A Novel Microprocessor-intrinsic Physical Unclonable Function

Abhranil Maiti
Patrick Schaumont

ECE Department, Virginia Tech

FPL 2012, Oslo, Norway

08/30/2012

Outline

- **What is a microprocessor-intrinsic PUF? Why do we need it ?**
- **Detail of the microprocessor-intrinsic PUF**
- **Results**
- **Conclusion and future work**

Motivation

SW IP is valuable and needs to be protected.

⇩

One way of preventing illegal use of SW IP is to restrict its execution to an authenticated processor. Additionally, in FPGAs, we need to authenticate the HW too.

⇩

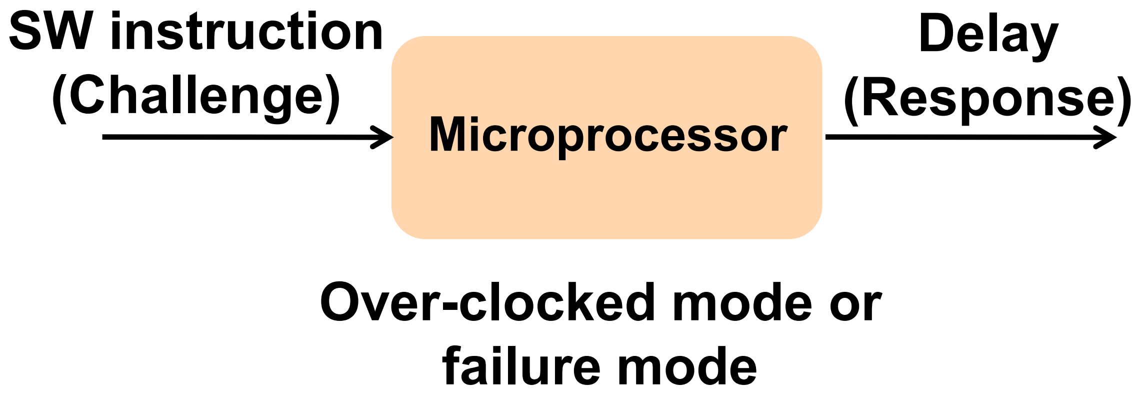
How can we do it cheaply?

⇩

Idea: extracting the characteristics of a microprocessor using software programs.

Microprocessor-intrinsic PUF

- **Extracts variability in a microprocessor pipeline to identify a chip.**
- **Accepts a software instruction as a challenge and produces the delay in a data path or a control path as the response.**
- **The delay is measured by *over-clocking* the microprocessor.**



Why do we need a new PUF?

- **Majority of the proposed PUFs require significant hardware resources e.g. RO-PUF requires a pair of ROs to generate a single response bit.**

- **There are PUFs that requires no additional hardware *(intrinsic)* : memory-based PUFs.**
  - **SRAM PUFs** (J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, “Fpga intrinsic pufs and their use for ip protection,” in *Proceedings of the 9th international workshop on Cryptographic Hardware and Embedded Systems*, ser. CHES ’07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 63–80.)
  - **Flip-flop-based PUF** (R. Maes, P. Tuyls, and I. Verbauwhede, “Intrinsic pufs from flip-flops on reconfigurable devices,” in *3rd Benelux Workshop on Information and System Security (WISSec 2008)*,Eindhoven,NL, 2008, p. 17).

  They require power-cycling to generate the CRPs.

- **The proposed PUF is intrinsic but requires no power-cycling.**

Benefits of the proposed PUF

- **A microprocessor is a ubiquitous circuit element, present in almost every embedded application.**

- **It provides an easy way of integrating low-level hardware information with the high-level software applications.**

- **Any post-processing of the PUF data can be flexibly done in software obviating any need of costly error-correction hardware.**

Related work

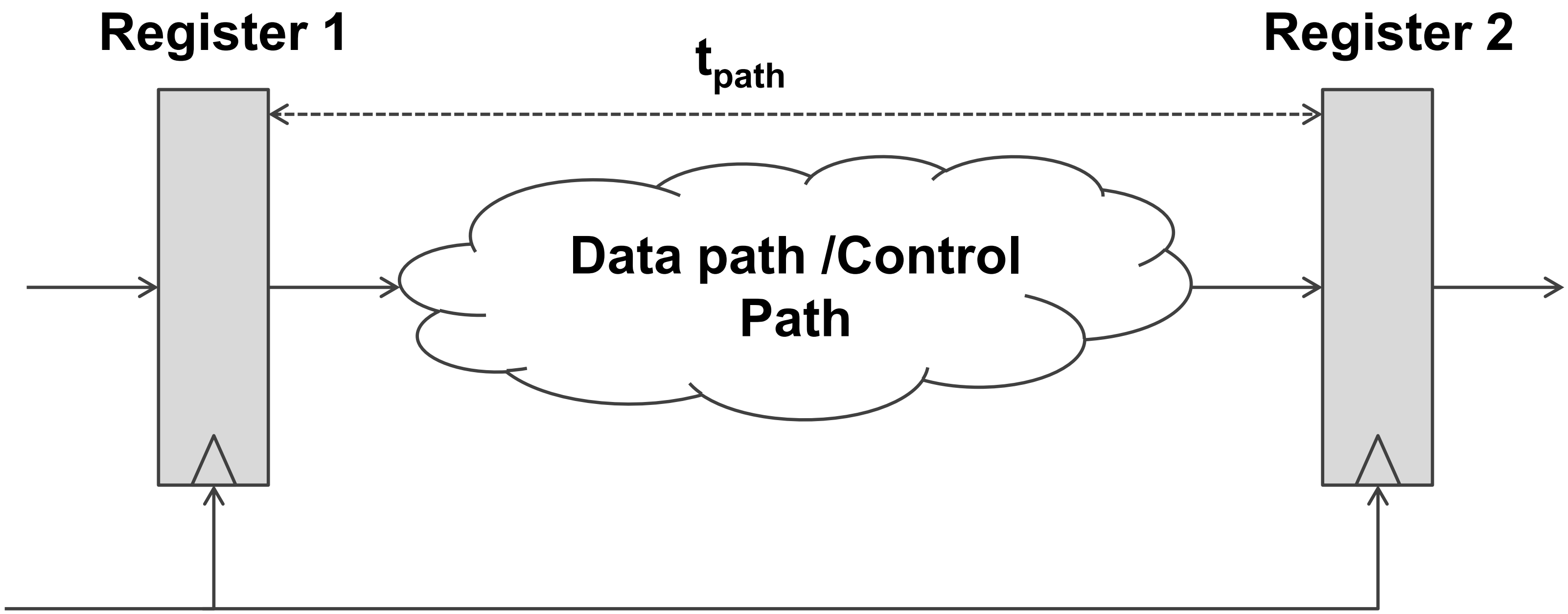
- G. Suh, C. O'Donnell, I. Sachdev, and S. Devadas, “Design and implementation of the aegis single-chip secure processor using physical random functions,” in *Computer Architecture, 2005. ISCA '05. Proceedings. 32nd International Symposium on*, june 2005, pp. 25 – 36.

- D. Y. Deng, A. H. Chan, and G. E. Suh, “Hardware authentication leveraging performance limits in detailed simulations and emulations,” in *Proceedings of the 46th Annual Design Automation Conference*, ser. DAC '09, 2009, pp. 682–687.

Outline

- **What is a microprocessor-intrinsic PUF? Why do we need it ?**
- **Detail of the microprocessor-intrinsic PUF**
- **Results**
- **Conclusion and future work**

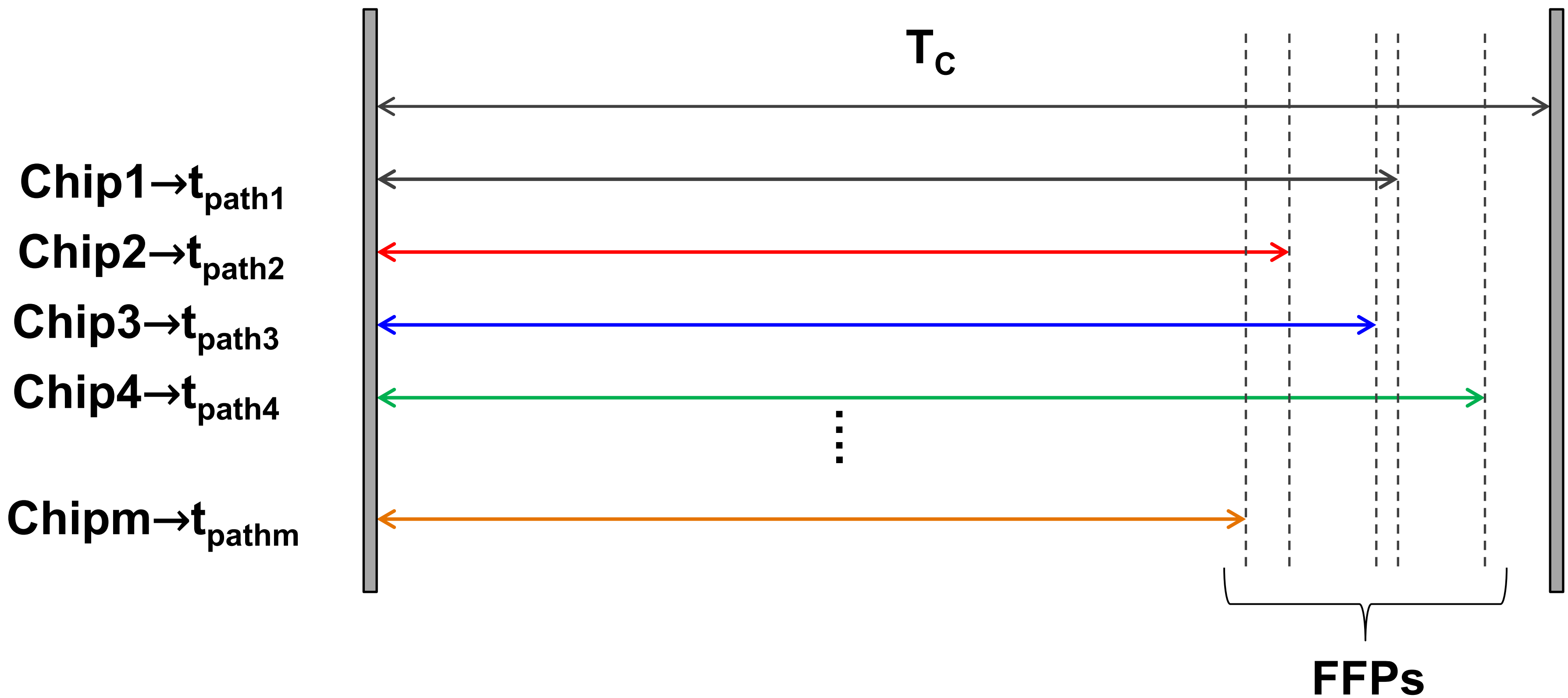
Basic Idea : a pipelined delay path



Clock (frequency = f, period = T_c, $T_c = 1/f$)

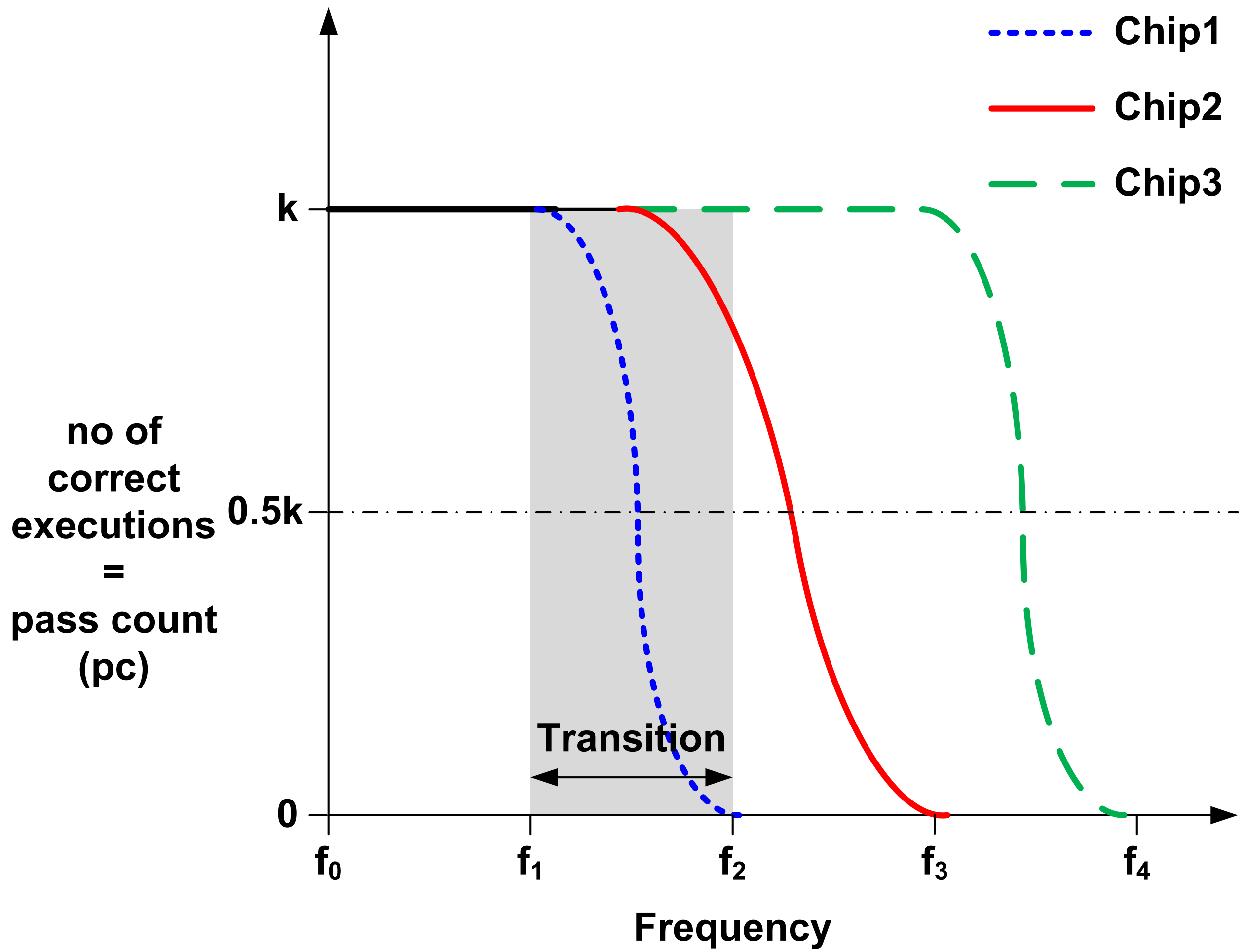
$$T_c \geq t_{c\_q} + t_{path} + t_{setup}$$

Frequency Failure Points



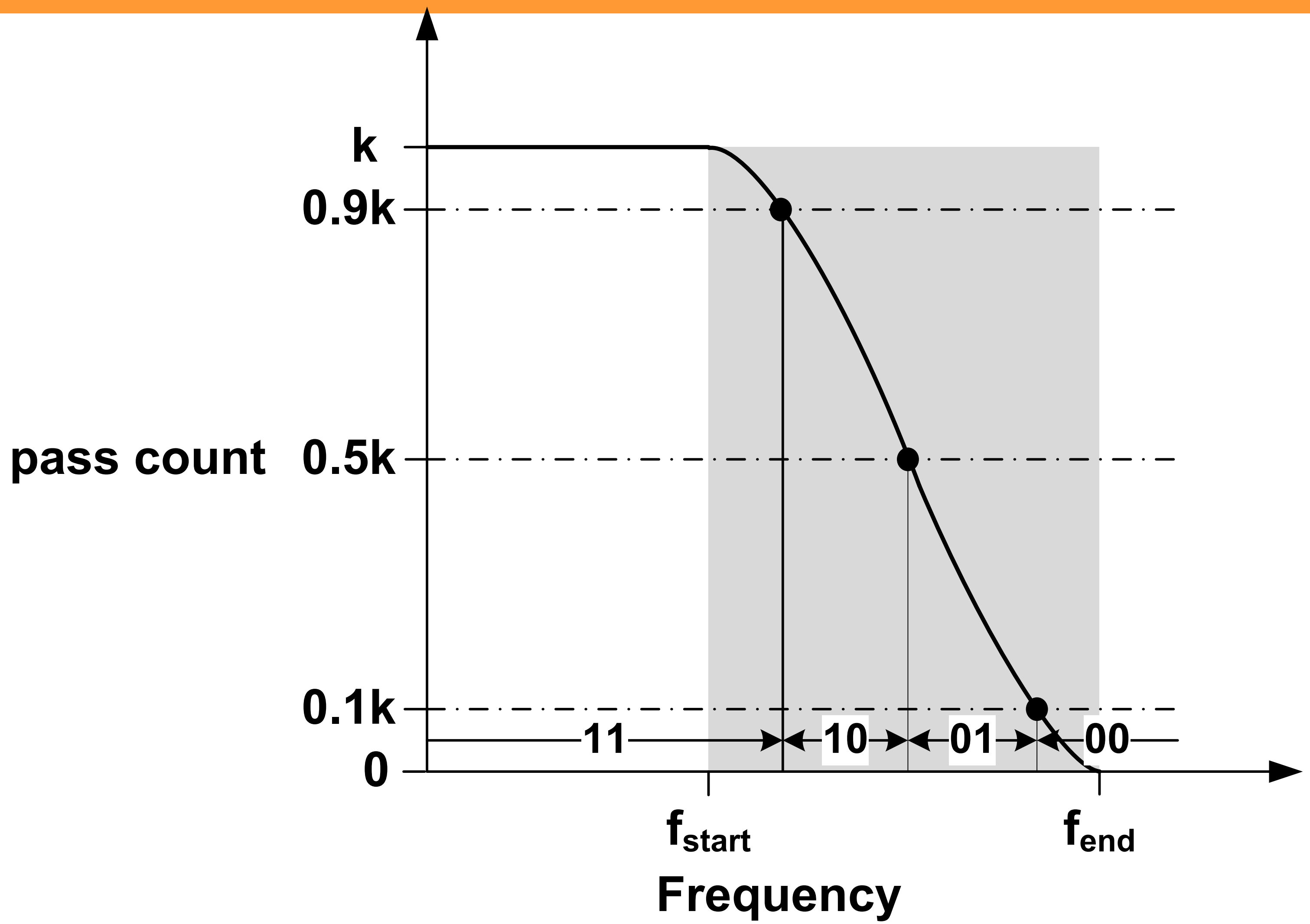
FFP = Frequency Failure Points

Failure transition



Failure transition information (FTI)
= { f_{start}, f_{end}, pc values in the transition region}

CRP formation



pc : 95, 87, 46, 17 and 5 ⟶ r : 11, 10, 01, 01 and 00 ⟶ 1110010100

Authentication mechanism

2. Searches the db and defines a challenge by selecting one or more instructions and a set of frequencies

1. Claims identity

3. Asks the chip to reproduce the responses

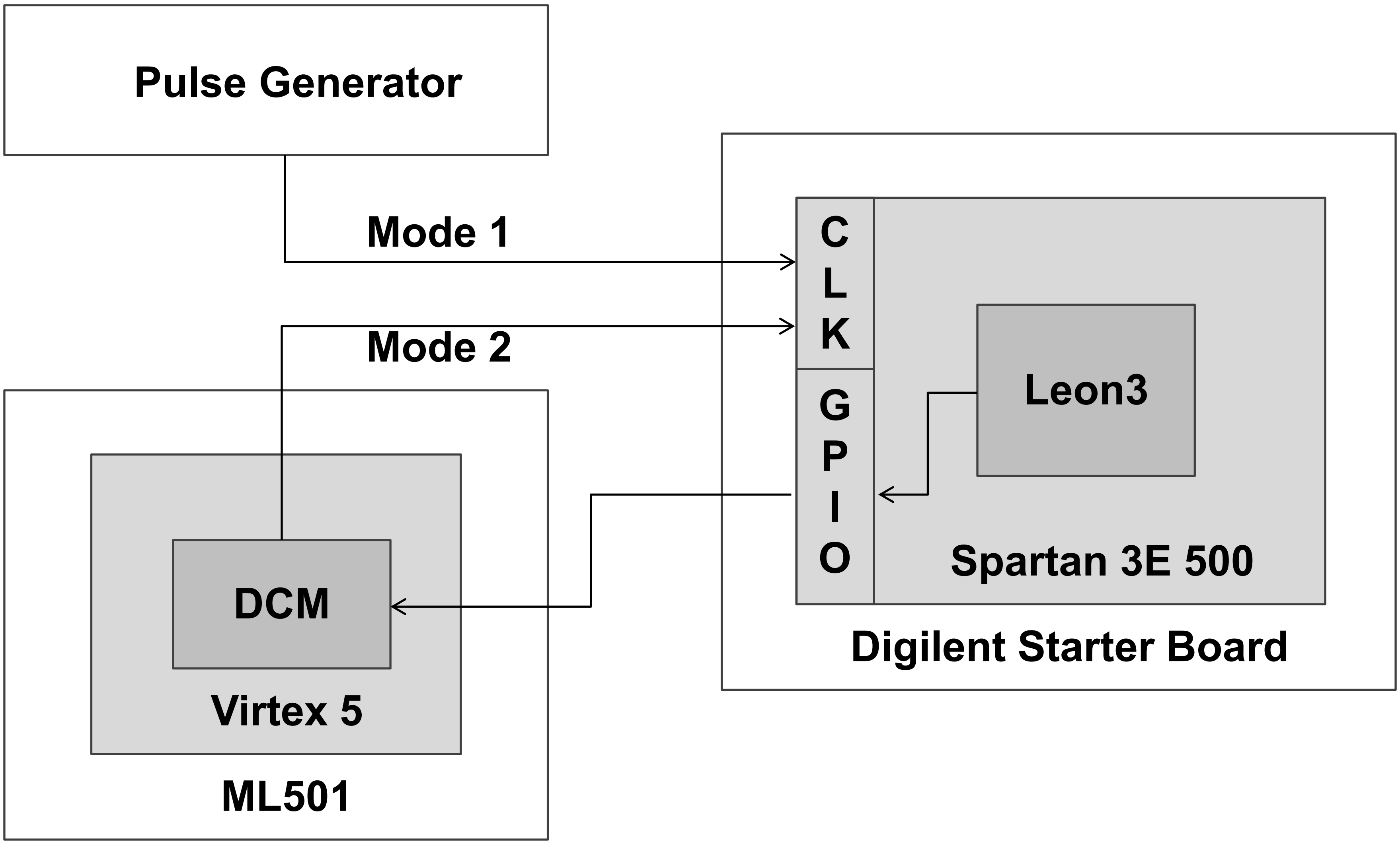
4. The chip generates the responses and sends back

5. If it matches with the stored one, the verifier authenticates the chip

Outline

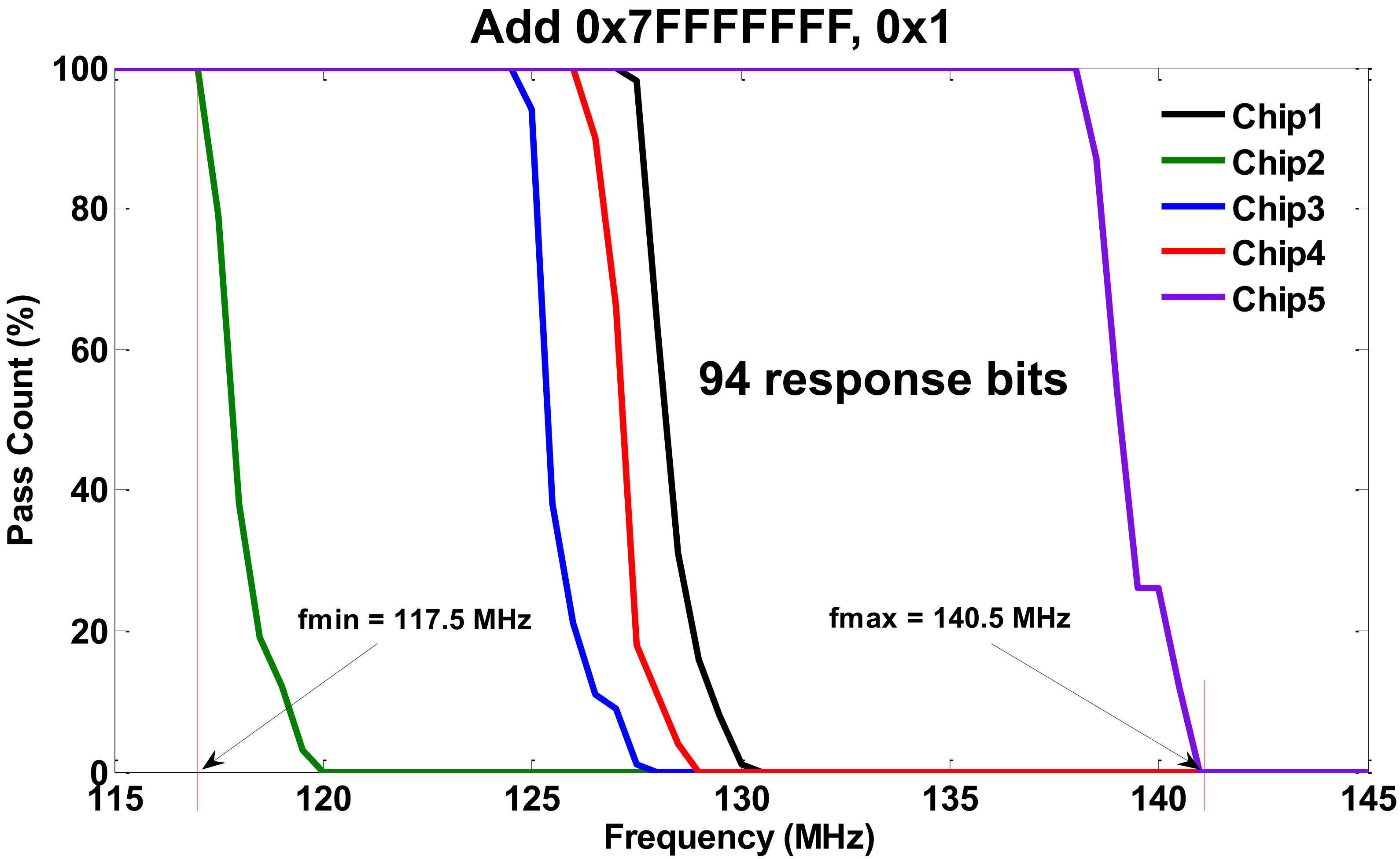
- What is a microprocessor-intrinsic PUF? Why do we need it ?
- Detail of the microprocessor-intrinsic PUF
- Results
- Conclusion and future work

Characterization Set Up



Measurements were taken at an interval of 0.5 MHz

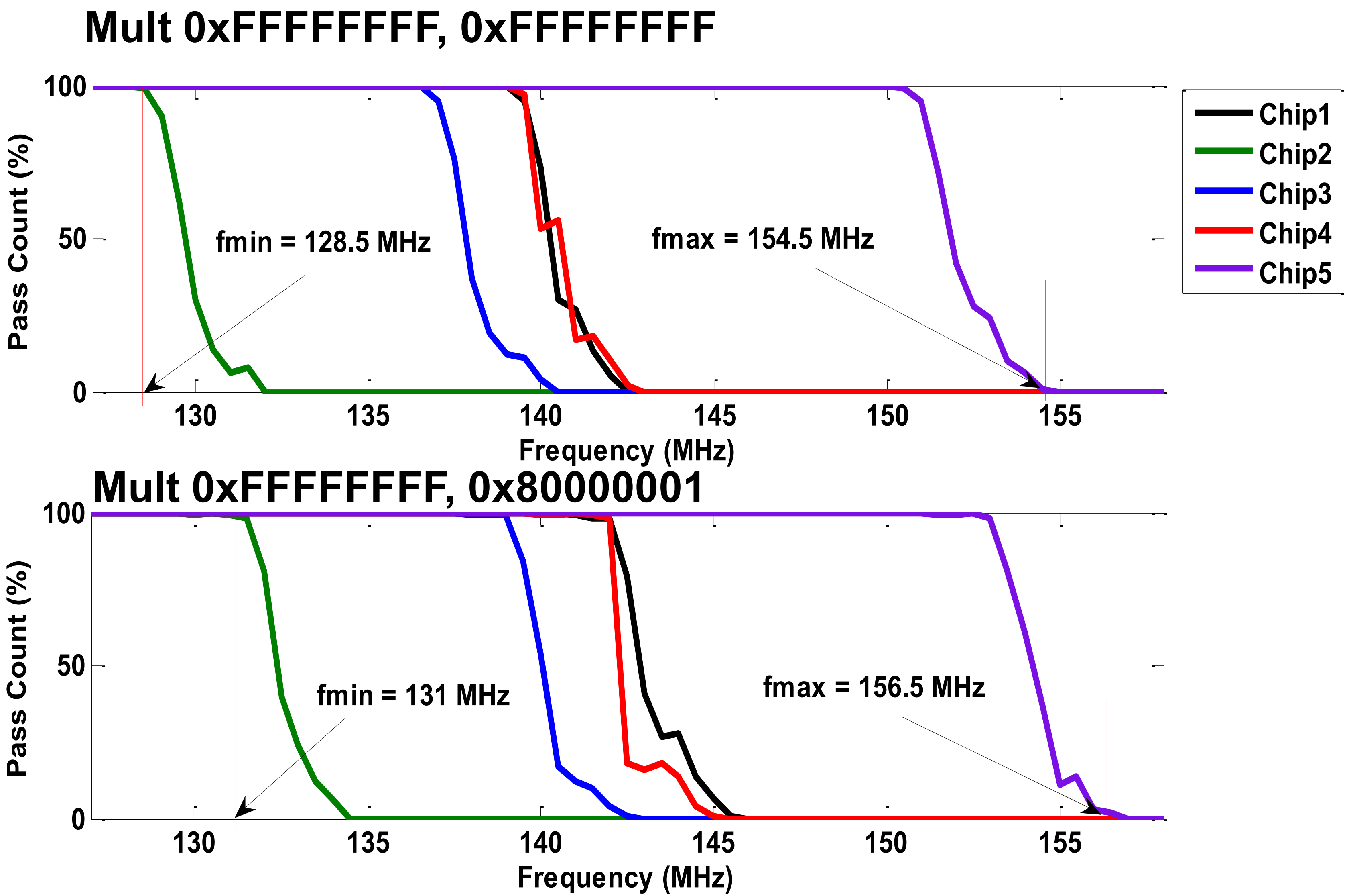
Results



No of sampling points = 1+ ((fmax – fmin) / sampling steps)

No of response bits = No of sampling points × 2

Results contd..



PUF Evaluation Parameter

Instruction	Operation	Uniqueness	Reliability
Addition	0x7FFFFFFF + 1	38.7 %	97.4 %
Multiplication	0xFFFFFFFF × 0xFFFFFFFF	36 %	98.1 %
	0xFFFFFFFF × 0x80000001	36.1 %	98 %
Division	0xFFFFFFFE00000001÷ 0xFFFFFFFF	38.1 %	99 %
	0x000000000000FA0 ÷ 0x00000014	37.3 %	95.6 %
Logic	0xFFFFFFFF AND 0xAAAAAAAA	36 %	99 %
Control	BGE	40.6 %	98.3 %

Outline

- **What is a microprocessor-intrinsic PUF? Why do we need it ?**
- **Detail of the microprocessor-intrinsic PUF**
- **Results**
- **Conclusion and future work**

Conclusion

- **Variability in a microprocessor pipeline can identify a chip.**
- **Multiplication and division instructions showed more variability and produced responses that are based on input operands.**
- **Uniqueness of the proposed PUF deviates from the ideal value. It needs further improvement.**
- **Though the PUF showed high reliability at normal operating condition, it needs to be tested under varying temperature and supply voltage.**

This research was supported in part through NSF grant no 0964680 and NSF grant no 0855095.

Thank you

Questions ??