TROOPERS

Hardware Side Channel Attacks
.. on the cheapiest!

Albert Spruyt
Alyssa Milburn

About Us

Alyssa

PhD student

@noopwafel

Albert

Unemployed

Today

- Side Channel Analysis
- (Cheap) Hardware
- Demos!
- A bit of Fault Injection

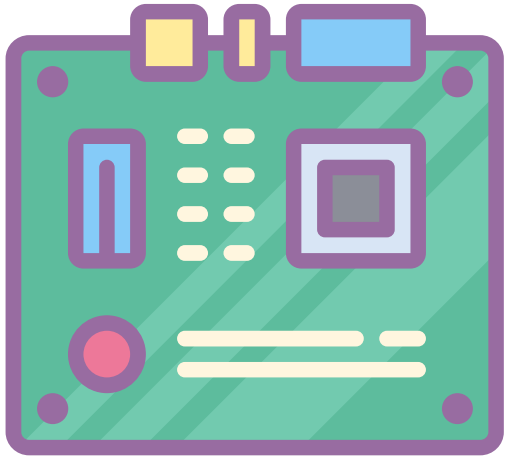
Side Channel Analysis is full of...

- Scary math
- Complicated words

- You can talk to us later to learn more!

High-level overview

Commands



- **Attackers need:**
  - Physical access
  - Some input (or output)

The context

- **Smartcards**
  - credit cards, access cards, passports
- **Secure microcontrollers**
  - crypto wallets, U2F/YubiKey
- **Random IoT devices**
  - lightbulbs, ...

Why care about side channels?

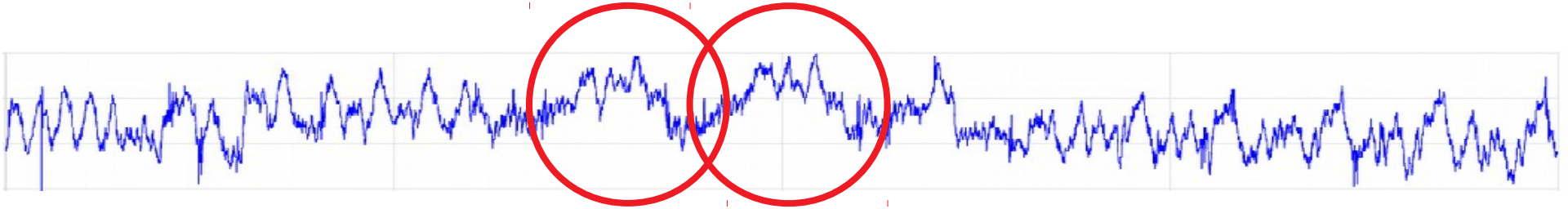
PIN check

```
for (n: 1 → 4)

  if (secret_pin[n]
      != input[n])

    fail();
```

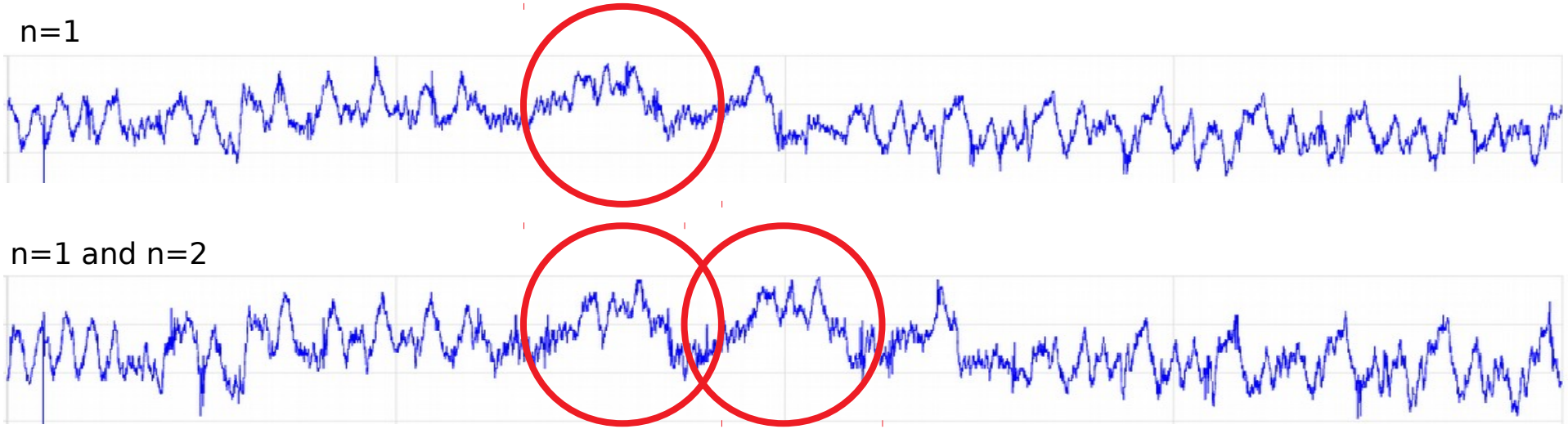
PIN check



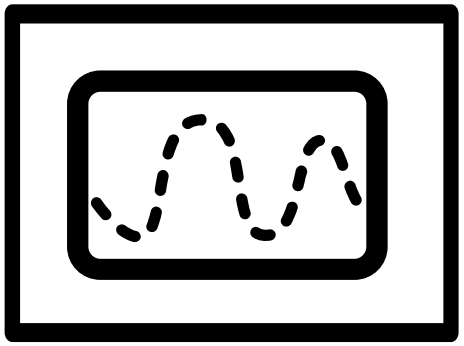
```
if (secret_pin[n]
    != input[n])

  fail();
```

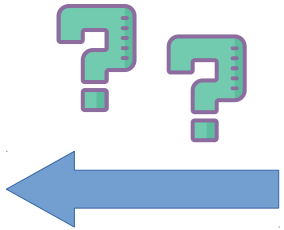
PIN check

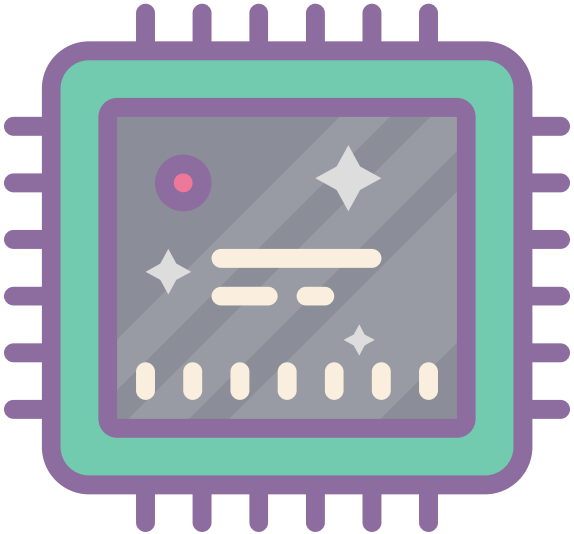


How do we measure power?

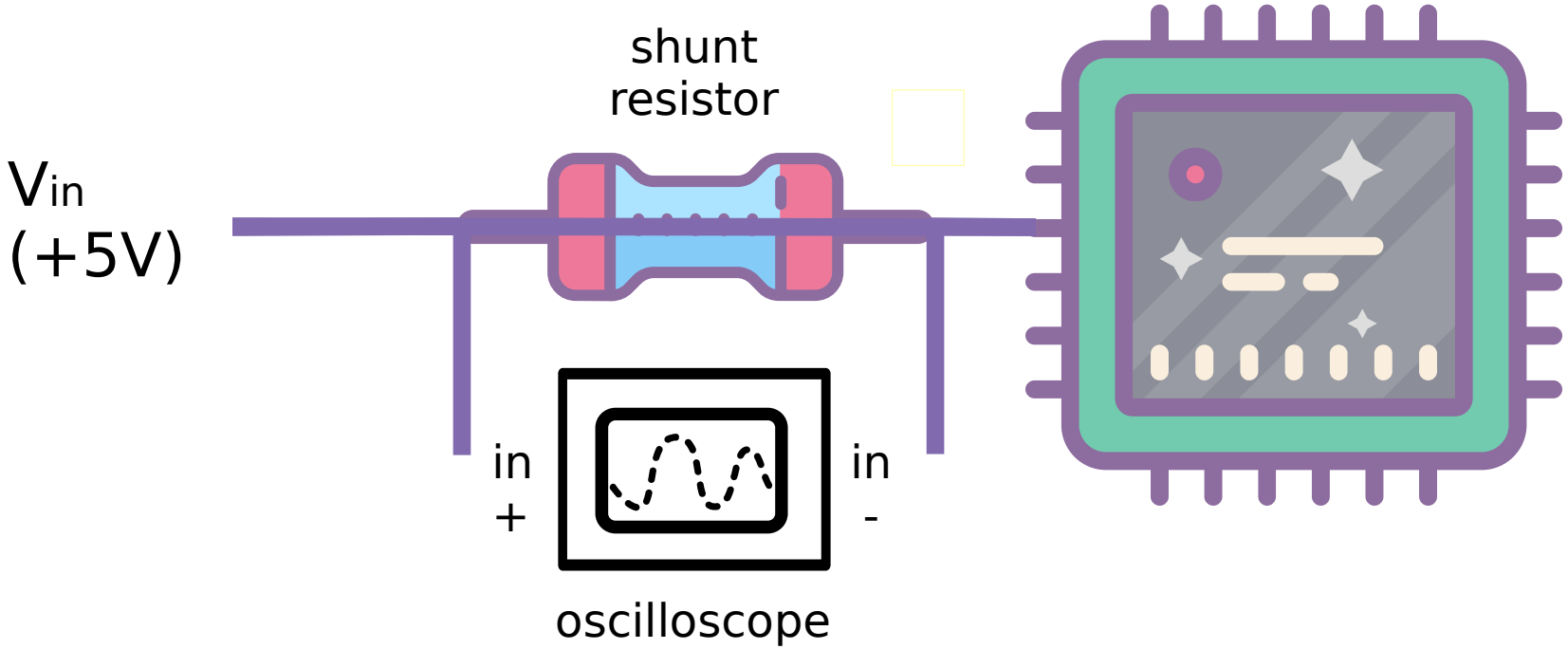


Oscilloscope





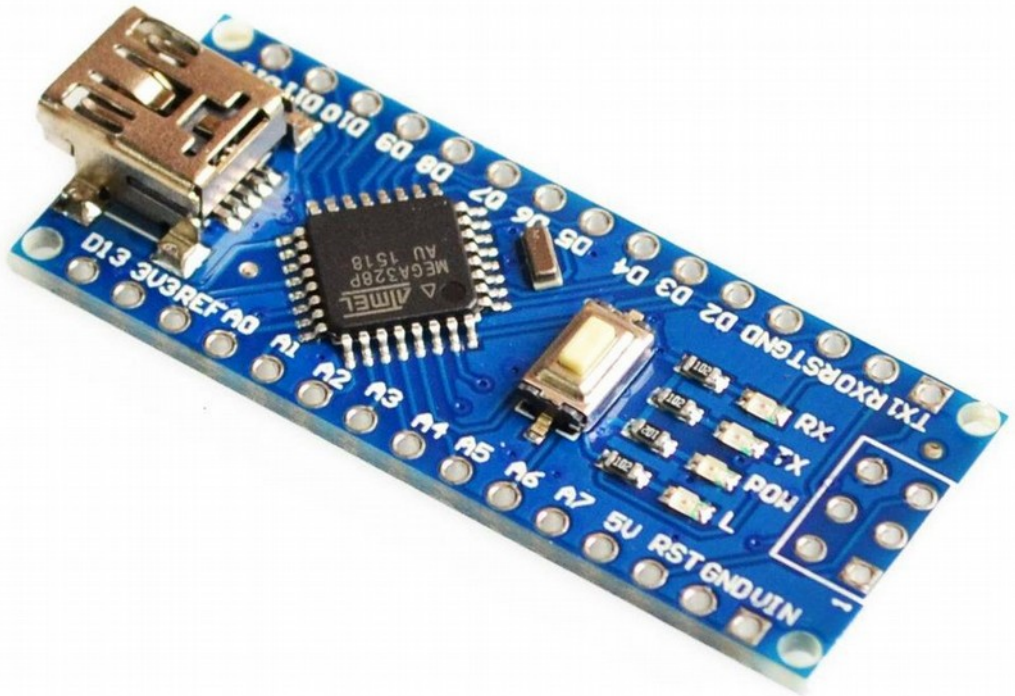
Power cut!



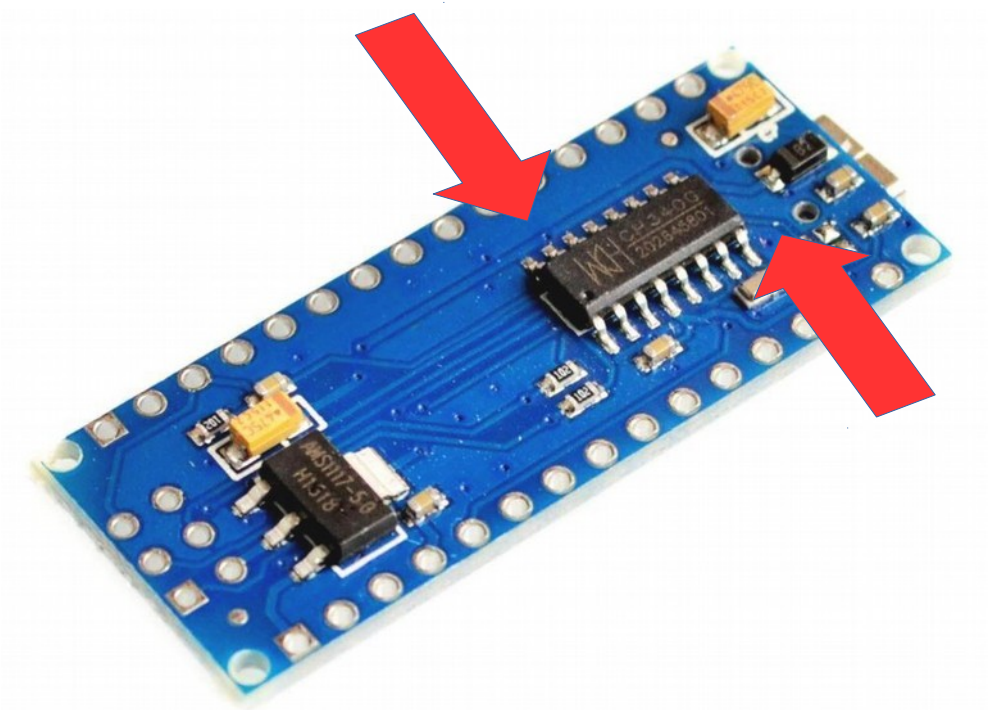
Today’s target

Arduino Nano

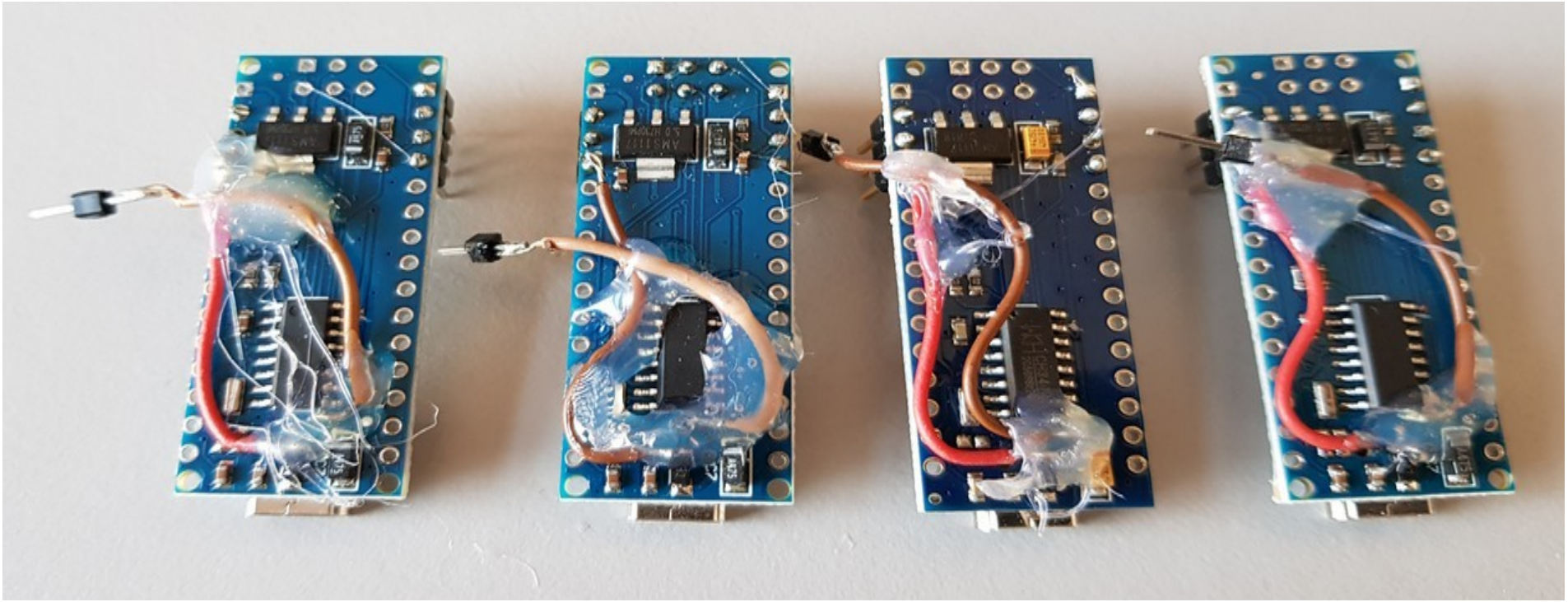
- 16 Mhz
- ~3-5 euro
- Not secure



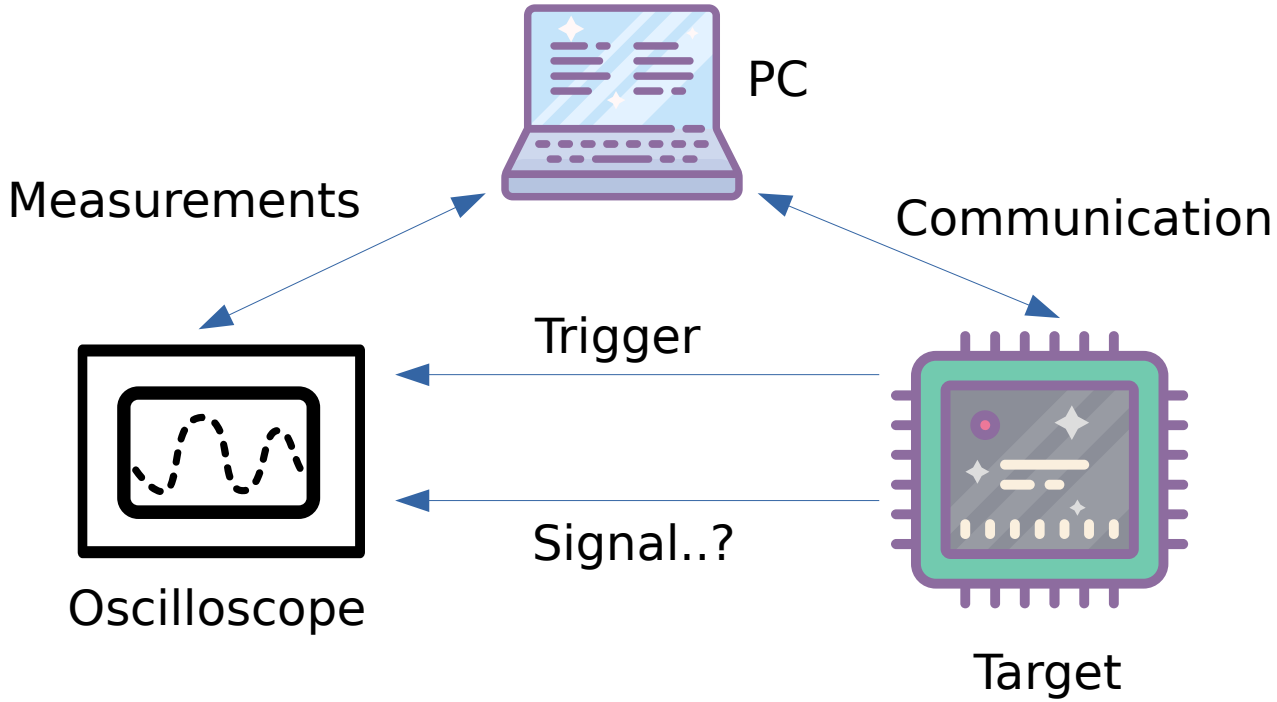
Real power cuts



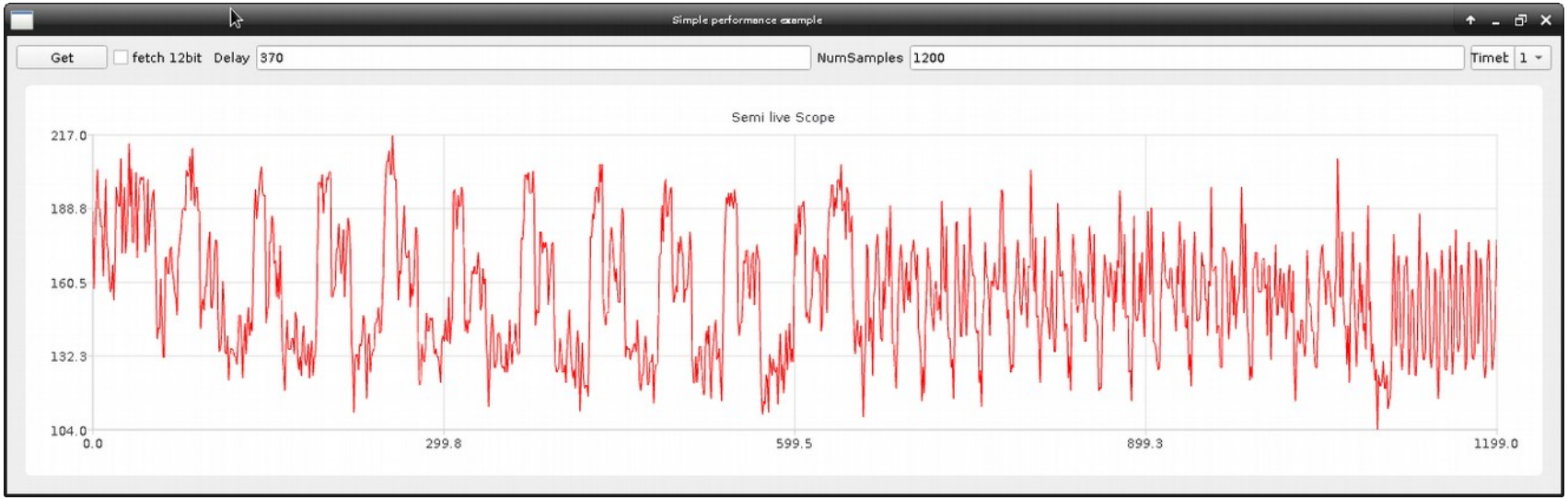
Real power cuts



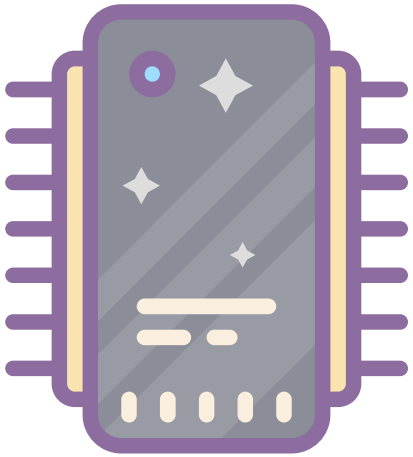
Big picture


PC
Measurements
Communication
Trigger
Signal..?
Oscilloscope
Target

DEMO


Simple performance example
Get
fetch 12bit
Delay
370
NumSamples
1200
Timet
1
Semi live Scope
217.0
188.8
160.5
132.3
104.0
0.0
299.8
599.5
899.3
1199.0

What’s going on?



1 / 0 / 1 / 0

Hamming weight

0x00: 00000000

→ hamming weight 0

0x05: 00000101

→ hamming weight 2

0xFF: 11111111

→ hamming weight 8

0x11: 00010001

→ hamming weight 2

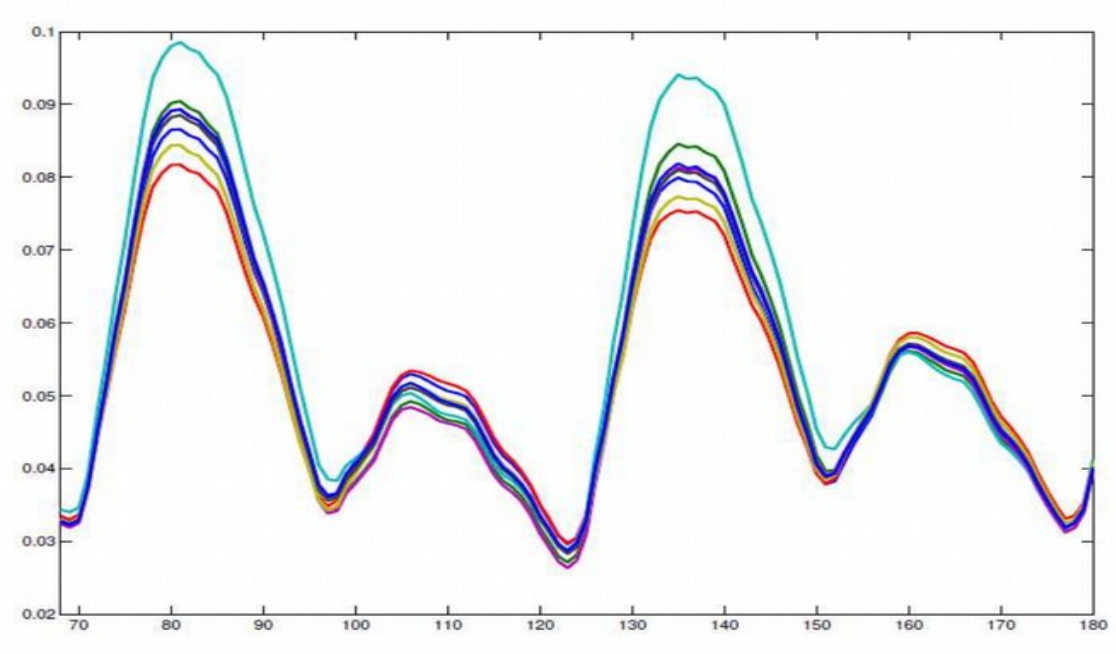
Leakage?

- Calculate with (random) bytes
- Take power traces

Can we match the **Hamming Weight** of the byte to the power traces?

Hamming weight

- **Power profiles based on different data**
- **Averaged**
- **We can see the data being processed!**



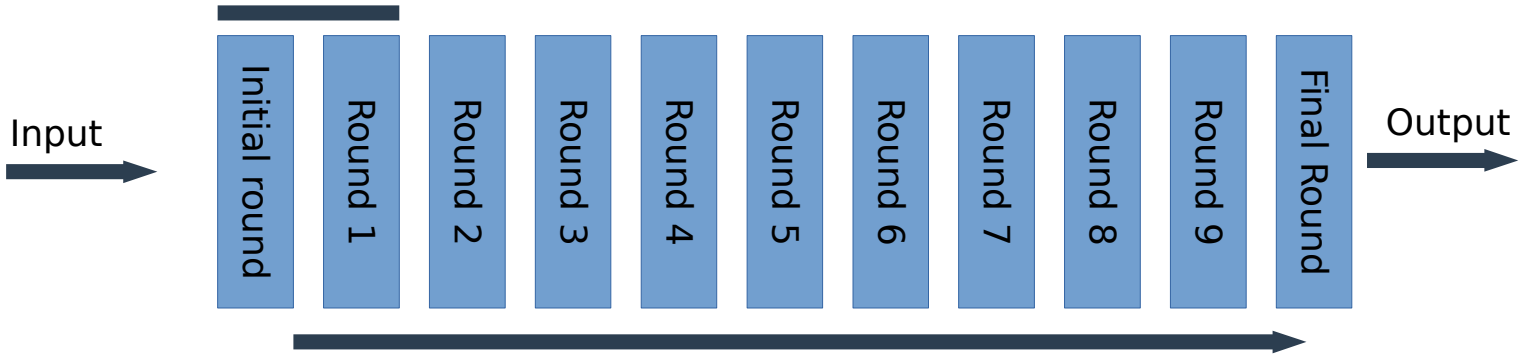
Source: Side channel analysis, practice and a bit of theory. Ilya Kizhvatov

Keys

- **We’re going to steal encryption keys**
  - Everyone needs to have keys

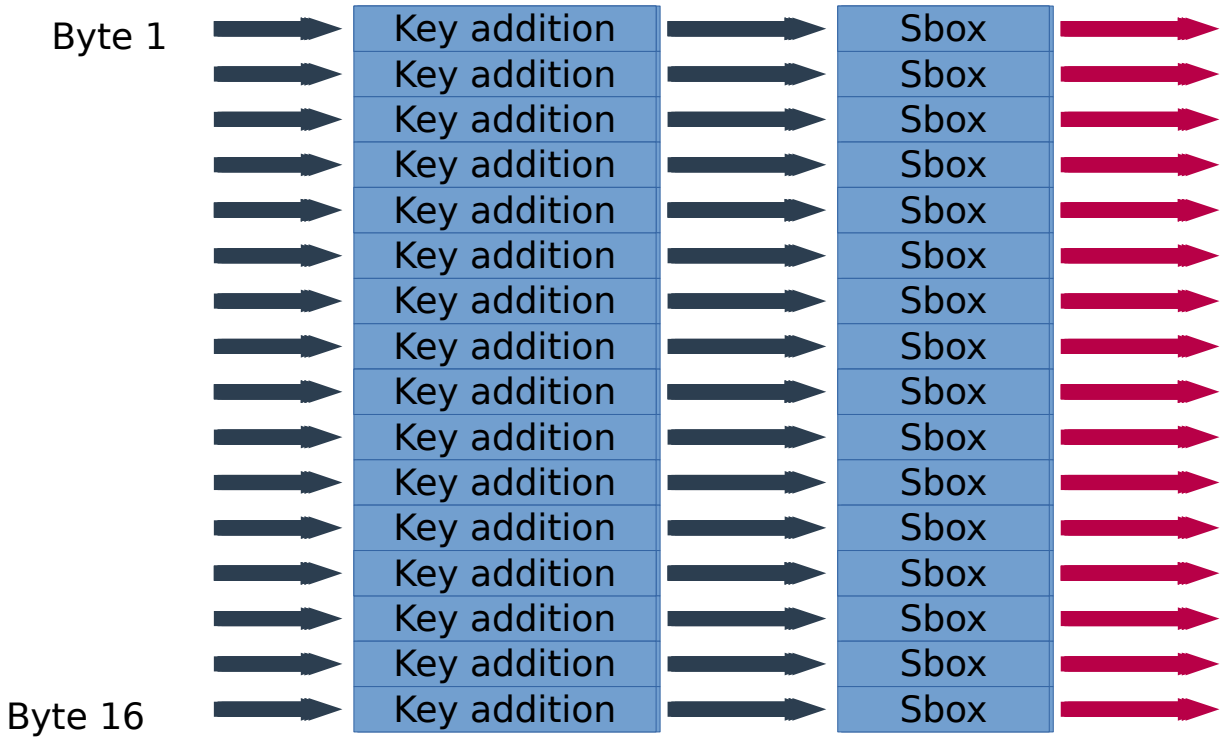
Super secure encryption

- **AES-128: unbroken and secure**



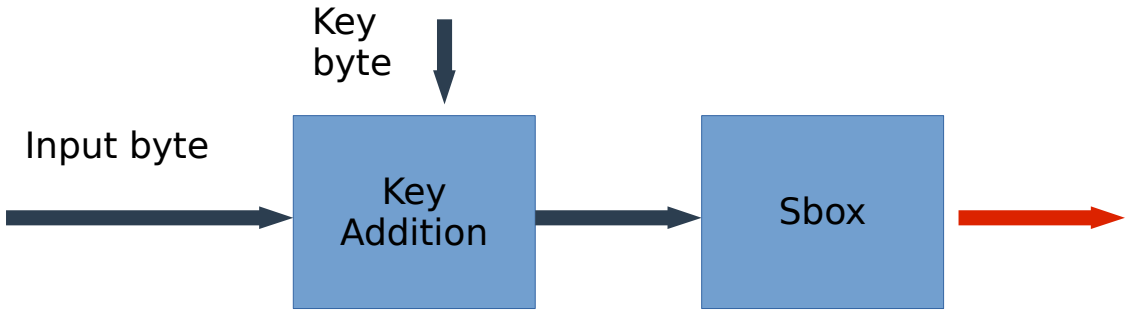
AES trace

Early AES



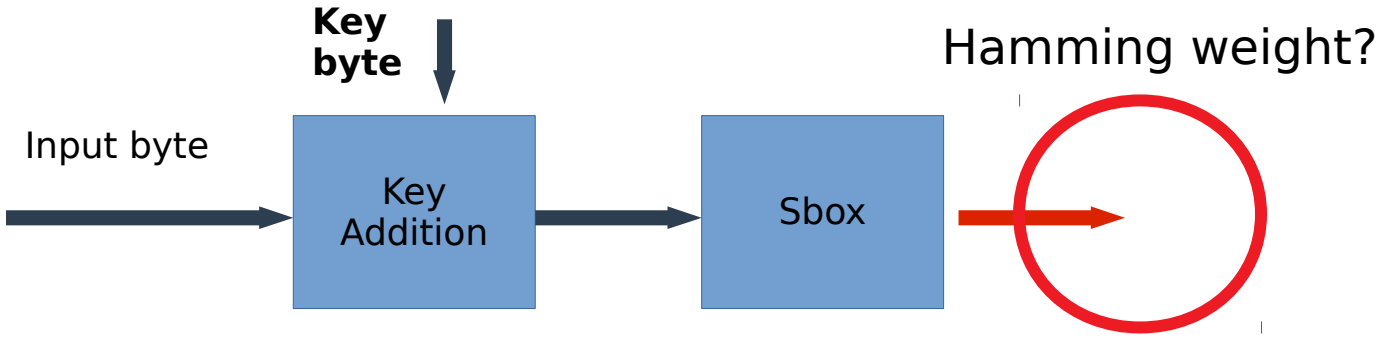
Single byte only

- **We can look at each byte separately!**



Single byte only

- **Only 256 possible key bytes**
  - Try them all!



The master plan

- **For every key guess:**
  - For each input:
    - Calculate **Hamming Weight** after the S-box
    - Compare that with the actual **leakage**

- **Pick the guess with the best fit!**

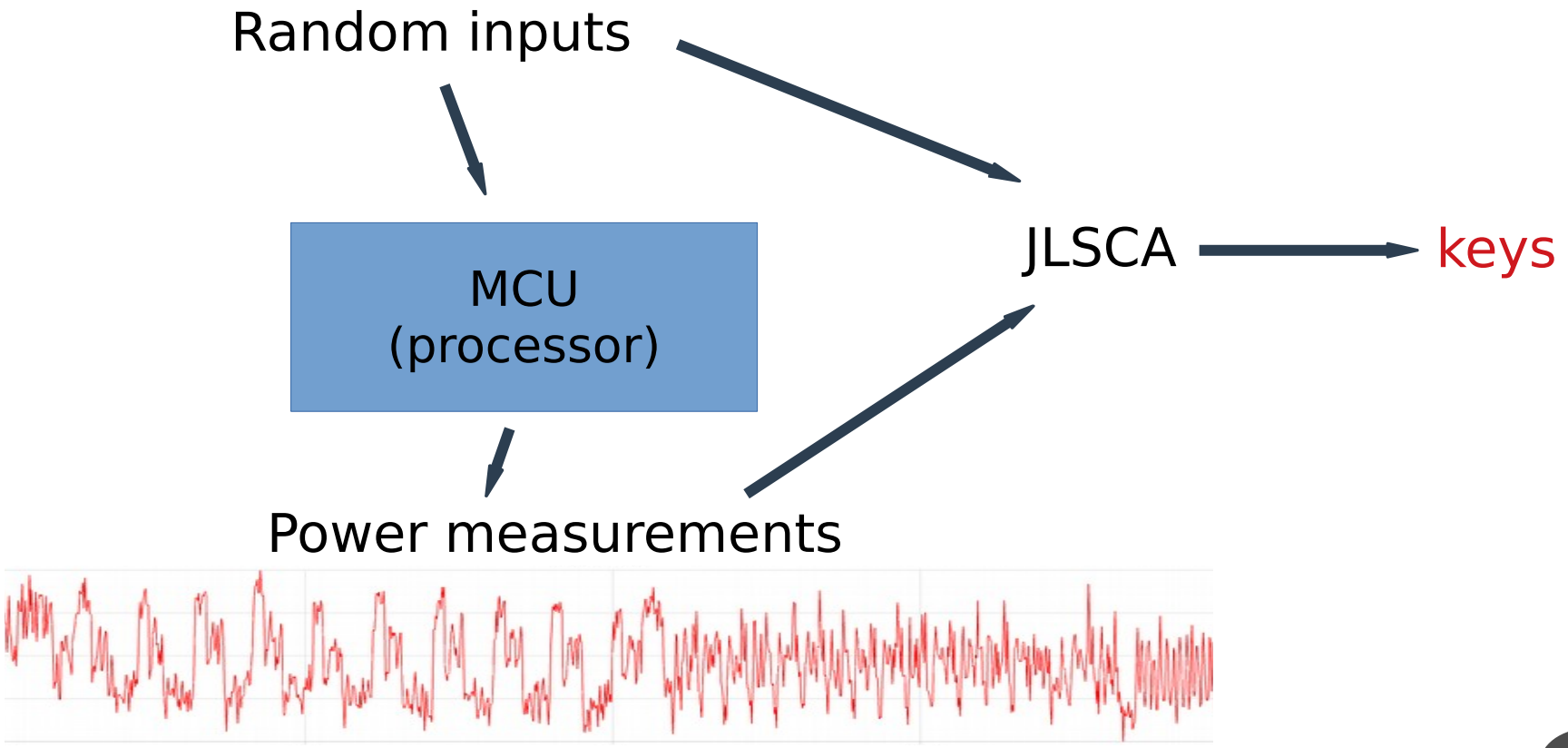
Correlation
Power
Analysis

We’re lazy!

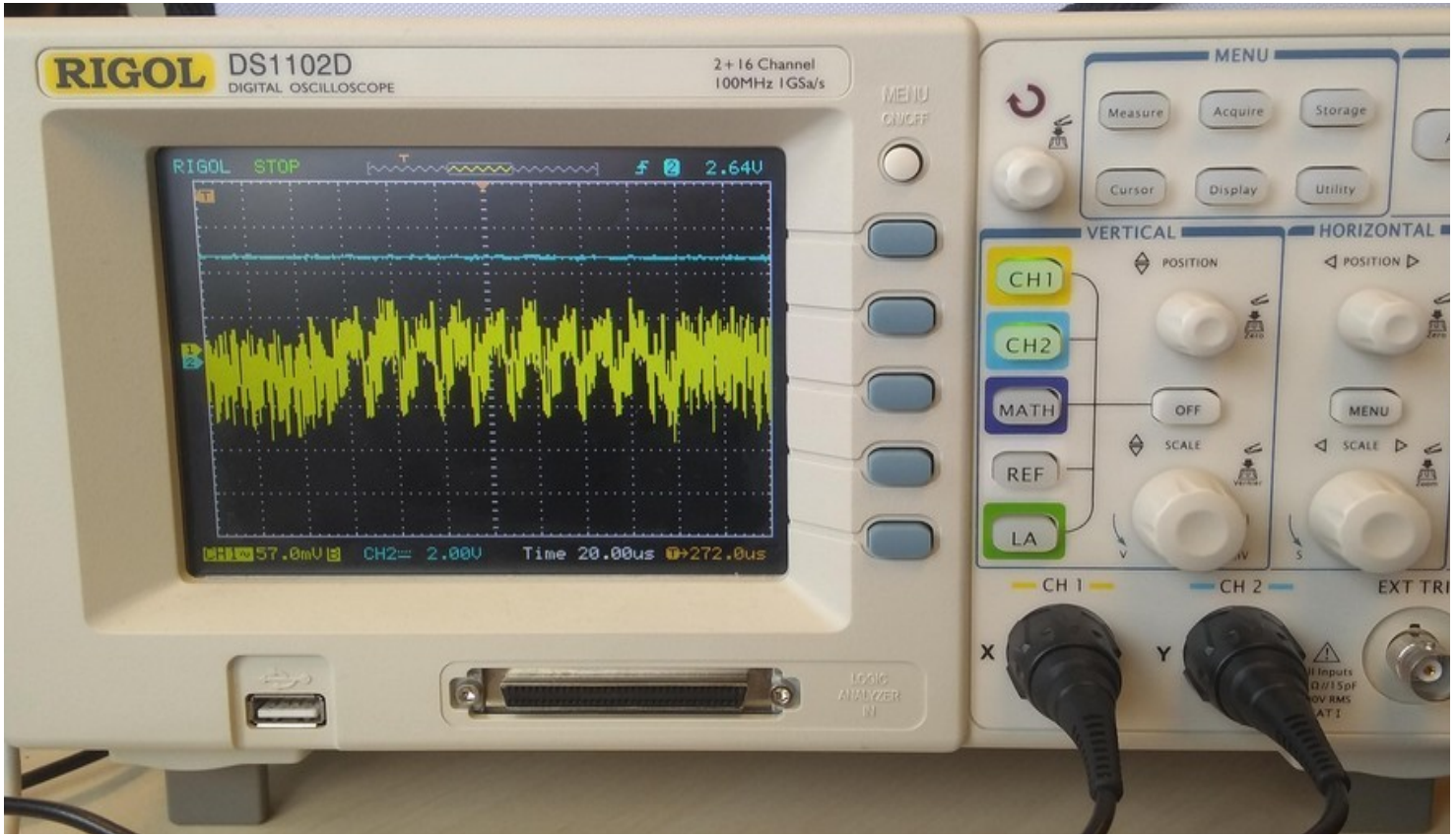
- **Open source: JLSCA**
  - Does CPA for us
  - Also supports fancier attacks
  - Runs fast on a cheap laptop

  (Thanks Cees!)

New plan


Random inputs
MCU
(processor)
JLSCA
keys
Power measurements

Oscilloscope?



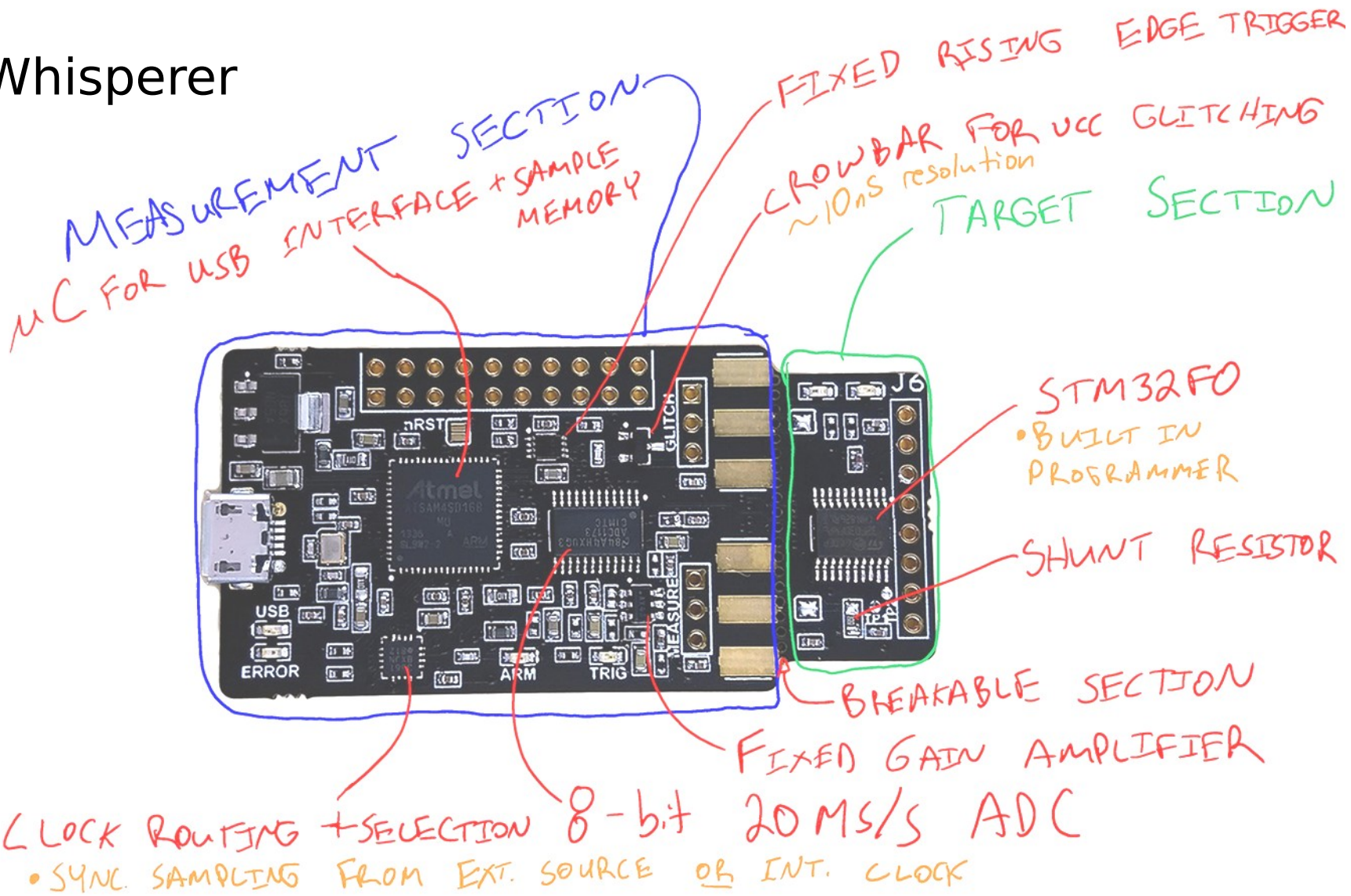
Oscilloscopes

LeCroy WaveRunner	$17 000
PicoScope 3406D	$2 500
Rigol DS1054Z	$500

Previous Work

ChipWhisperer	$250
Hantek USB oscilloscope	$60
ChipWhisperer Nano..?	$50

ChipWhisperer Nano



Solving our hardware woes

Let’s build an awesome, cheap scope!

Let’s hack something together!

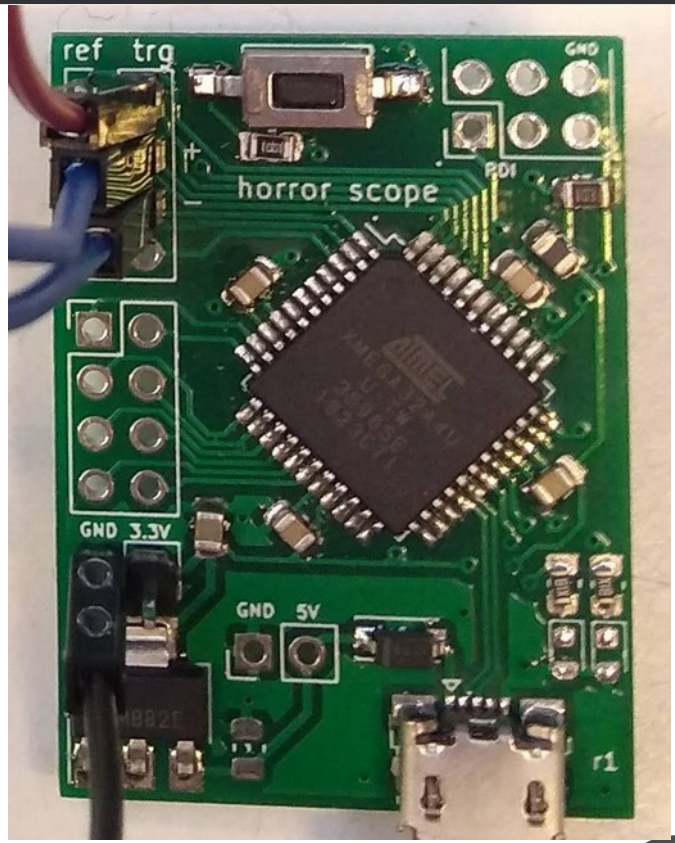
What do we need?

- GPIO to trigger
- ADC to measure
- Memory to store measurements

HorrorScope

Atmel XMEGA

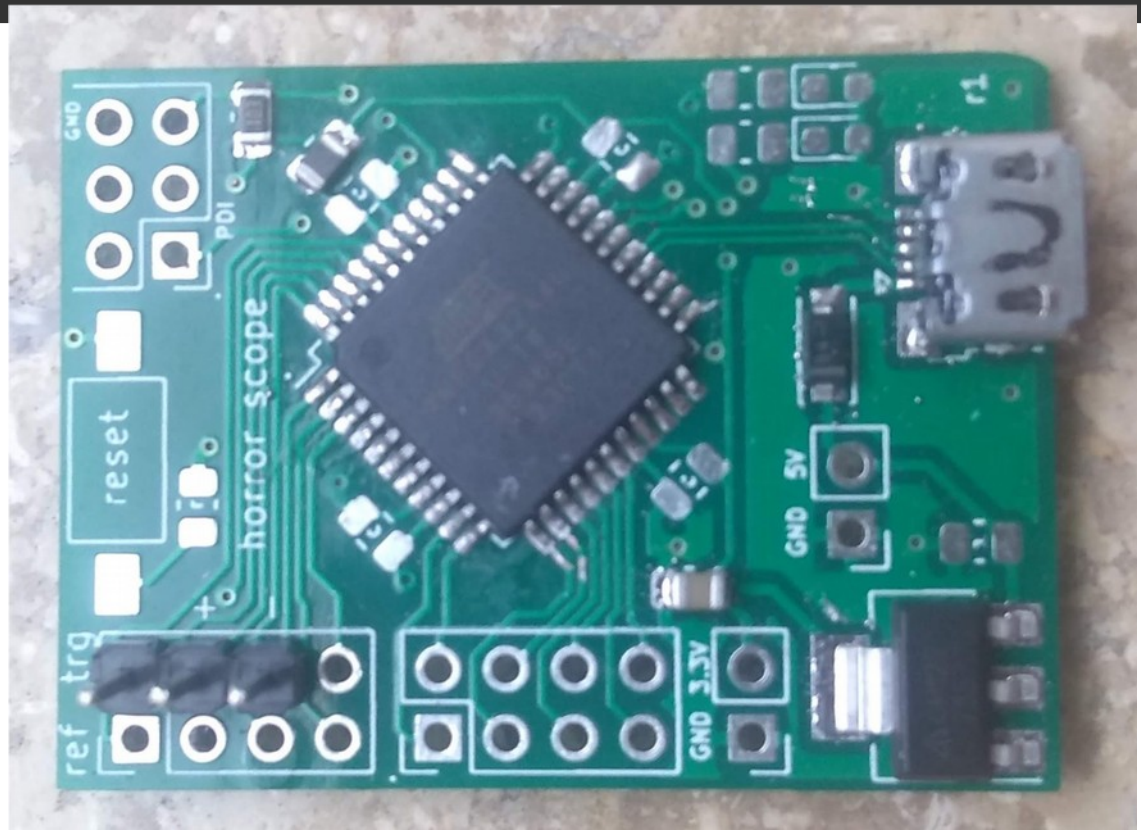
- USB 2.0
- 12-bit ADC @2 MSPS



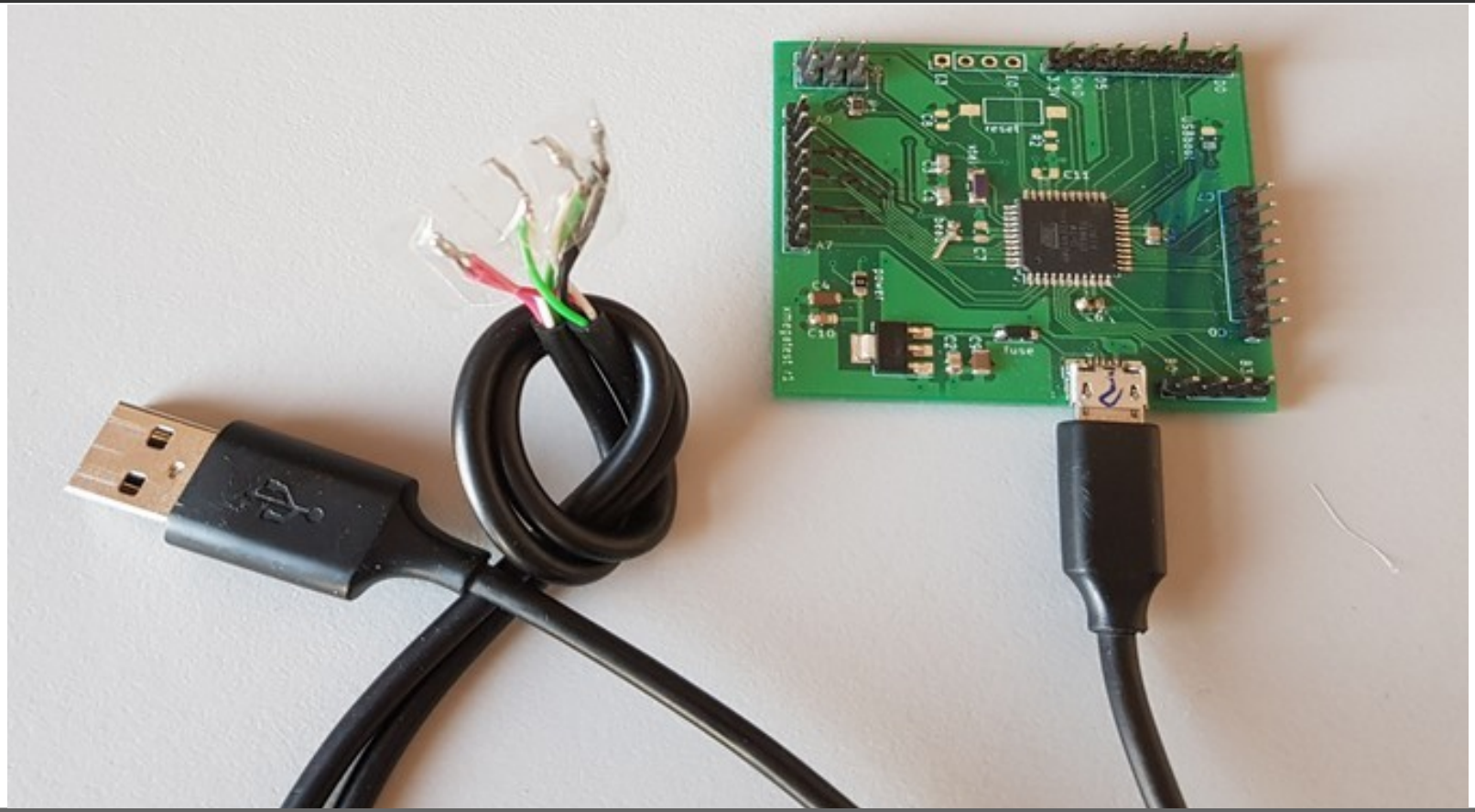
Bill of Materials (BOM)

~5 euro ex. VAT

Xmega:	2.50eur
PCB:	1eur

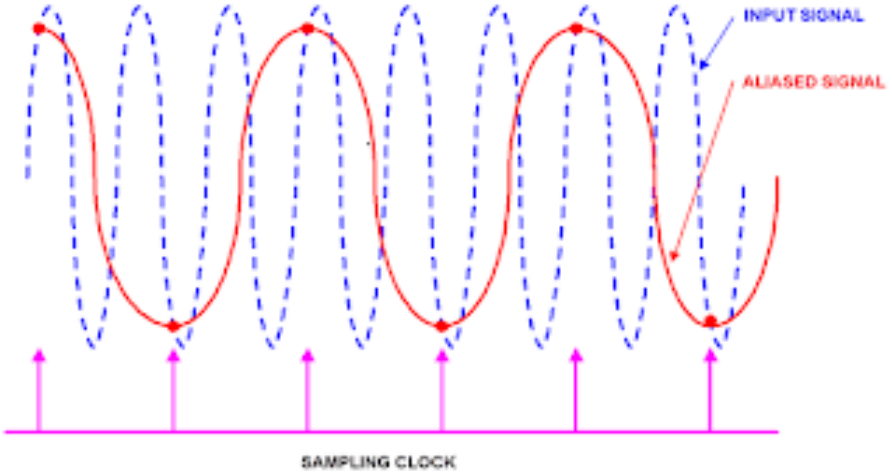


Funny story 1



~~More problems~~
Design considerations

Sampling below Nyqist frequency



Source: http://blog.teledynelecroy.com/2013/06/back-to-basics-sampling-rate.html

Nano:	16Mhz
Our ADC:	2Mhz

Xmega datasheet

28. ADC – 12-bit Analog to Digital Converter

28.1 Features

- One Analog to Digital Converter (ADC)
- 12-bit resolution
- Up to two million samples per second

- **Just a suggestion**

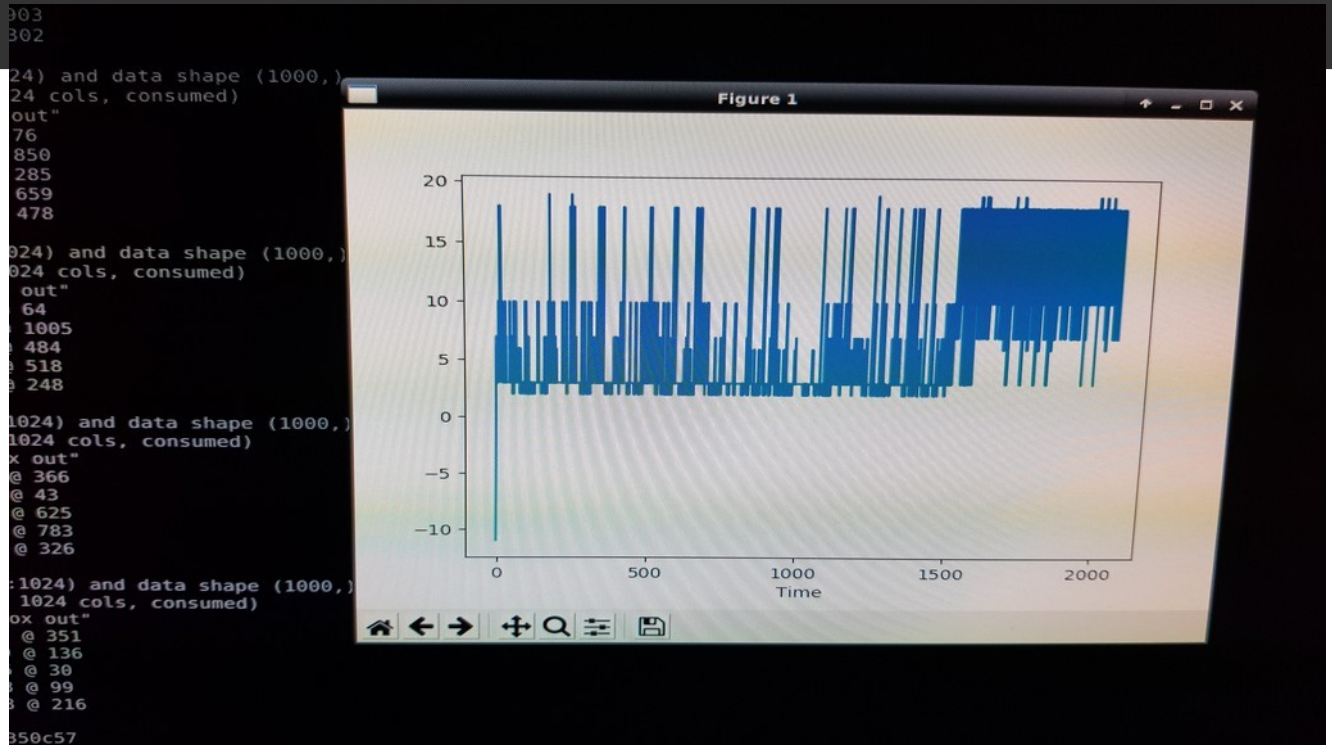
~~More problems~~
Design considerations

No analog front-end

DC offset, resolution, noise, …

- ~~AC coupling,~~ ~~use AREF~~

Funny story 2



Coding is hard

More traces!

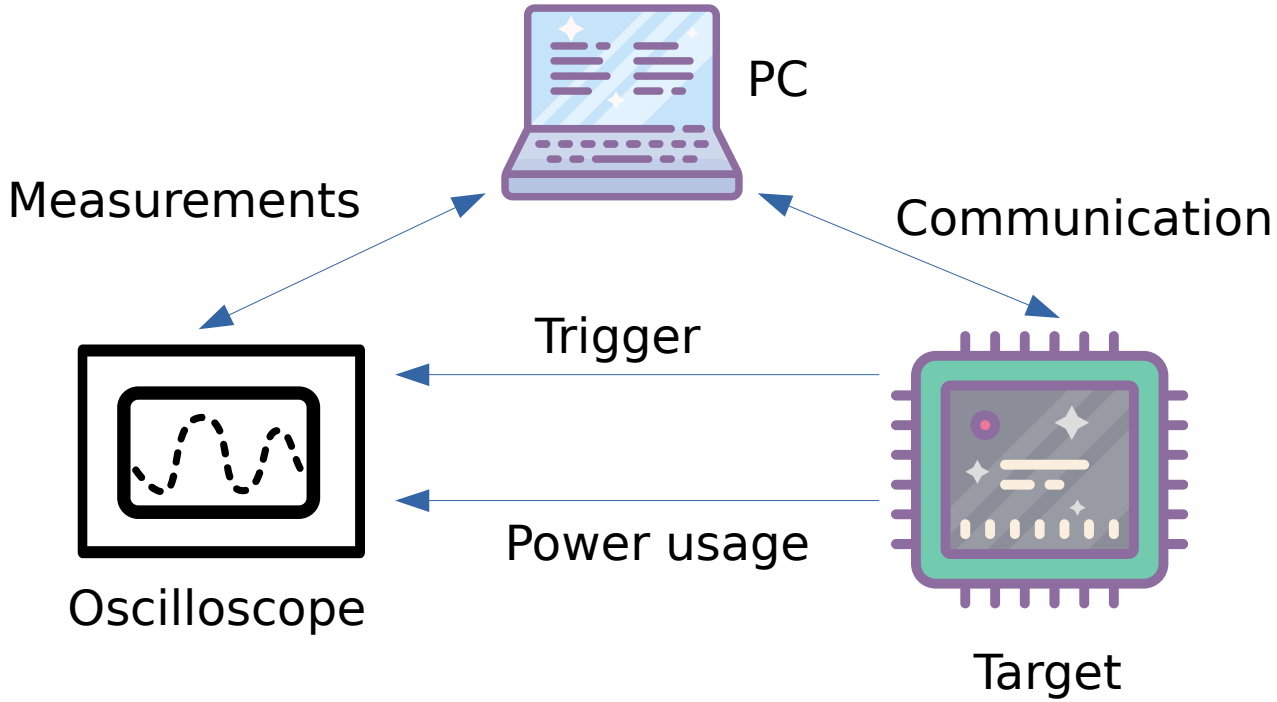
100 traces averaged: A wild AES appears!

Acquisition

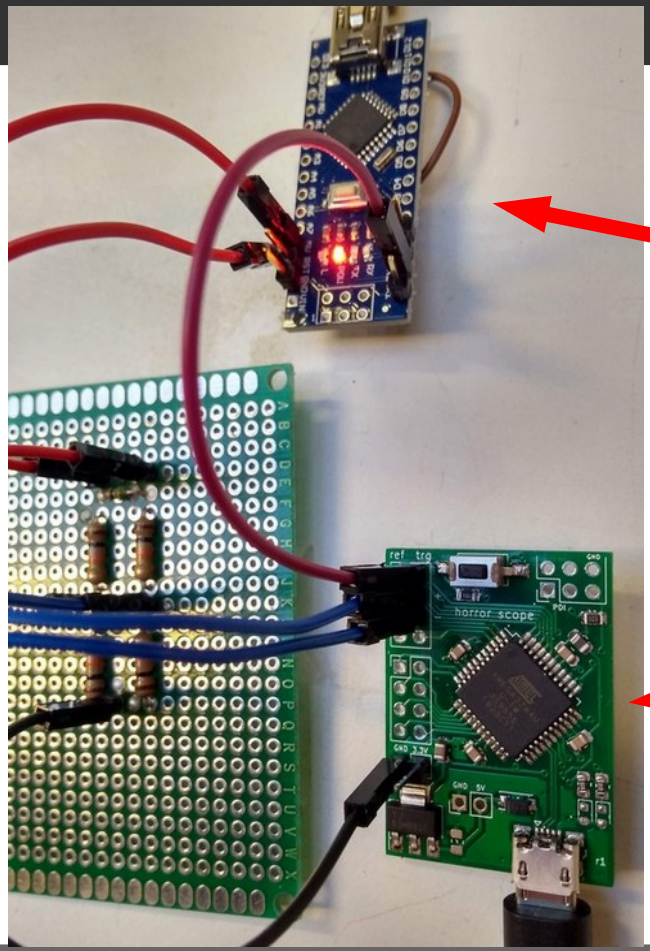
-

So: let’s try it!

Setup


PC
Measurements
Communication
Trigger
Power usage
Oscilloscope
Target

Real world setup



We have:

“Target”:
Arduino Nano + AES

“Oscilloscope”:
HorrorScope

Triggering

How does the scope know when to measure?

- Not enough SRAM to sample all the time
- We need to sync the scope to the target

So, whats the last thing we control?

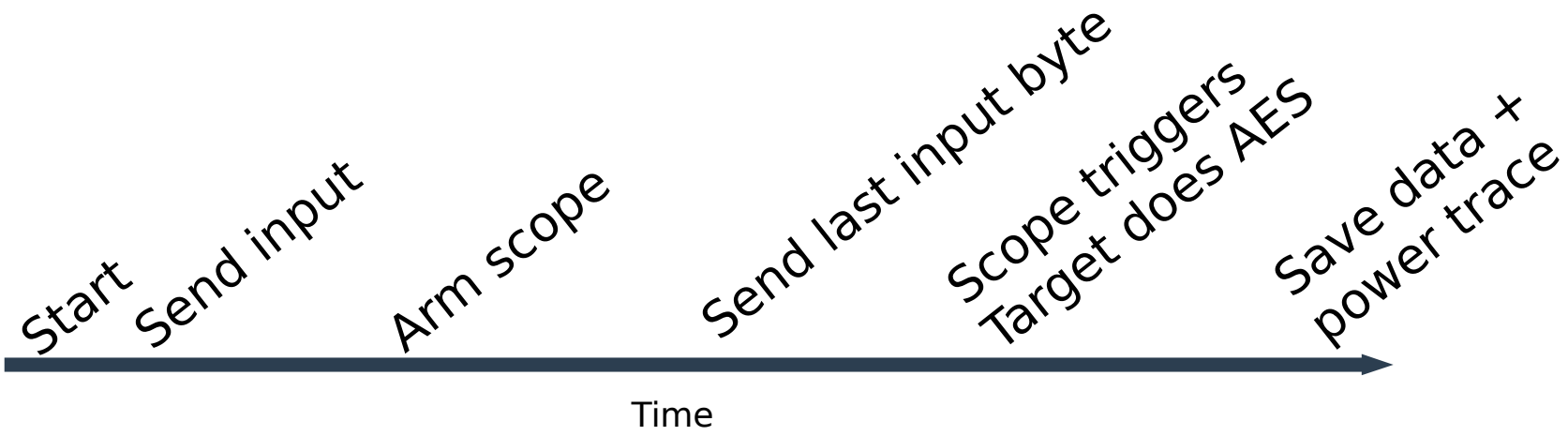
- Sending the input

Triggering plan

- **HorrorScope measuring procedure**
  - Send command to Scope (arm)
  - GPIO pin turns high
  - Actually start measuring

  - Connect Scope GPIO pin to RX on Nano

When to start measuring …



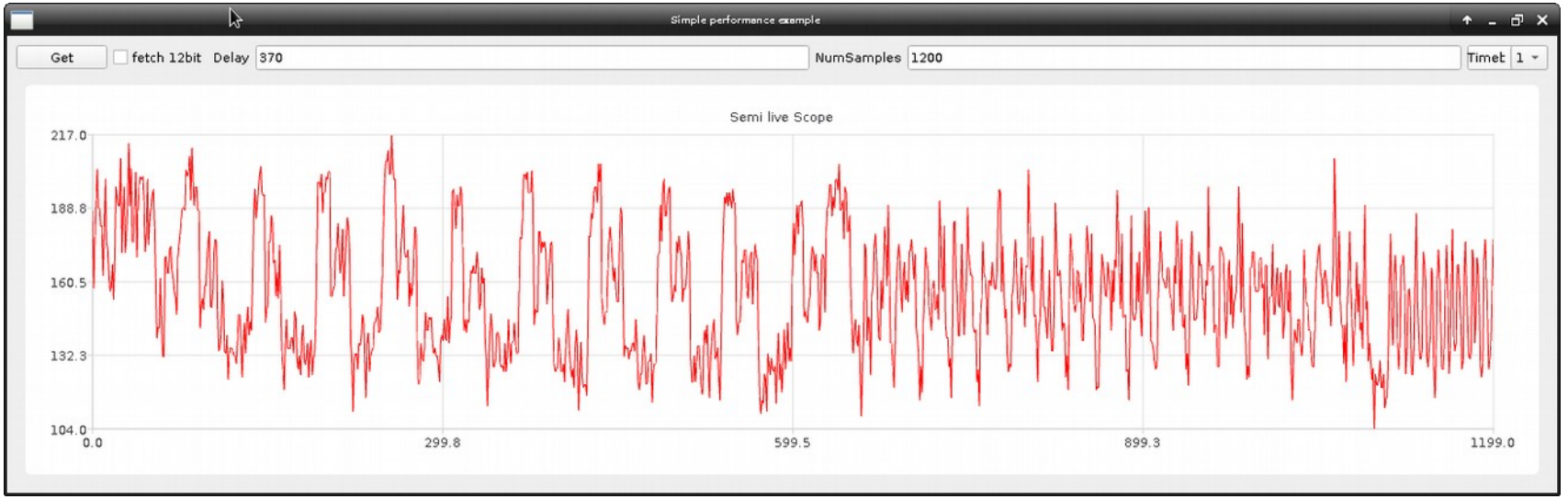
Collect traces...

We need a lot of traces

- Make sure the first/or last round is in view
- Select a high Sample speed
  - Ensure there’s a margin before/after the round

Take Traces

- **AES: initial round, 9 rounds, a final round**

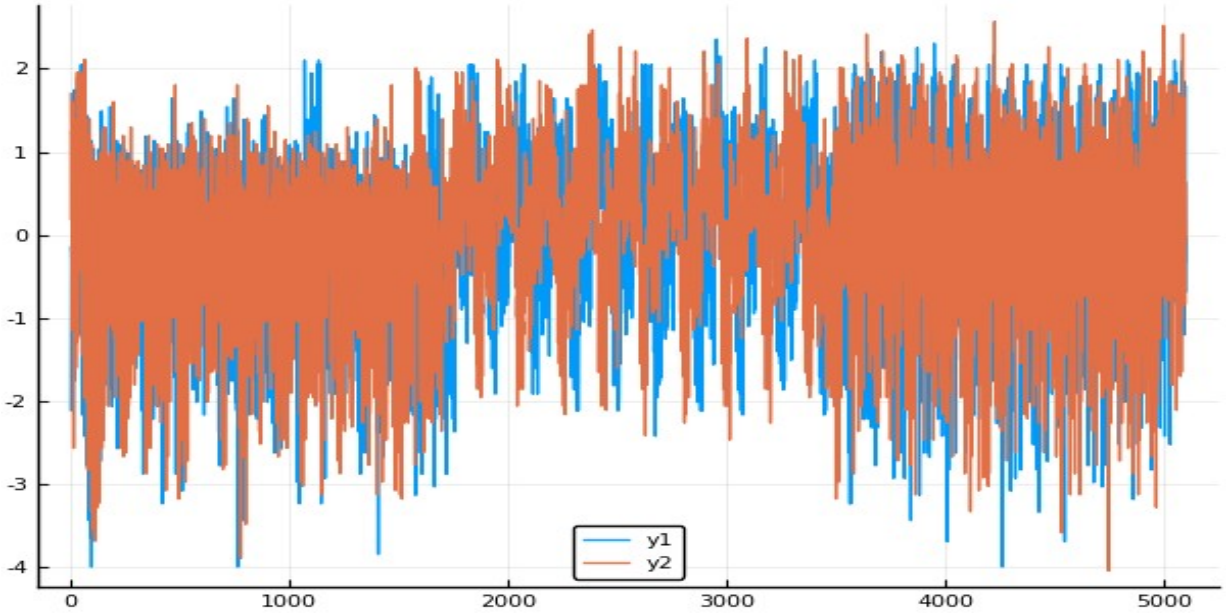


Acquiring Traces …

…. Here’s one we made earlier

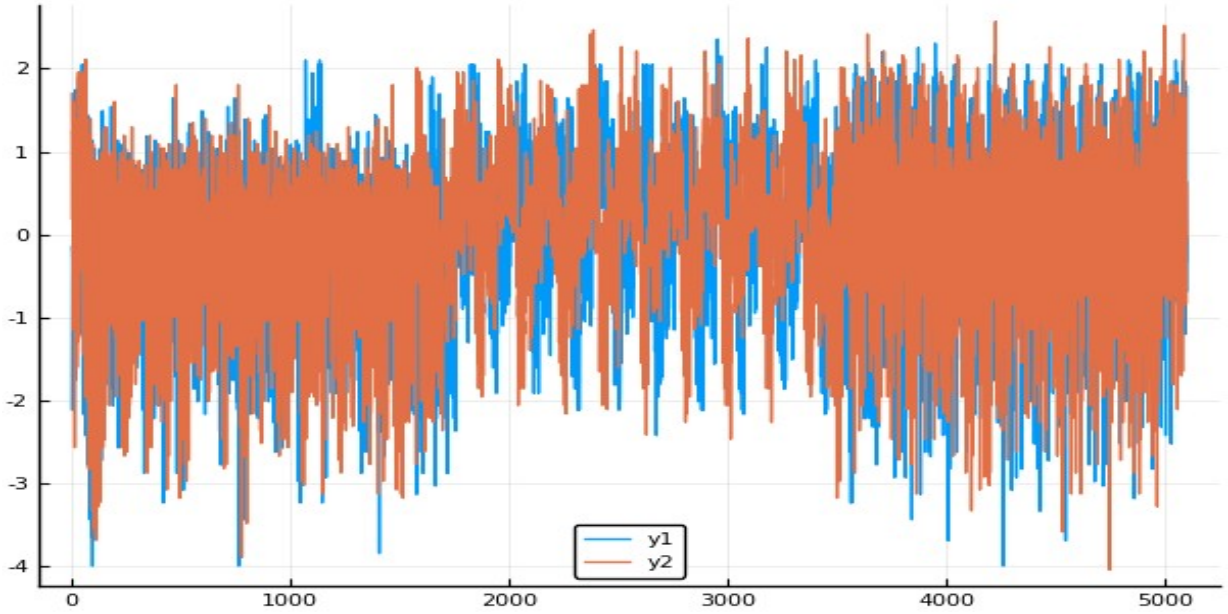
Traces are bad

- **Why are they bad?**

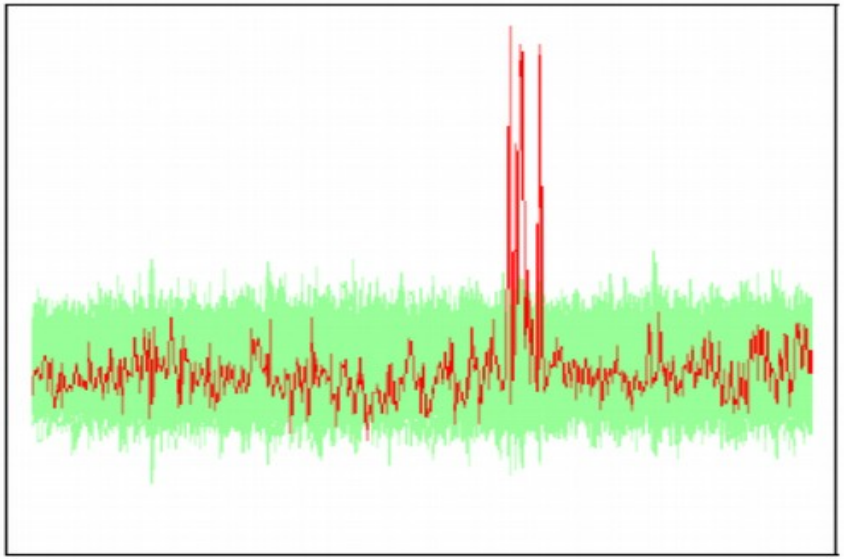


What’s wrong?

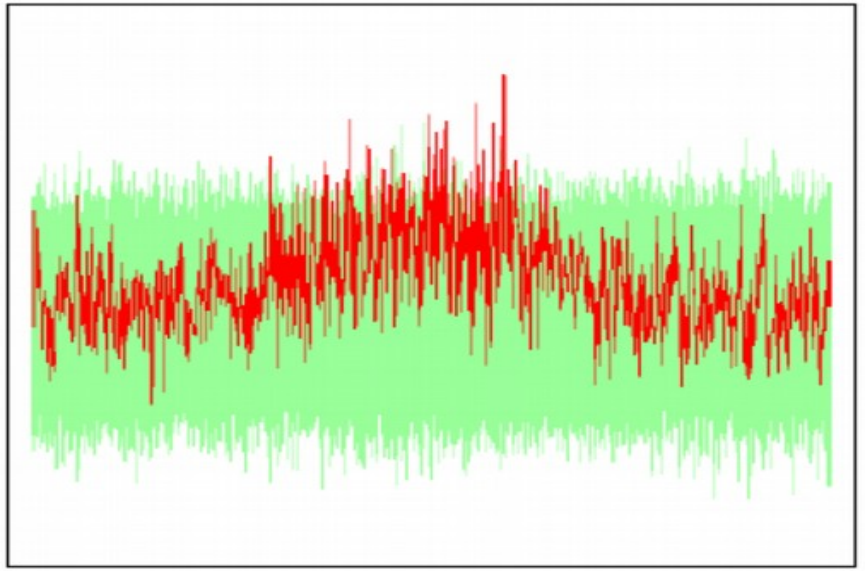
- **Misalignment** ⟺



Signal spread



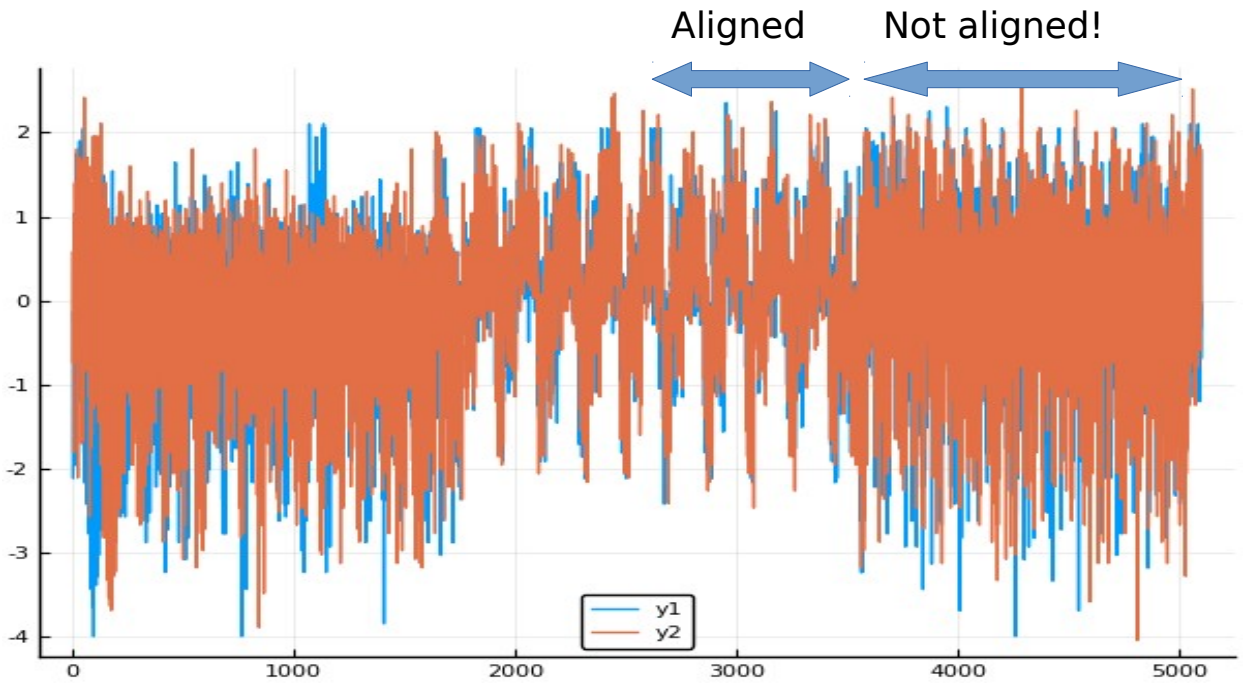
well aligned traces



misaligned traces

Source: Side channel analysis, practice and a bit of theory. Ilya Kizhvatov

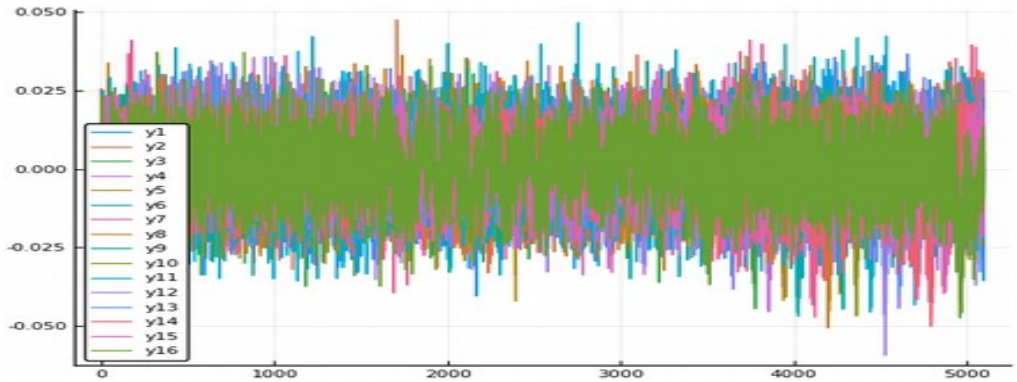
Aligned



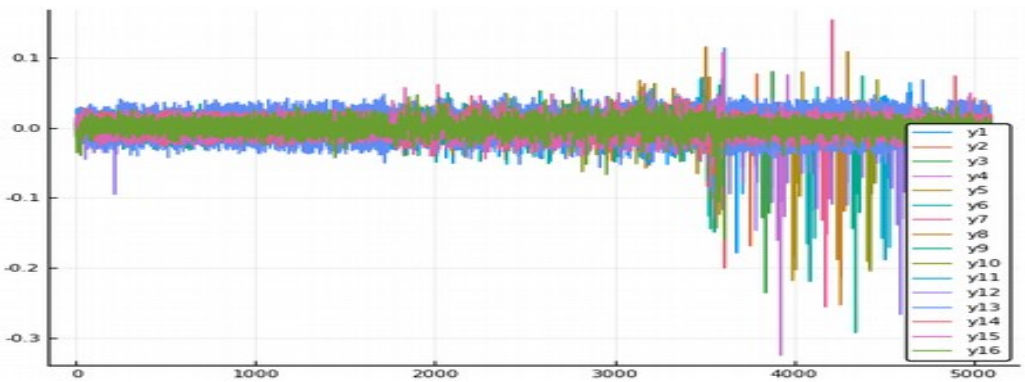
Thank you JLSCA!

Why is alignment important?

Before



after



Let’s get keys

- **DEMO**
  - Jupyter notebook

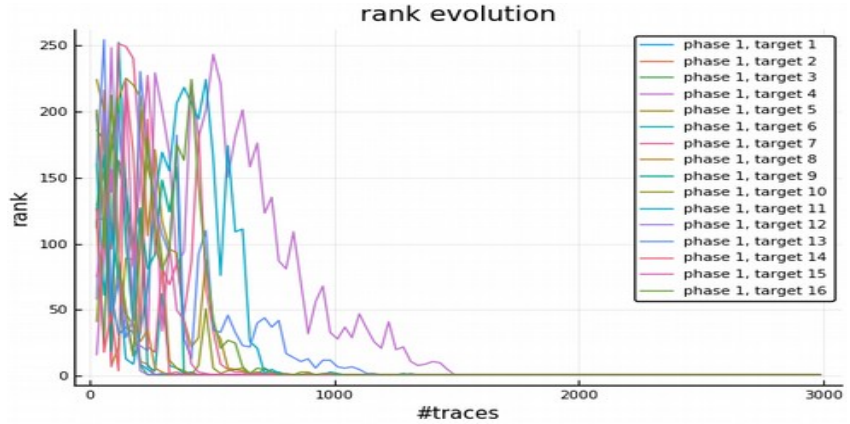
Comparing the scopes

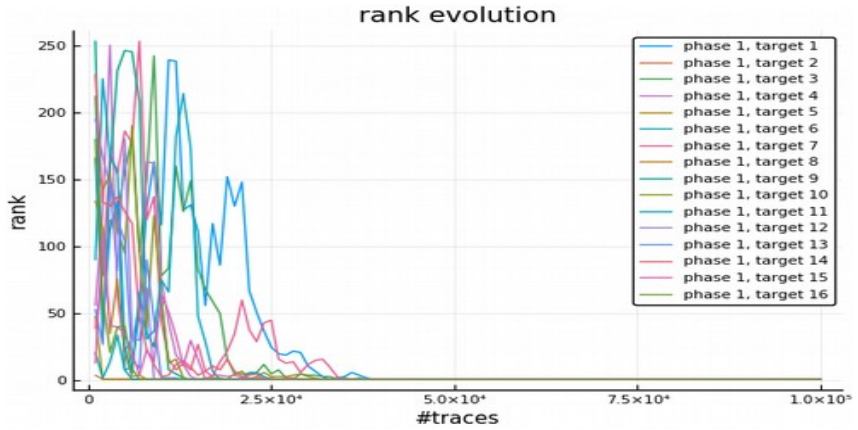
Vds1022 ($70):

~1.5k traces

HorrorScope:

~30k traces





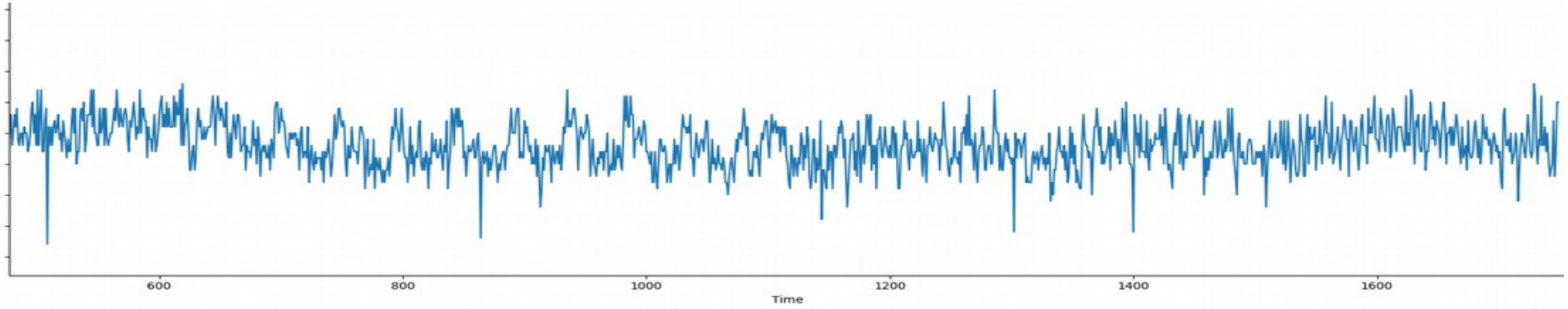
Funny Story 3

- **How many mistakes can we make before it doesnt work?**

- **Silkscreen-Off-By-One**

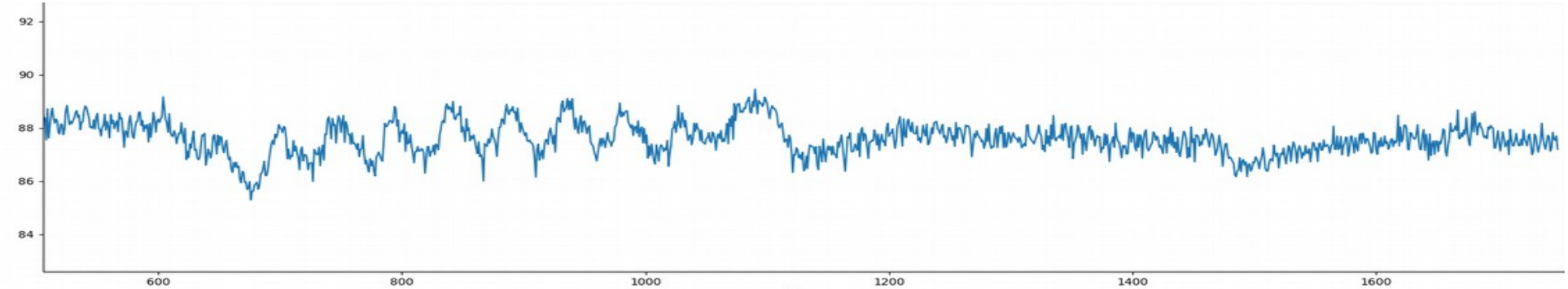
Funny story 3

- People told us the Xmega ADC was bad



Funny story 3

- **1000 averaged, no alignment**



Strength in numbers

- Side Channel Attacks require overcoming the noise
- Noise can be reduced through taking more traces, but not in every case

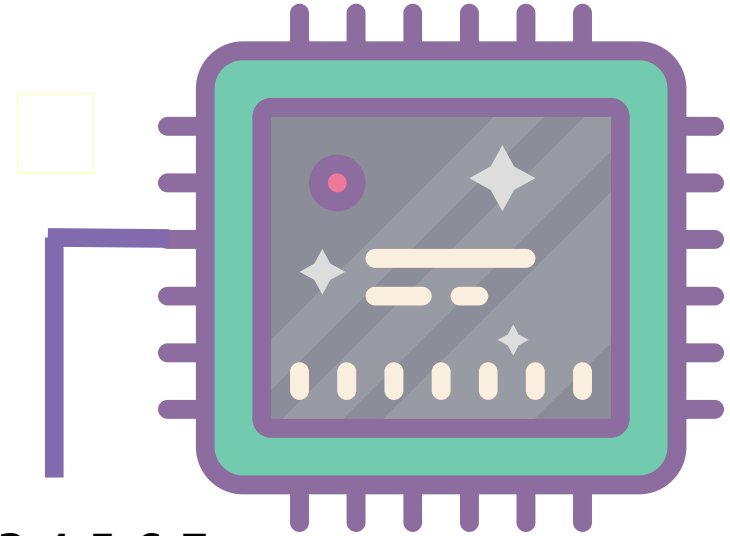
Fault Injection

HorrorScope can sort of power the Nano

- The Nano wants 5v
- Xmega GPIO pins provide 3.2v

Powering the Nano

V_{in}
(+5V)



PORT C
PIN 0,1,2,3,4,5,6,7

Ok faults, now what?

- **Perform Fault Injection and Differential Fault Analysis**
- **Inject faults into AES and recover the key**

What can you do about it?

- **Threat model**
  - Do you *need* to resist physical attacks?

    Best defense: make sure it doesn’t matter!

- **Basic steps**
  - Use hardware with built-in countermeasures

    .. and check it with a (Horror)scope :)

    (or a ChipWhisperer)!

Conclusion

- **Hardware attacks are cheaper than we thought**
- **Side Channel Analysis is something *you* can do at home – and you should try it**

Special Thanks

- **Cees-Bart ‘ceesb’ Breunesse**
  - https://github.com/Riscure/Jlsca
- **Rafa Boix Carpi**
  - For saying it can’t be done
- **Ilya Kizhvatov**
  - Letting us steal his pictures
- **Workshop attendees**
  - For their feedback and love

Homework

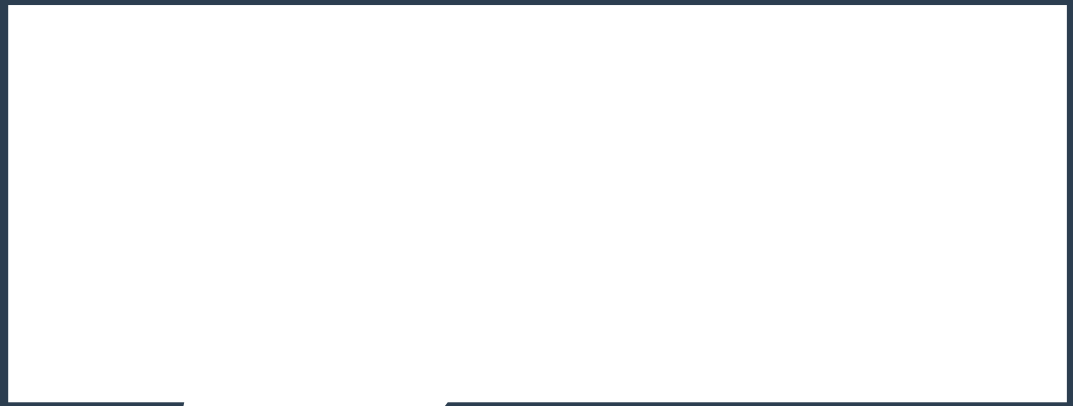
https://github.com/albert-spruyt/HorrorScope/

Provided: power traces and Jlsca notebook.

You should now be able to get the key!

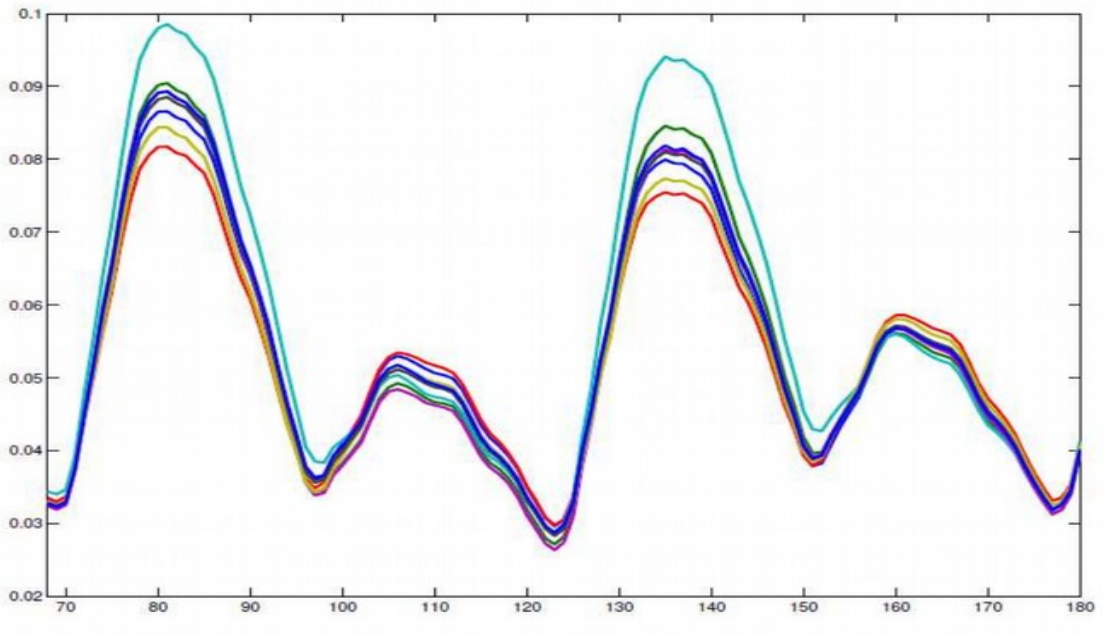
(Also in the repo: schematics/source/etc)

.. we’re hoping for a port to the STM32 – SCA for $2?



Hamming weight

- Power profiles based on different data
- Averaged
- We can see the data being processed!



Source: Side channel analysis, practice and a bit of theory. Ilya Kizhvatov

Power cut: easy?

Power cut: UFO