PUF on a Simple Microcontroller

Josef Hlaváč, Mikhail Platonov, Róbert Lórencz

Faculty of Information Technology
Czech Technical University in Prague

Introduction & motivation

- Physical Unclonable Functions (PUF)
  - PUF output specific to a particular piece of HW
  - relies on various (parasitic) effects that are (hopefully) impossible to reproduce
  - “Hot” research topic in recent years
  - Various uses in security
    - Anti-counterfeiting
    - Authentication
    - Identification
    - …

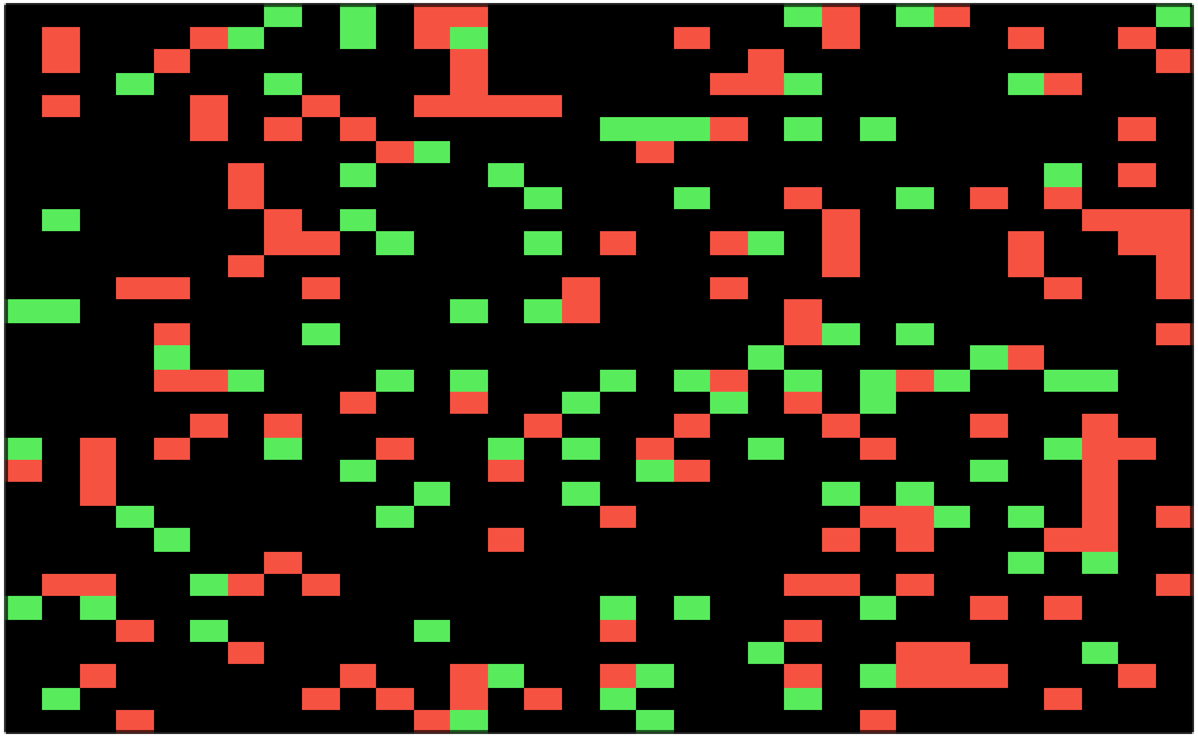
Problem statement

- Problem statement:
  - Most of current PUF research focuses on FPGA
  - Can we implement PUF on a simple AVR-series microcontroller?
- Approach
  - The most obvious option: Initial SRAM contents

Initial tests

- Is the idea plausible?
  - ATtiny2313
    - 8-bit AVR series microcontroller
    - 128 bytes of SRAM
- Results
  - 147 (14%) zero bits
  - 81 (8%) random bits
  - 796 (78%) one bits

Memory map



🟥 = 0 🟩 = random ⬛ = 1

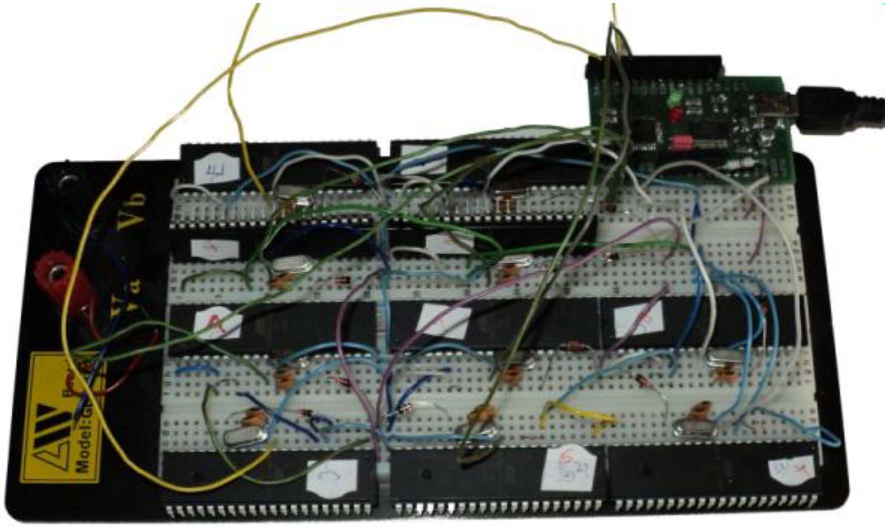
More tests

- How long does it take to “reset” the SRAM?
  - Influence of power-off time?
  - Influence of previous content?

- Does the pattern really differ in different chips?
  - Let’s try with more RAM, too

More tests

- Setup for more tests
  - ATmega1284
    - 8-bit AVR mega series microcontroller
    - 16384 bytes of SRAM
    - 10 devices



Influence of previous content

- Initial test
  - ATtiny2313
    - SRAM filled with 0, 10 minutes off: 181 zeros, 755 ones
    - SRAM filled with 1, 10 minutes off: 186 zeros, 757 ones

    → not much difference

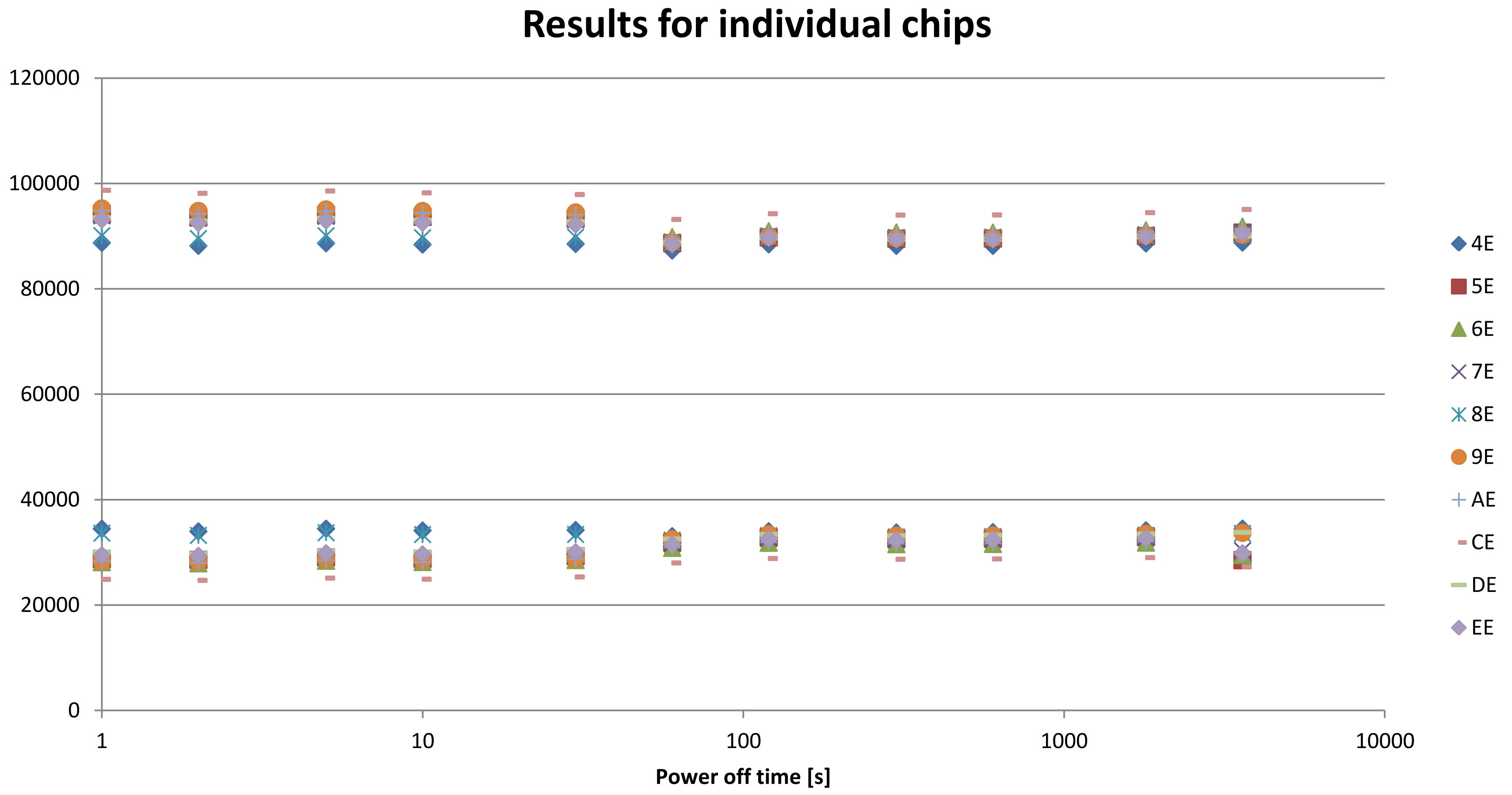
Influence of previous content

- What others say
    - Retained data in SRAM lost after few ms [3,4]

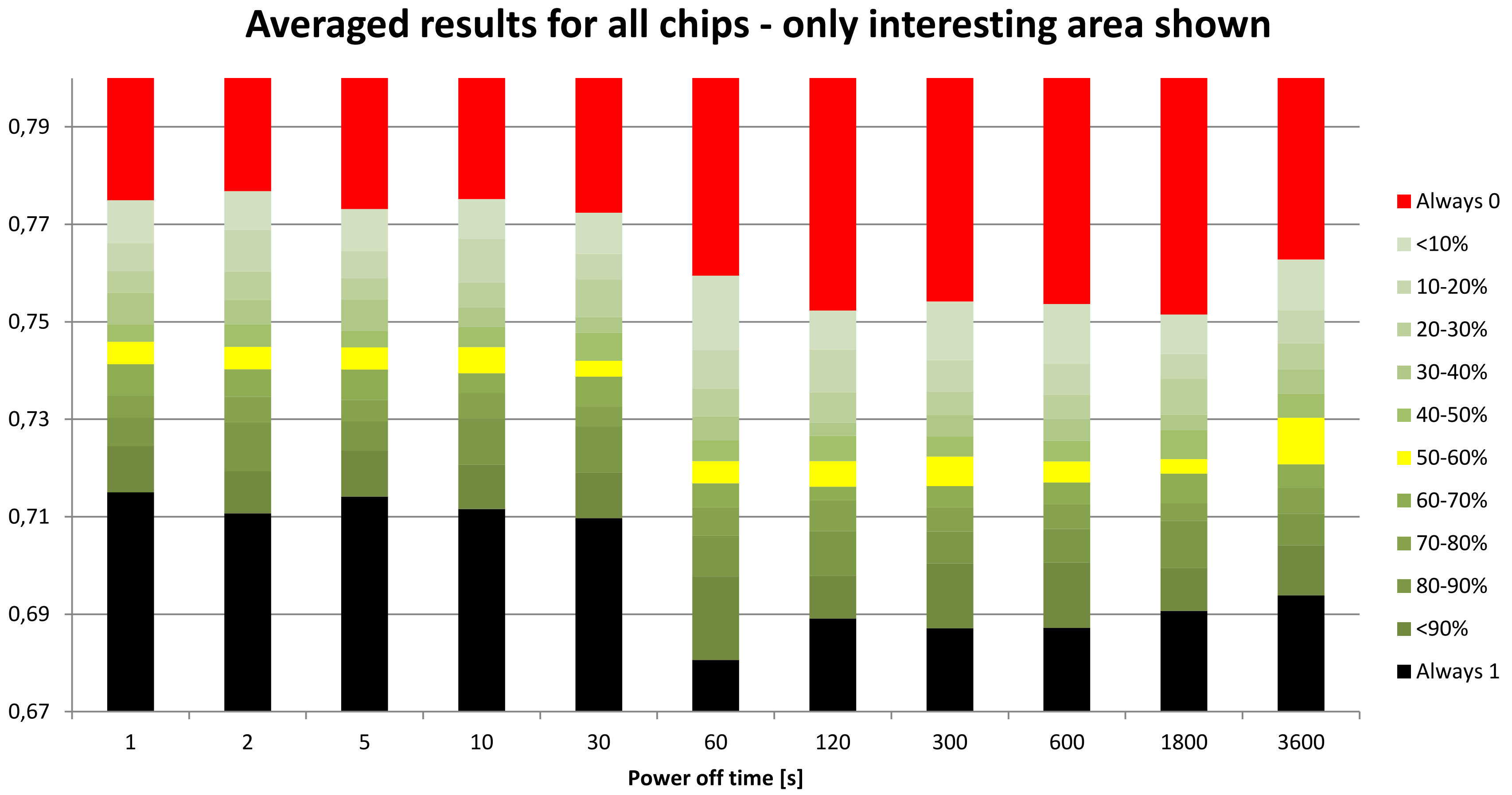
      vs.

    - Measurable effect even after tens of mins [2]
- More tests
    - ATmega1284
        - SRAM filled with 0 (since 1 is much more common)
        - Various power-off times, from 1 s to 1 hour
        - 10 chips, all with the same datecode

Stable bits vs. power-off time



Bits vs. power-off time



Influence of previous content

- Results
  - No influence detected
    (but the results are not quite conclusive)
  - “Step” in between 30 and 60 secs is likely caused by the experiment setup
  - SRAM seems to be biased towards “1” in AVR devices

Differences among chips

- Critical question: Is the memory pattern really unique for each chip?

Differences among chips

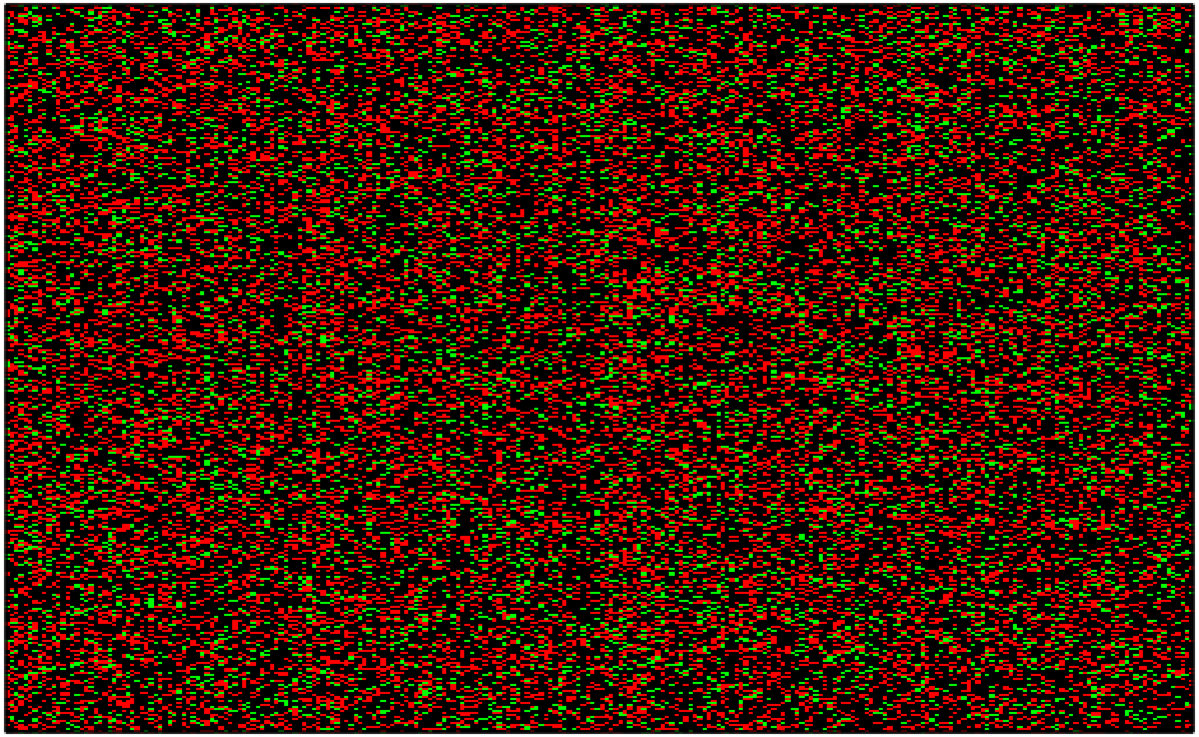
- In one chip:
  - Stable zeros: 13% (min) / 20% (avg) / 25% (max)
  - Stable ones: 35% (min) / 61% (avg) / 67% (max)

    (measurement error suspected in one chip)
- Two different chips:
  - Stable zeros: 1.8% (min) / 4.0% (avg) / 6.1% (max)
  - Stable ones: 20% (min) / 37% (avg) / 45% (max)

    (“stable” = stable bits at same locations in both chips)

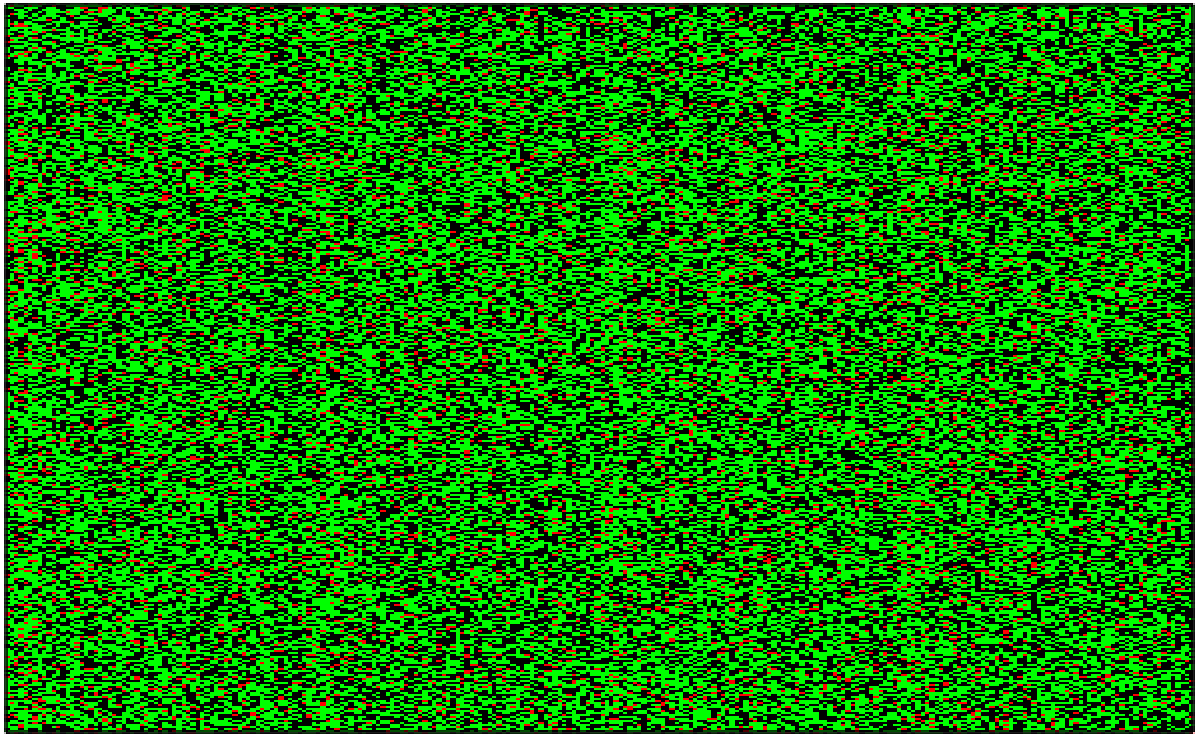
    (measured across all possible pairs)

Memory map - one chip



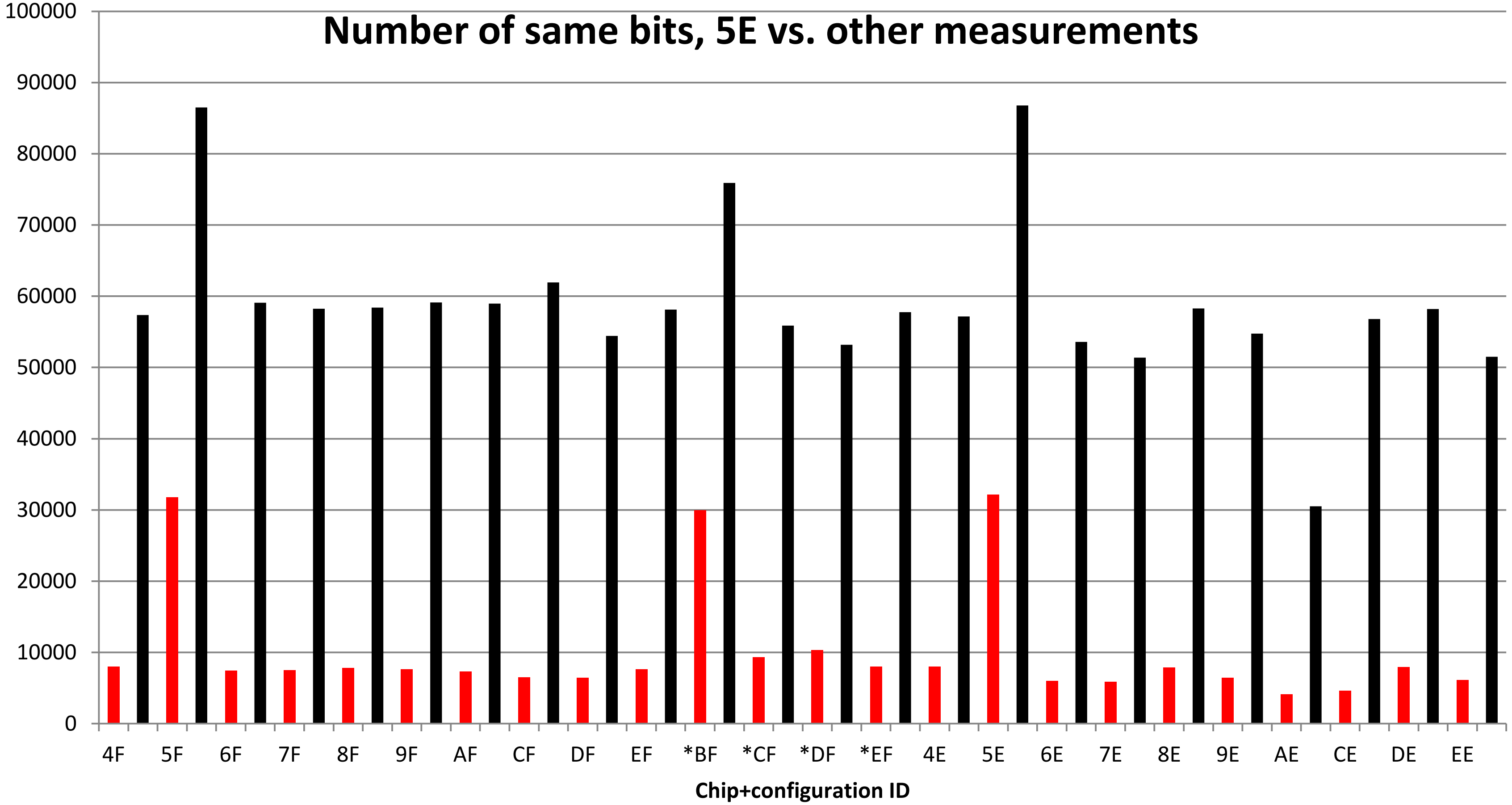
🟥 = 0 🟩 = random ⬛ = 1

Memory map - two chips

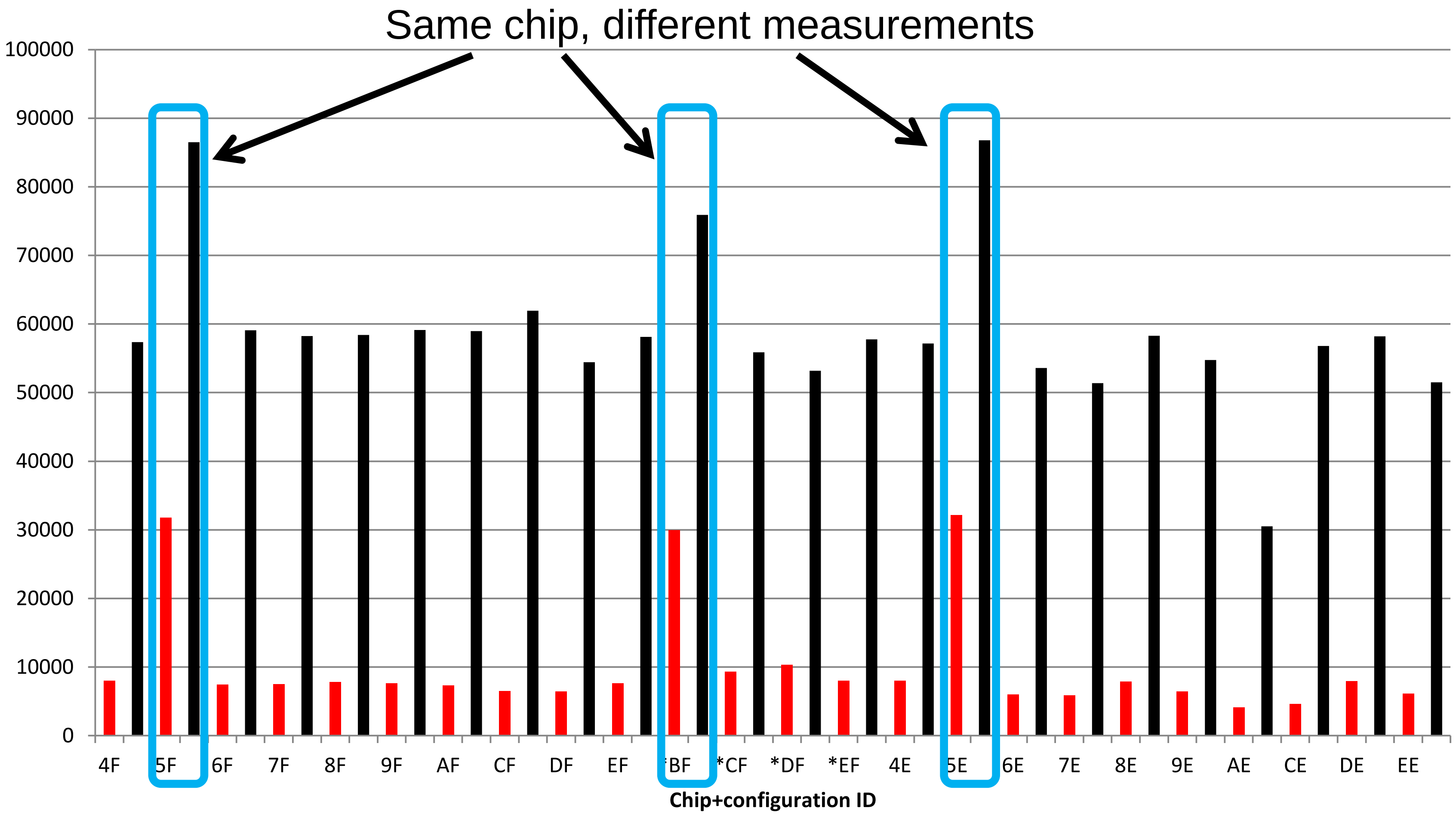


🟥 = stable 0   🟩 = different/random   ⬛ = stable 1

Stable 0 / stable 1 in various cases



Stable 0 / stable 1 in various cases



Same chip, different measurements

TODO a.k.a. Future work

- More experiments needed
  - Long-term aging, “burn-in” effects
  - Supply voltage
  - Temperature
  - Structural considerations
- Suggest a good way of using the PUF
  - Inspiration can be drawn from [1], ...

Conclusion

- SRAM in the AVR series microcontrollers can be used to construct a PUF
- Preliminary results presented
- More work needed ☺

References

[1] Holcomb, D.E. et al.: Power-up SRAM state as an Identifying Fingerprint and Source of True Random Numbers. IEEE Trans. on Computers, vol. 57, no. 11, 2008.

[2] Colopy R., Chopra J.: SRAM Characteristics as Physical Unclonable Functions. A Major Qualifying Project Report, no. MQP-BS2-0803, Worcester Polytechnic Institute, 2009.

[3] Guajardo J. et al.: FPGA Intrinsic PUFs and Their Use for IP Protection. In: Proceedings of CHES 2007, LNCS 4727, pp. 63–80.

[4] Skorobogatov S.: Low temperature data remanence in static RAM. Technical report No. 536. Computer Laboratory, University of Cambridge, 2002.