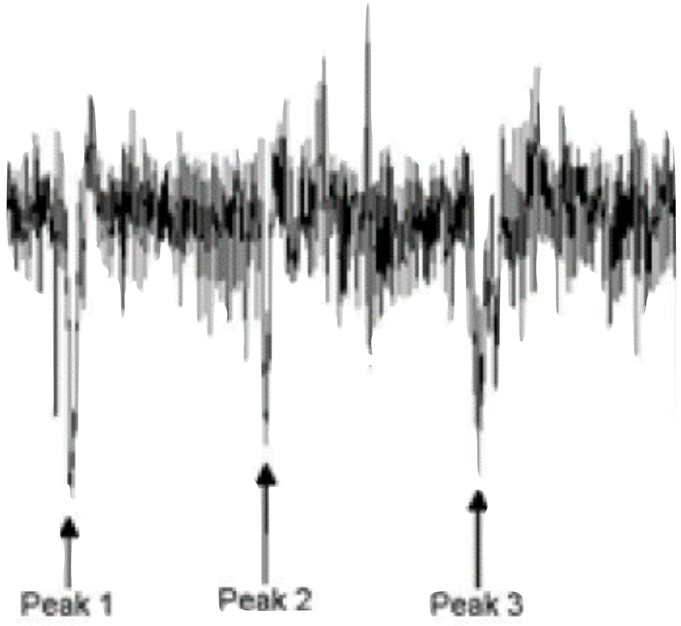


Lesson: never write crypto code that conditions on secret data!

Differential power analysis (DPA)

- Scenario: run a block cipher multiple times with same key, diff message
- Observe slight differences in the power consumed…
  - Between different messages
  - Between rounds for the same message

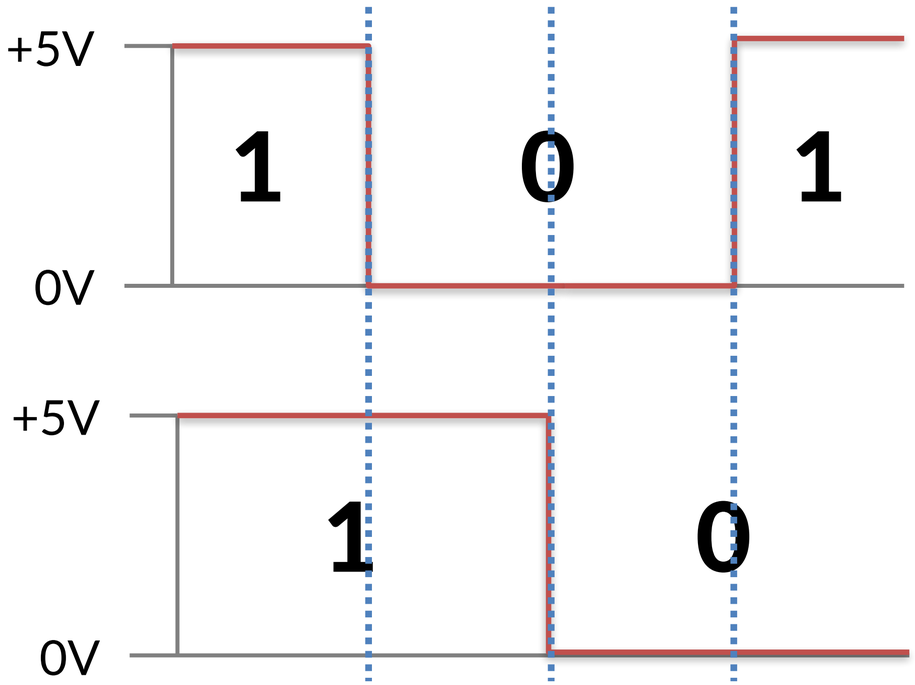




- Difference is data-dependent!
  Can we use this to find the message or key?
- Hmm, what consumes power anyway?

Power consumed in transmission

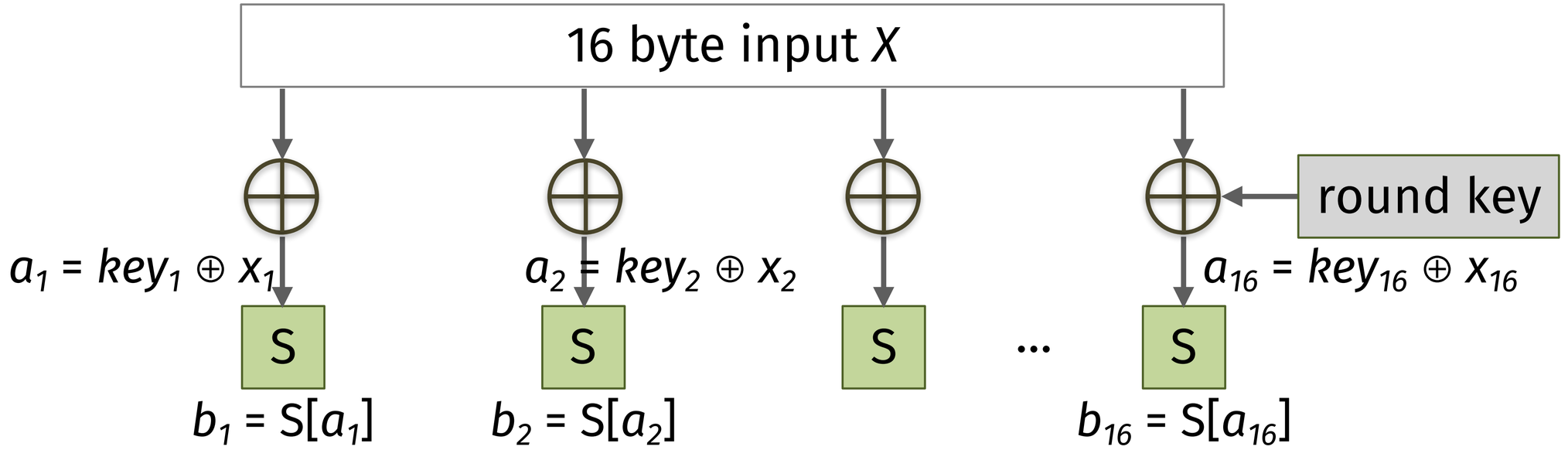
- For each wire on a data bus, store a logical 0/1 as the voltage of the wire
- Power consumed ~ Hamming weight



Power consumed in storage

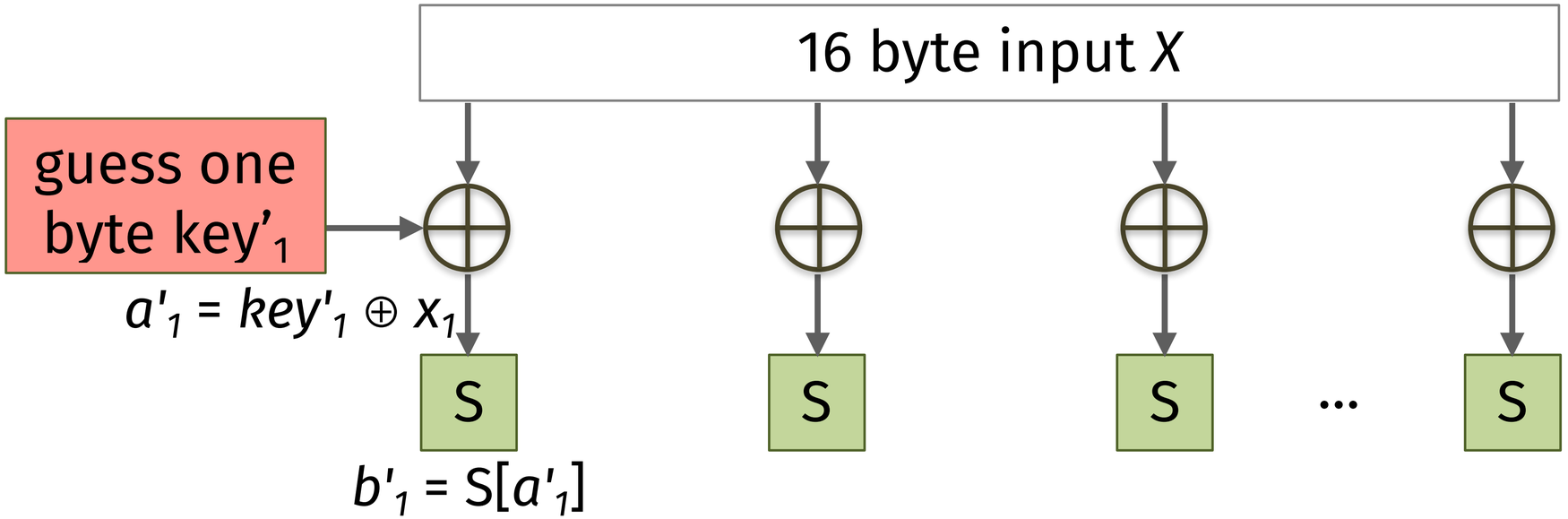
- Designed only to consume power during transitions 0 → 1 or 1 → 0
- Power ~ Hamming distance

Simplified picture of first round of AES



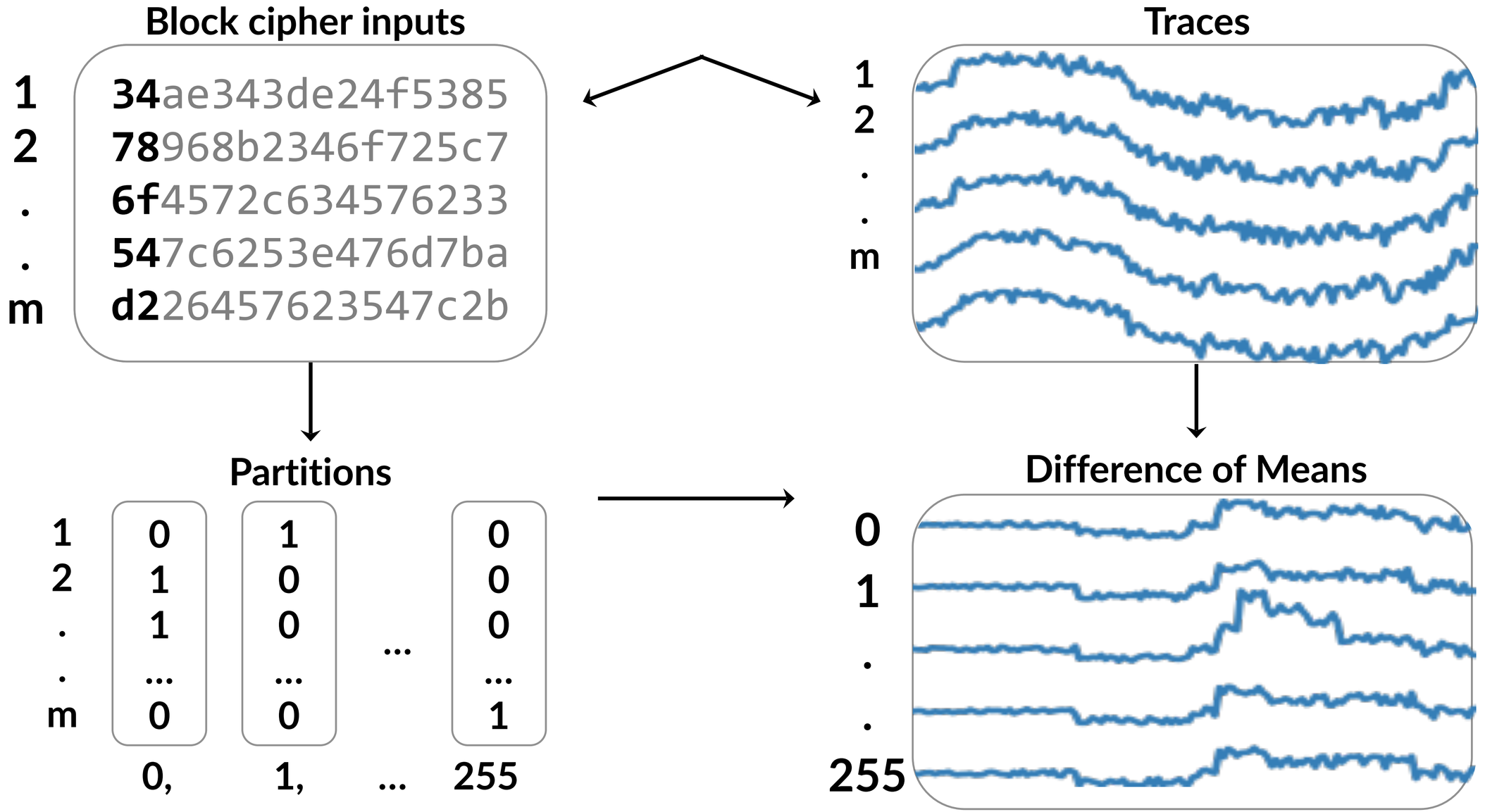
- The first round of AES begins with xor of key followed by S-box lookups
- Let *A* and *B* denote the 16-byte state before/after the first round S-box
- Claim: if Eve knows input *X* along with *A* (or *B*), she can recover the secret key

Attack on AES



- Obtain an input X (or output Y), and guess one byte of the key
- Compute the resulting byte of intermediate values *A'* and *B'*
- Hope that HW(*A'*), HW(*B'*), or HW(*A'* ⊕ *B'*) is correlated with power ?

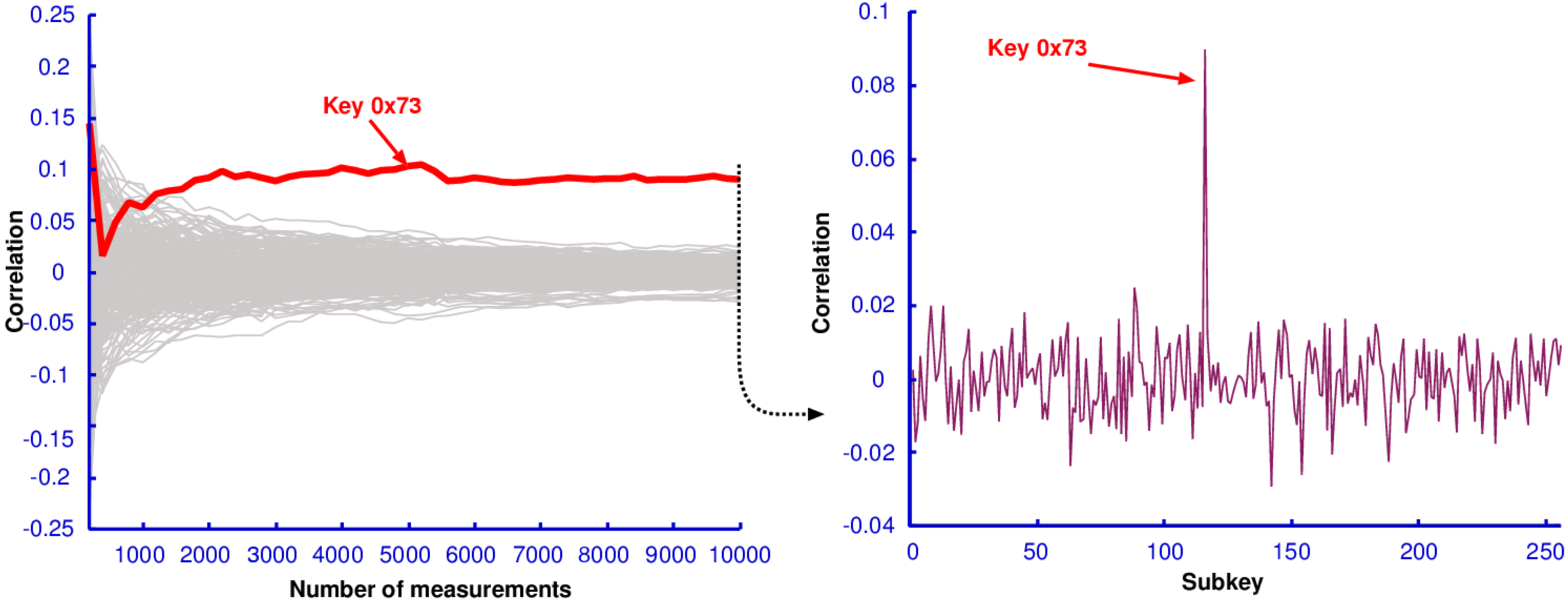
Differential Power Analysis (DPA)



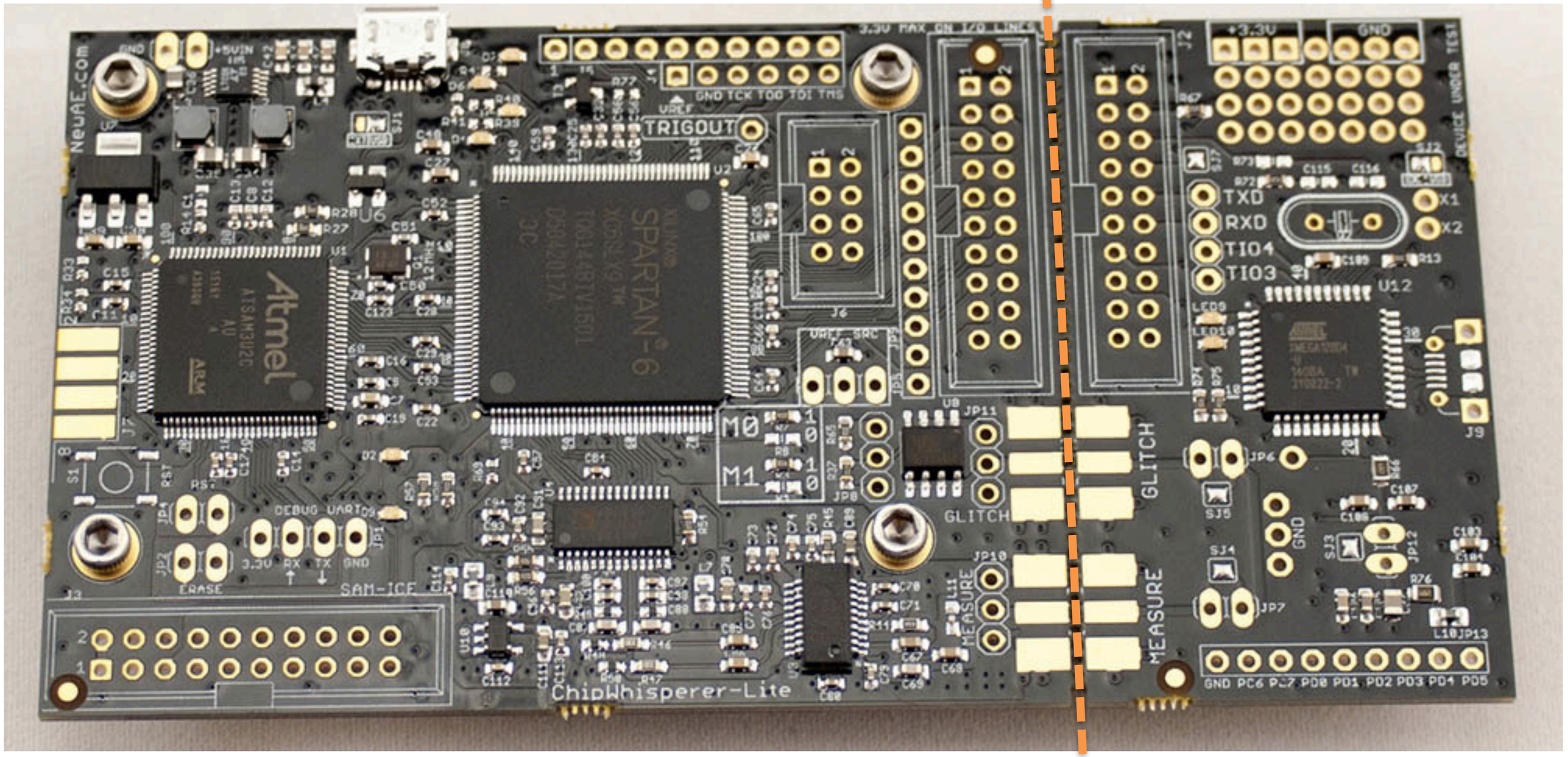
Kocher, Jaffe, and Jun, “Differential power analysis,” CRYPTO 1999.

DPA Example

Note: correlation of incorrect keys fades quickly with additional samples



Let’s see this in action ourselves



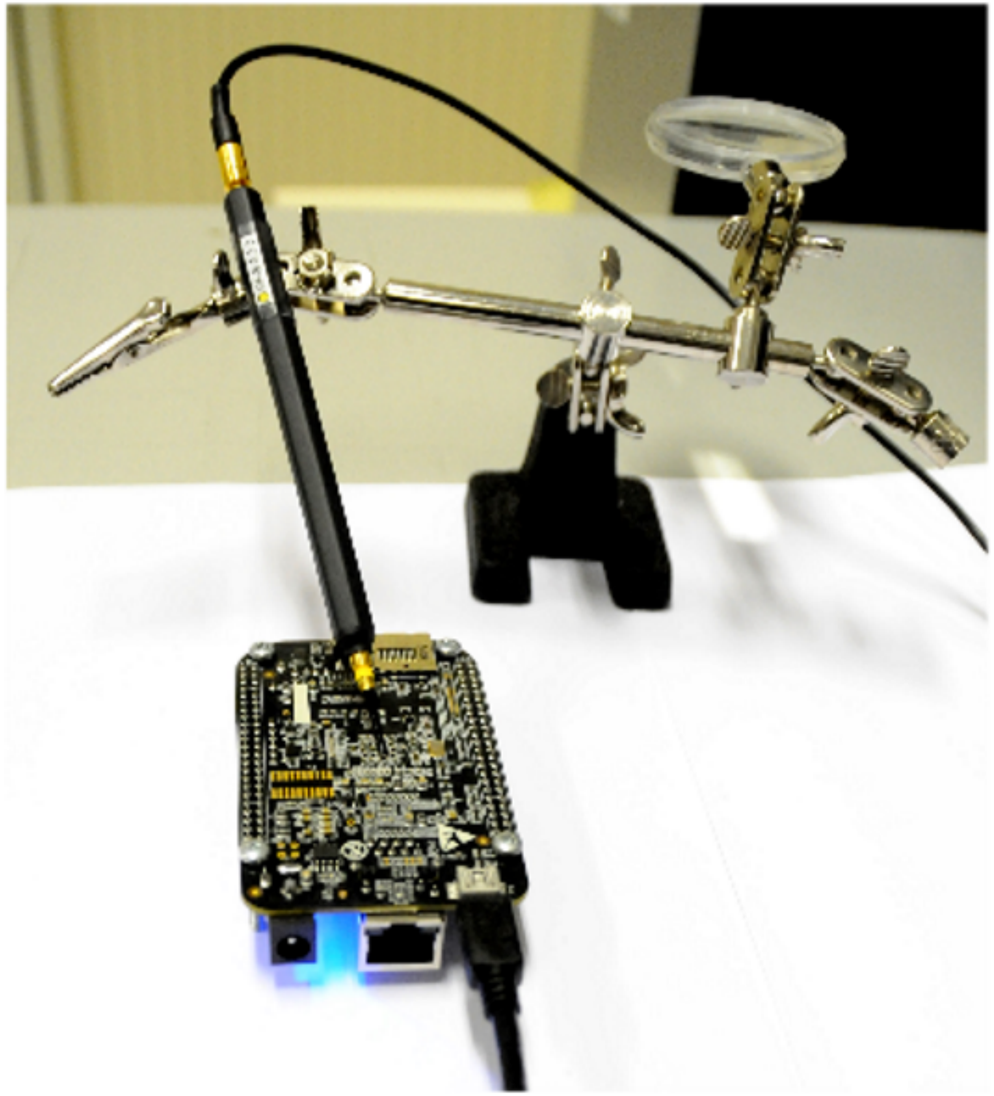
Attacker: oscilloscope to measure power

Target victim: FPGA that runs AES

Can Eve observe Alice’s power source?

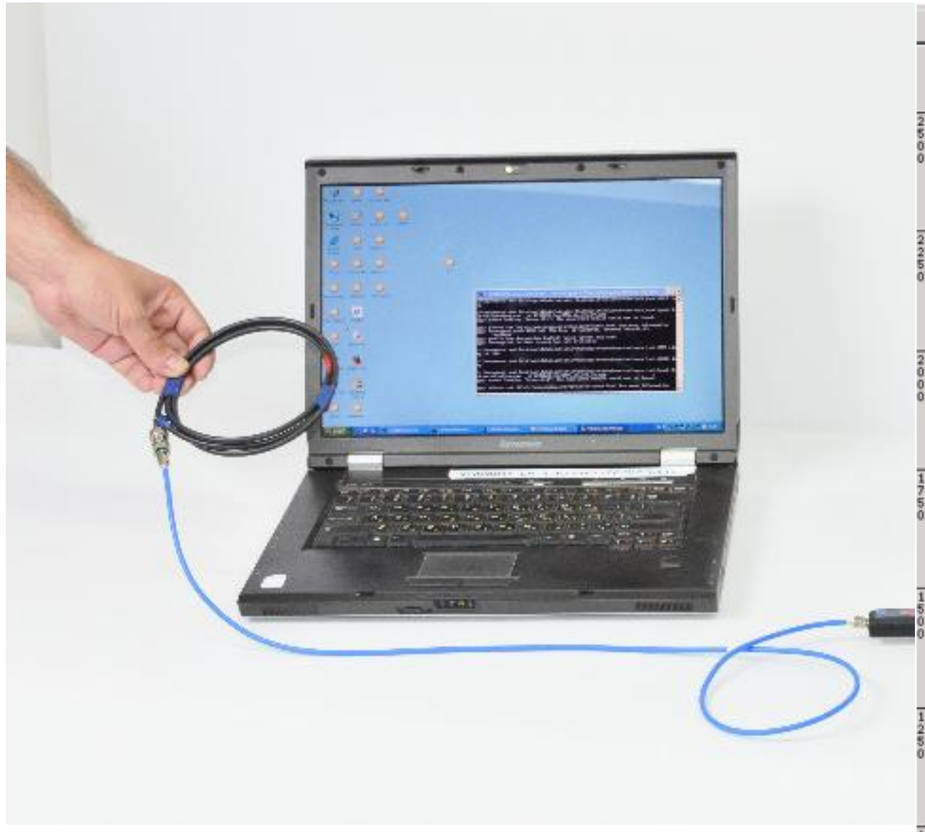
- Smartcards like your credit card or CharlieCard are passively powered by an external reader
- In cloud computing, the power is already monitored by the cloud provider

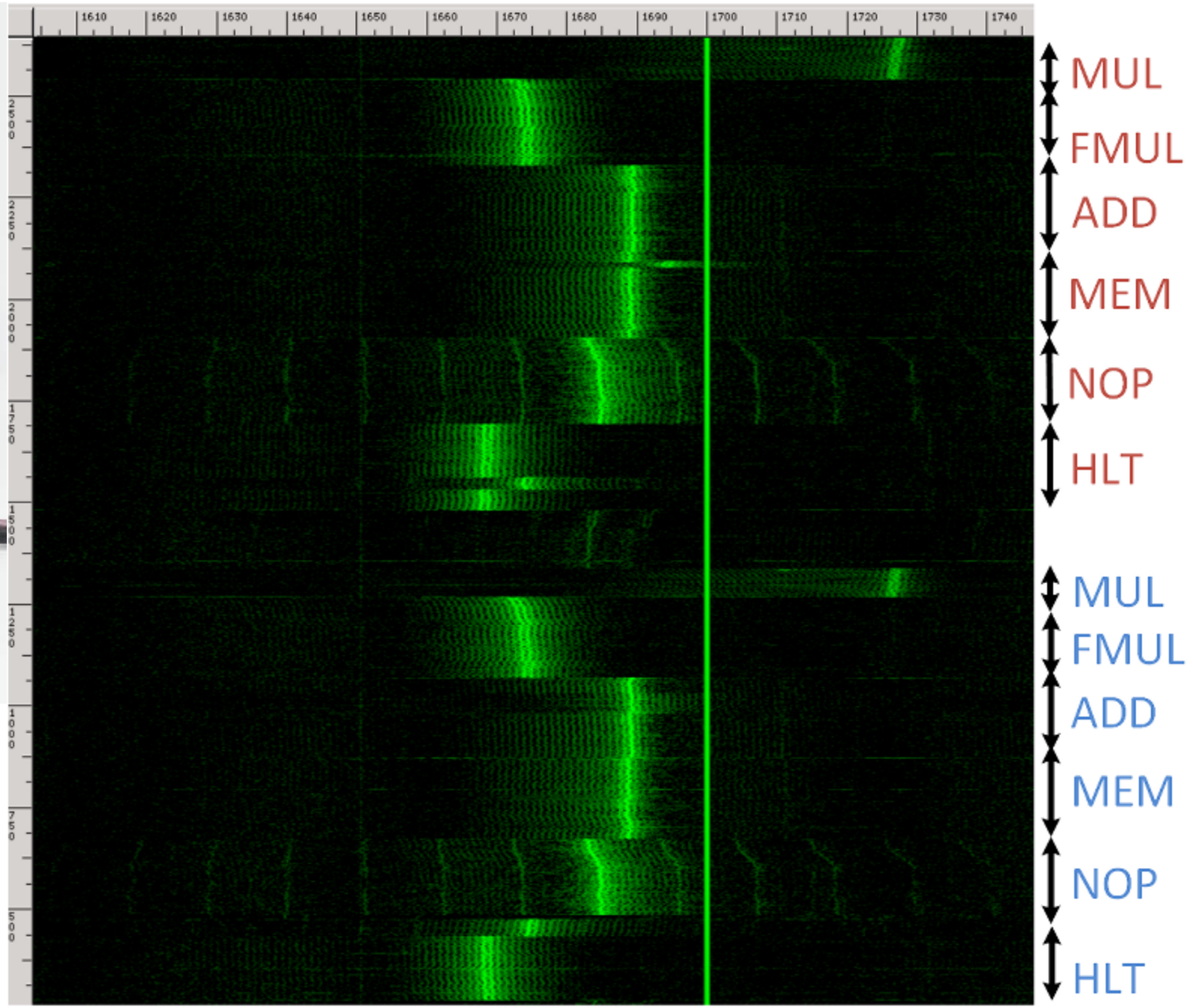
Alternatives: Electromagnetic probes



- Obtain data “similar” to power traces
- Can localize measurement to the unit performing crypto within a circuit board

Alternatives: Electromagnetic probes

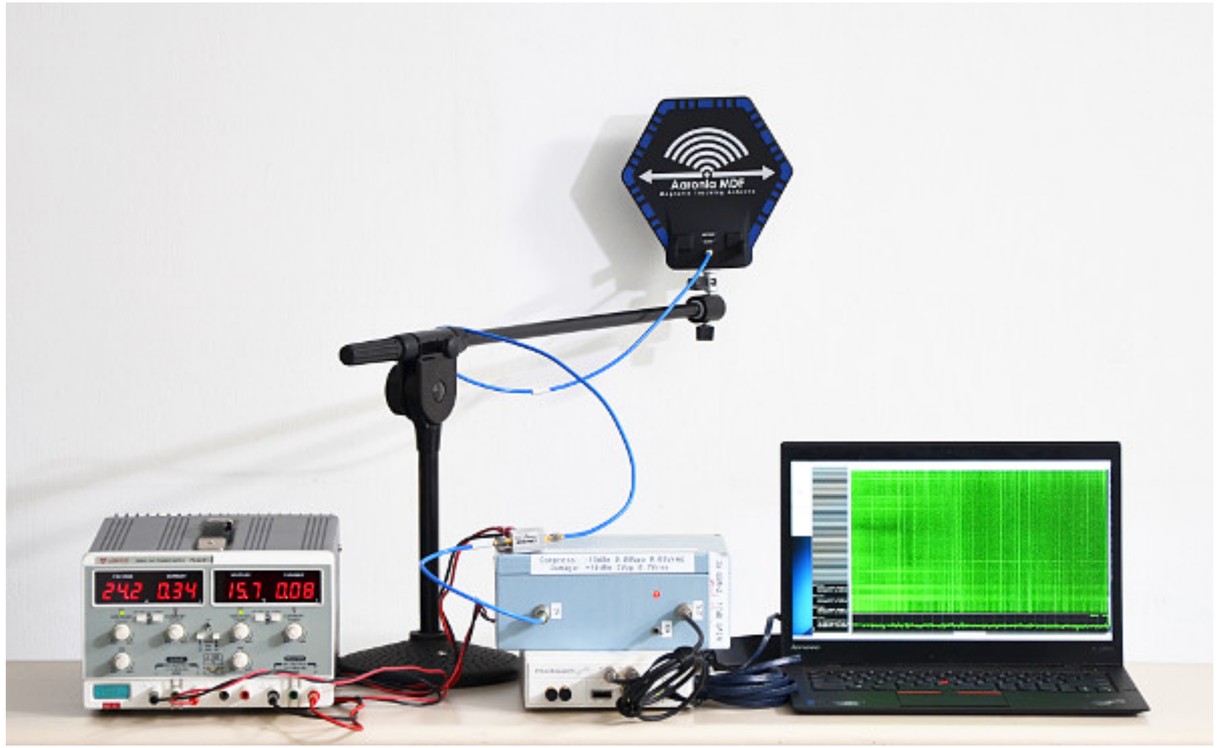




In the public key setting, even a low-frequency probe will do.

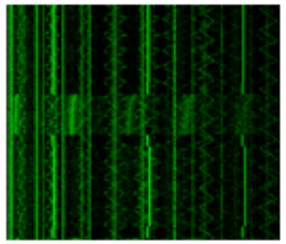
Can “see” differences in CPU instructions.

Alternatives: Electromagnetic probes



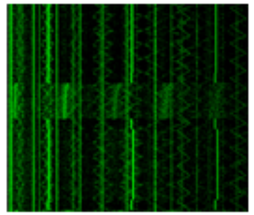
Can target a victim from a distance!

Alternatives: Chassis potential



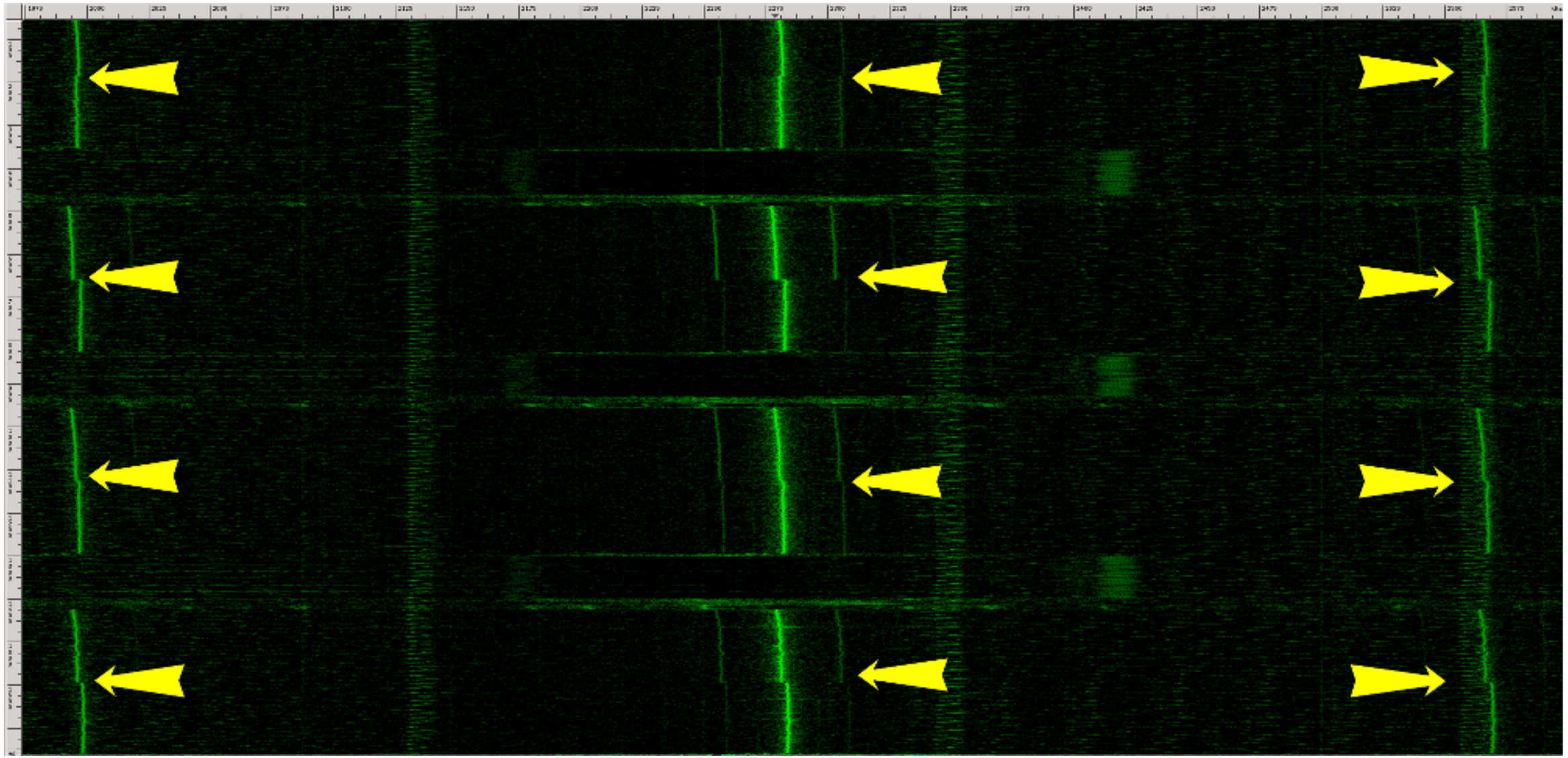
Key = 1110111011…

Alternatives: Chassis potential

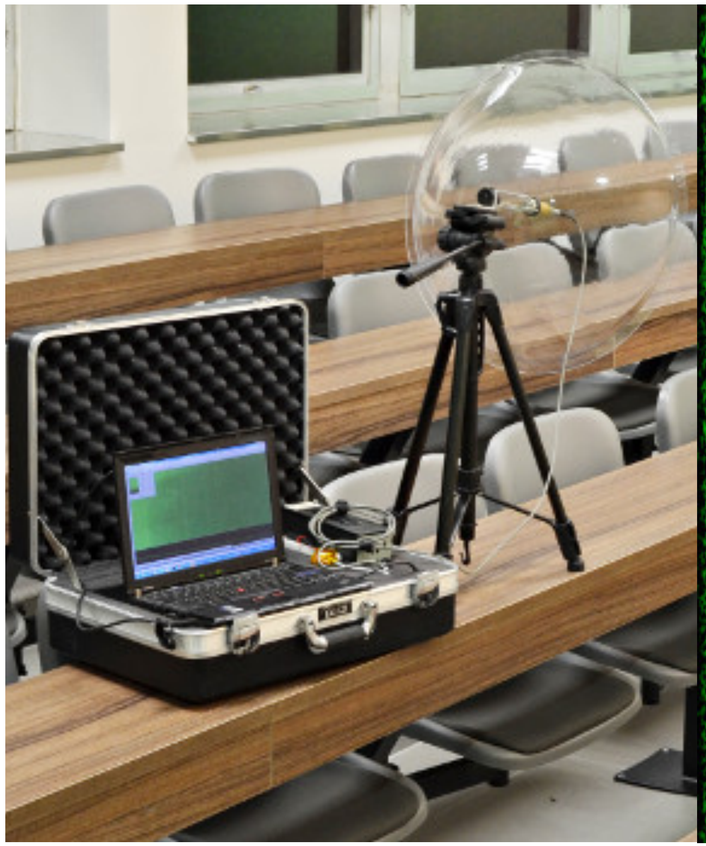


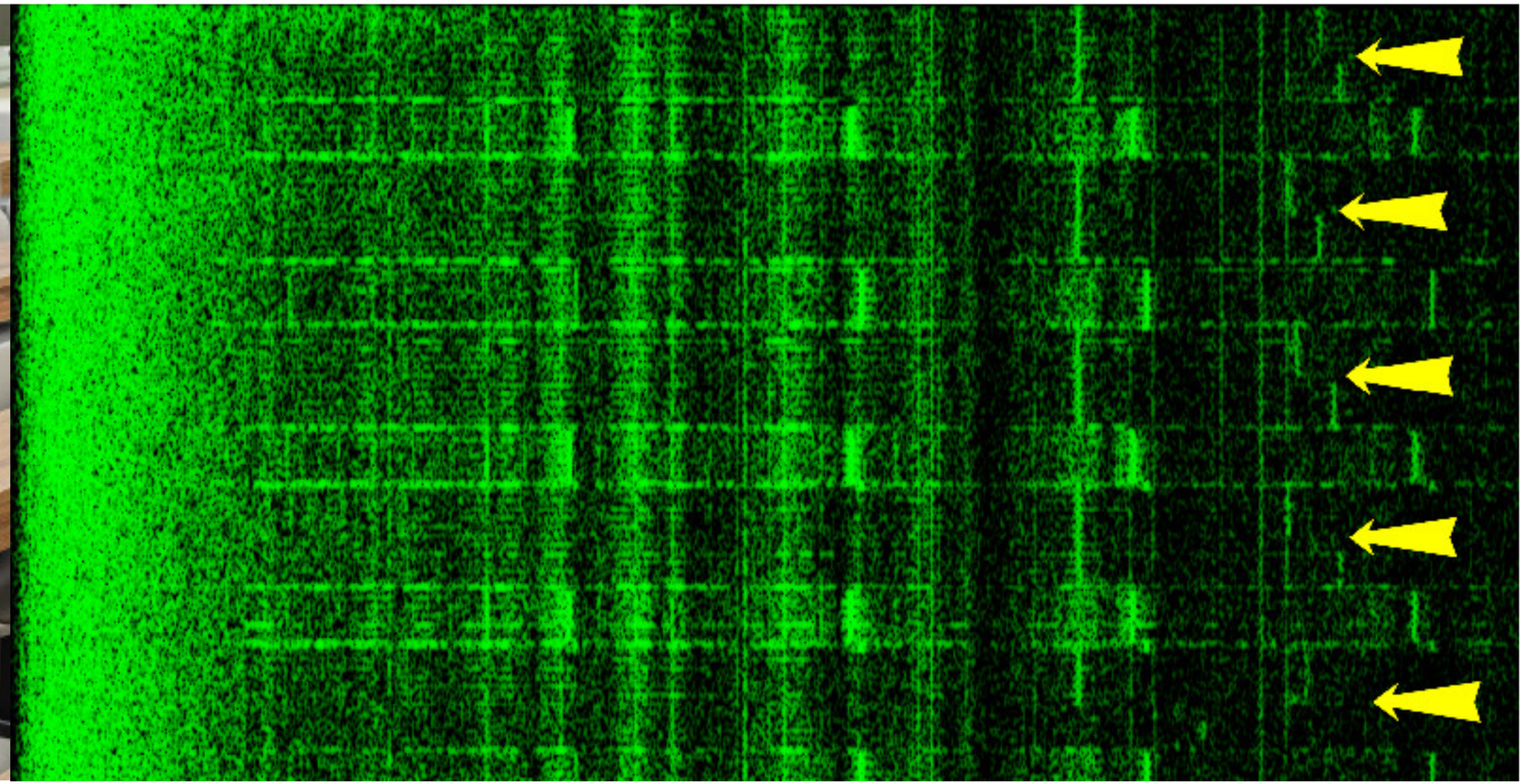
Key = 1110111011…

Alternatives: Chassis potential



Alternatives: Sound

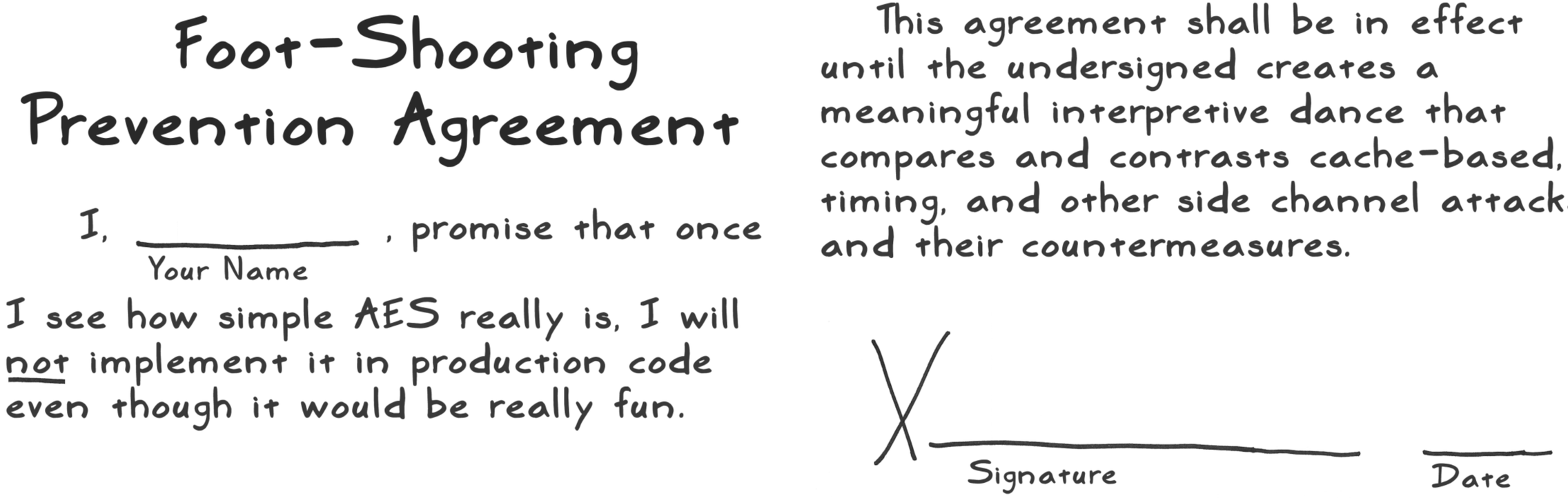




Countermeasures to avoid power analysis?

- Eliminate Mallory's ability to see the power signal
  - *Shielding:* Physically enclose system so emanations cannot be captured
  - *WDDL:* For every 0 → 1 transformation, perform a mirror op 1 → 0
  - *ECC:* Perform ops directly over a const-weight error correcting code of data
- Eliminate Mallory's ability to make sense of the power signal
  - *Masking:* Split circuitry into pieces that can be recombined to construct output
  - *Variety:* Don't have just one S-box, but rather several so that x is unknown (chosen from a public set of S-boxes as per Kerckhoffs' principle)
  - …

Side channels ⇒ difficult to implement crypto securely



Source:
moserware.com/2009/09/stick-figure-guide-to-advanced.html