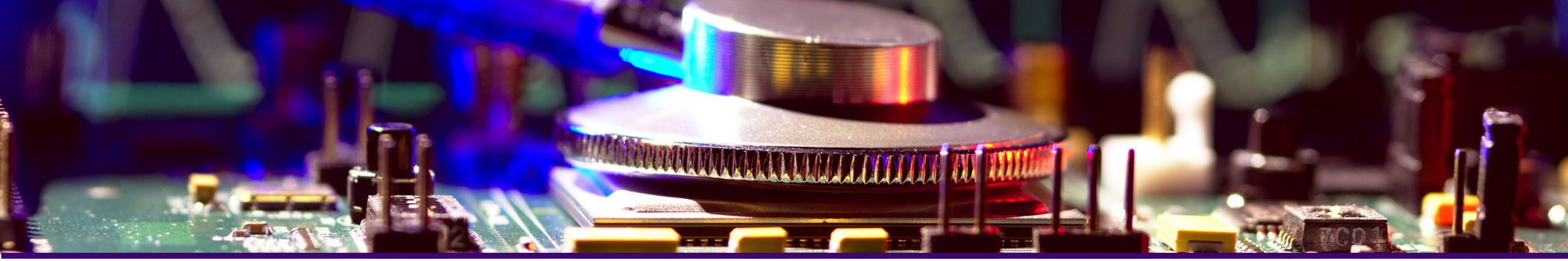


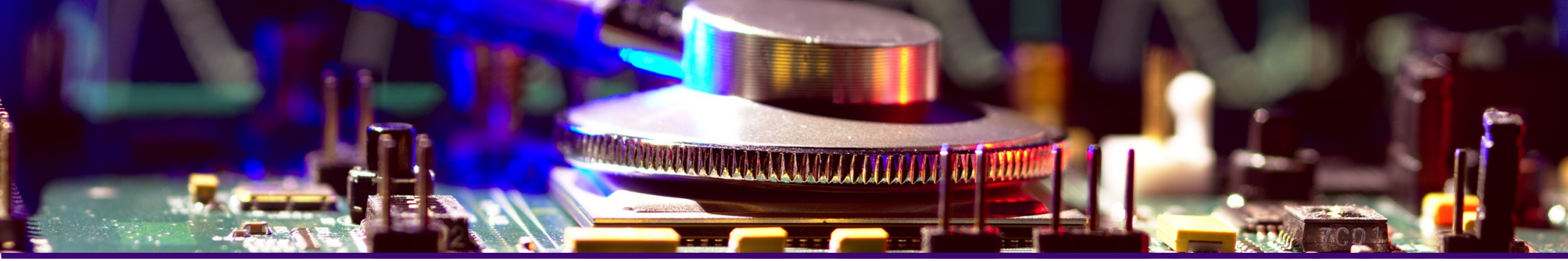
An Open-Source Platform for Evaluation of Hardware Implementations of Lightweight Authenticated Ciphers

Abubakr Abdulgadir*, William Diehl ** and Jens-Peter Kaps*

***Cryptographic Engineering Research Group - George Mason University**

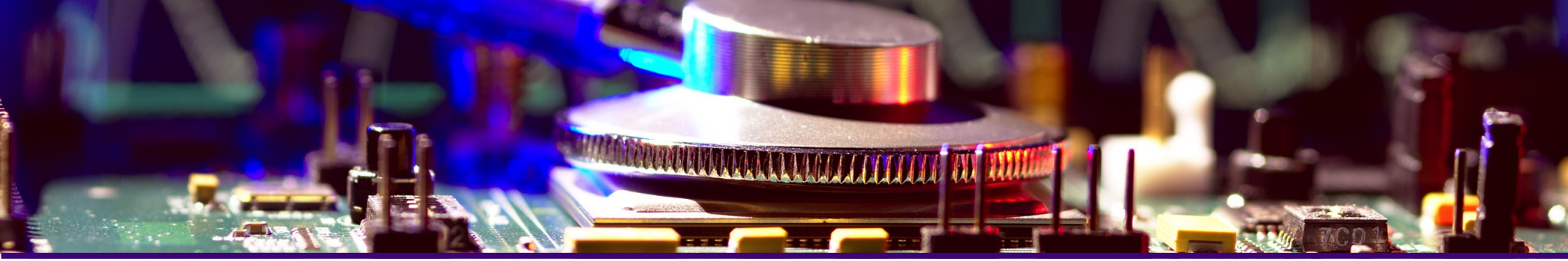
****Signatures Analysis Lab - Virginia Tech**

November - 2019



Overview

- Introduction
- Background
- Methodology
- Results



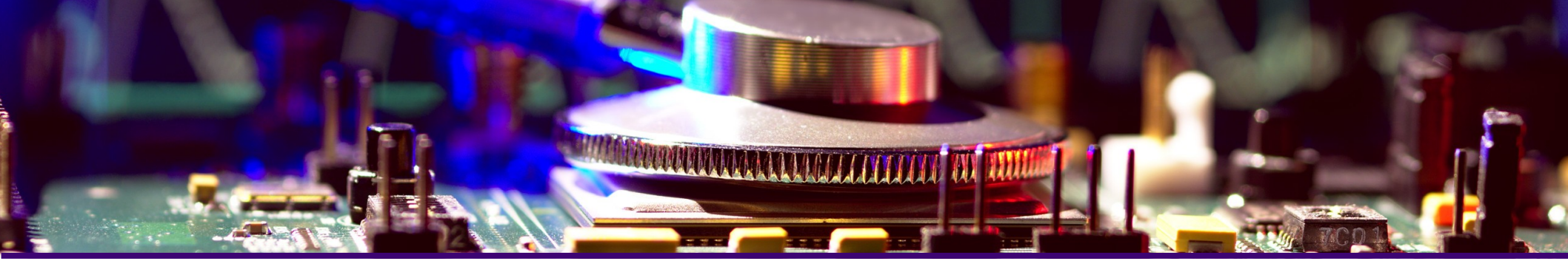
Introduction

Motivation

- NIST Lightweight Cryptography Evaluation Criteria:
  - Side-channel and fault resistance: Power side-channel, and others
  - Cost: Energy consumption, and others
  - Performance: Power consumption, and others
- Lightweight application are vulnerable to SCA.
- NIST Lightweight Standardization process.
  - 32 Round 2 candidates.
  - We need an efficient, easy to use side-channel analysis (SCA) platform.
- Existing solutions are either costly or need some work to adapt to LWC Hardware API.
- Save time!

Motivation

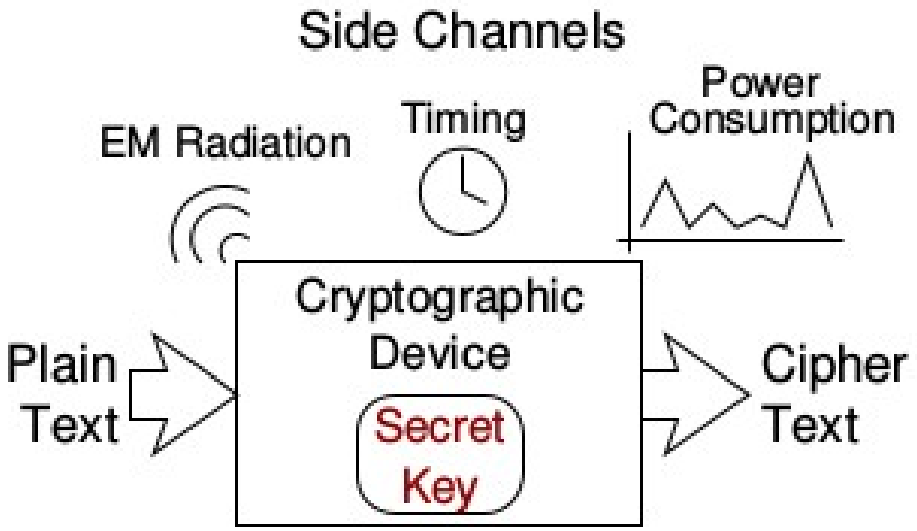
- Existing Solutions
  - Rambus DPA Workstation
  - Riscure Inspector
  - NewAE Chipwhisperer
  - SAKURA
  - Etc.
- We picked Flexible Opensource workBench fOr Side-channel analysis (FOBOS)
  - Already compatible with CAESAR Hardware API.
  - Needs speed improvement and new targets.



Background

Side-Channel Analysis

- A powerful method to extract secrets from cryptographic device.
- Power Side-Channel
  - Variability of power consumption leaks information about the secret.
- Some Variants
  - Simple power analysis (SPA)
  - Differential power analysis (DPA)
  - Correlation power analysis (CPA)



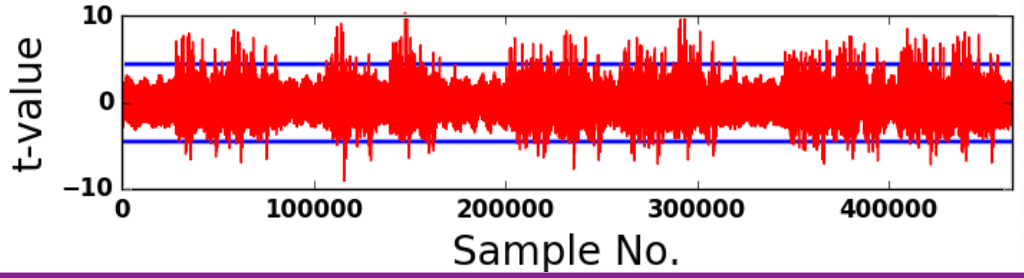
- Drawbacks
  - Requires power model.
  - Evaluates only one point of attack.
  - Inability to obtain the key does not guarantee that no sensitive information is leaked.

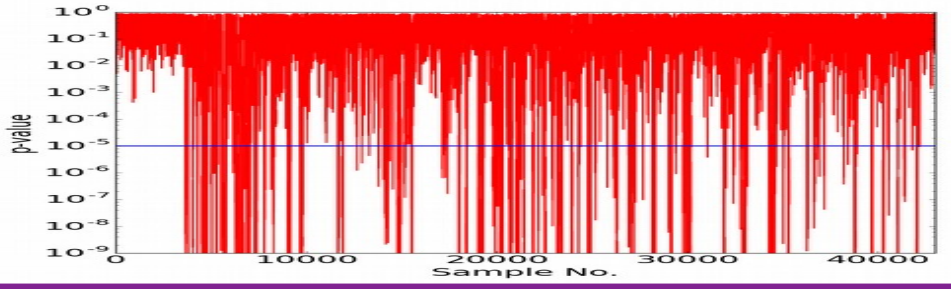
Leakage Assessment

- Covers the complete operation of a cipher quickly.
- If no leakage is detected, cipher implementation is secure.
- Drawback: Only tells the probability that information is leaking. Does not tell whether leak can be exploited to get sensitive information

- Welch's t-test
  - Test Vector Leakage Assessment
  - Shows difference of two populations
  - Secure if known indistinguishable from unknown

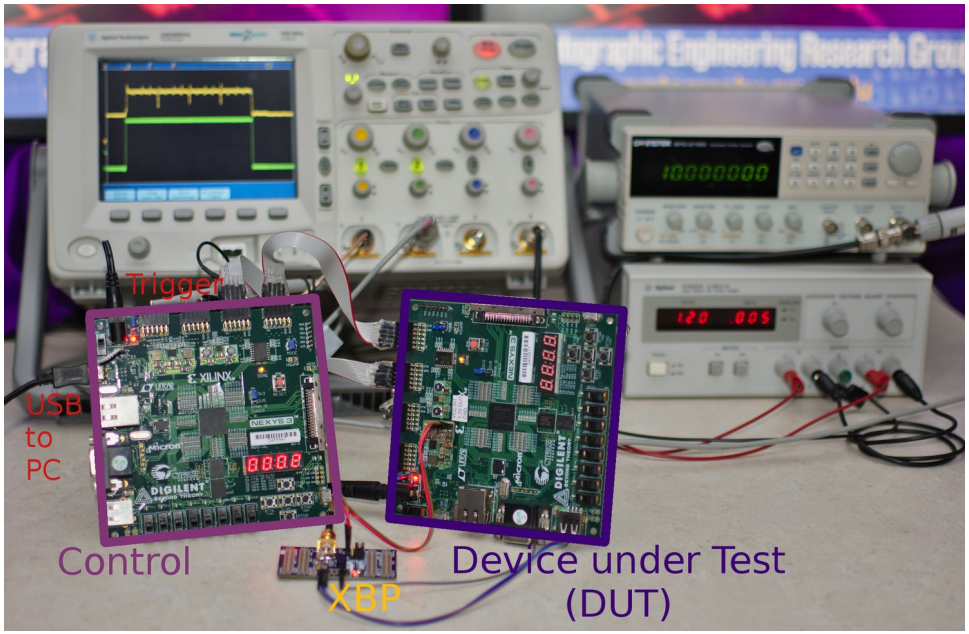
- Pearson's Chi-squared test
  - Complements Welch's t-test
  - Frequency of occurrence between classes





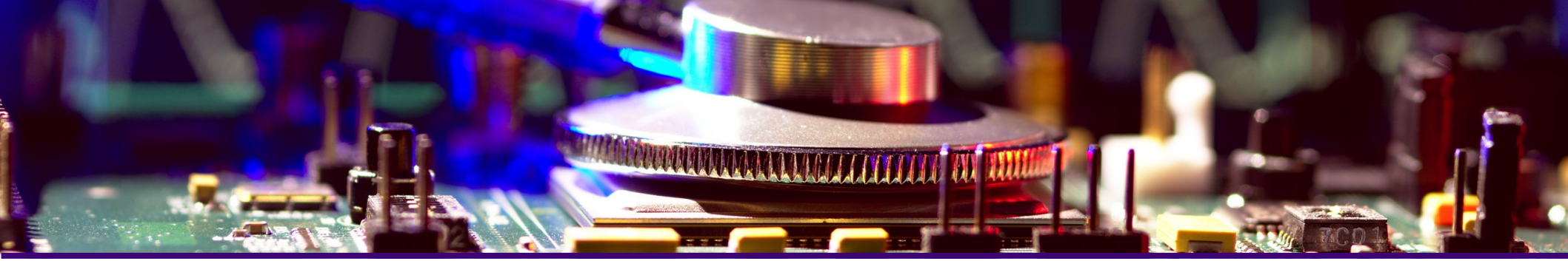
Introduction to FOBOS

- Flexible Opensource workBench fOr Side-channel analysis (FOBOS).
  - Loosely named after the Greek god Phobos (φόβος)
- Features
  - Complete “acquisition to analysis” platform for power analysis.
  - Control and Device under Test (DUT) on two different boards.
  - Uses commercially easily available boards.
  - Modular software in Python.
- Drawbacks
  - Slow: 2 AES traces per second



FOBOS Components

- FOBOS Data Acquisition
  - FOBOS Acquisition Hardware
    - Control board to interface with DUT.
    - DUT board and VHDL-wrapper for DUT.
  - FOBOS Acquisition Software
    - Controls FOBOS Acquisition Hardware.
    - Controls measurement equipment.
    - Stores measurements and setup information
- FOBOS Data Analysis
  - Statistics module
  - Post processing to reduce the amount of data to be evaluated.
  - Side-channel Distinguishers
  - Leakage Assessment

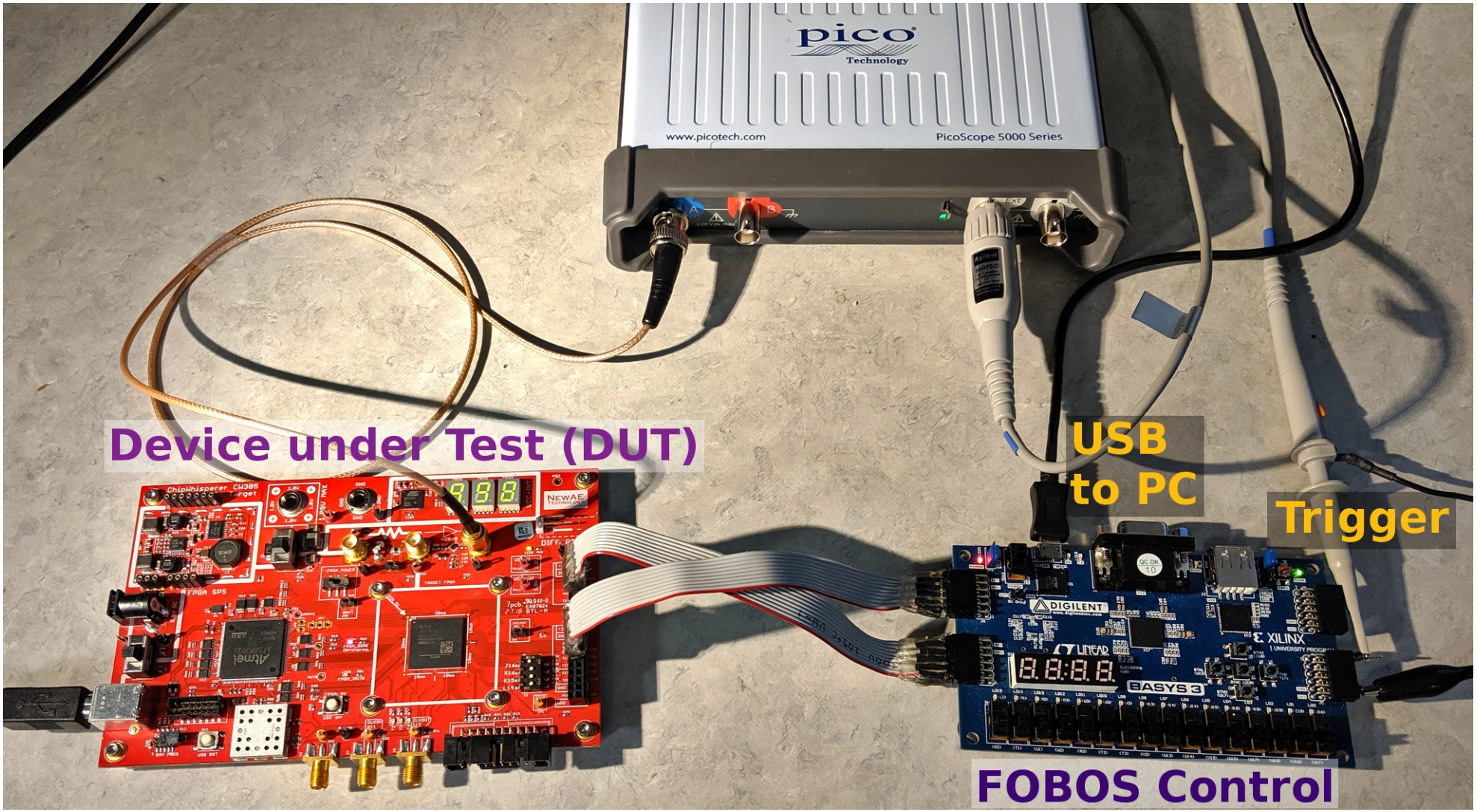


Methodology

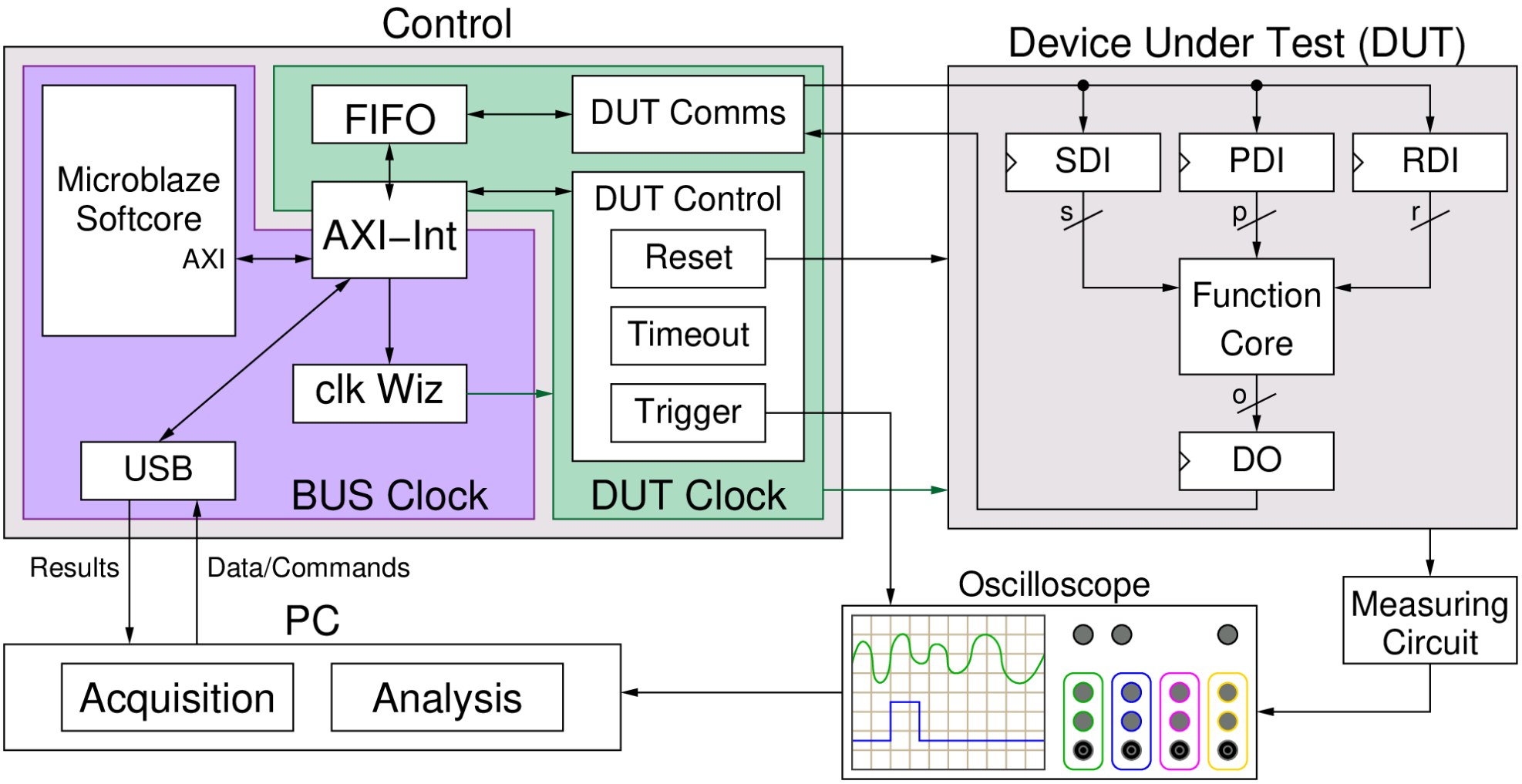
FOBOS 2

- Developed a new version of FOBOS with the following improvements:
  - Trace capture speed is improved (25x times)
  - Supports USB3-based oscilloscope (Picoscope).
  - Supports NewAE CW-305 Artix7-based DUT.
  - New control-board based on Digilent Basys3 has been developed. Using hardware-software codesign (Xilinx Microblaze controller).
  - New analysis scripts have been added such as the χ^2 -test script.

FOBOS 2 – Typical Setup



Acquisition Hardware



FOBOS 2 – Data Acquisition

- Software
  - Python scripts are provided to generate test vectors, configure the control board and the oscilloscope, and to run the attack / test.
- Control board features
  - Communication (PC, DUT)
    - PC ↔ Control via USB-UART
    - Control ↔ DUT via subset of AXI stream protocol
  - Triggering the oscilloscope
    - after a configurable number of clock cycles after DUT starts processing data.
  - Configurable DUT Reset
    - Useful to “abbreviate” run-time for first round attack.
  - Configurable Timeout
  - DUT clock generation (between 400 kHz and 100 MHz)

FOBOS 2 – Data Acquisition (contd.)

- DUT Board
  - VHDL provided for *Function Core* wrapper which handles all communication with *Control.*
  - Wrapper compatible with CAESAR Hardware API and LWC Hardware API.
  - Supported Boards
    - Digilent Nexys 3 (Xilinx Spartan 6) with some modifications.
    - NewAE CW-305 (Xilinx Artix 7) no modifications needed.
- Oscilloscope
  - Supported Oscilloscopes
    - Picoscope 5000 via USB 3
    - Agilent (Keysight) DSO6054A via Ethernet
    - Soon: Rigol 1000Z via Ethernet

Test Vectors

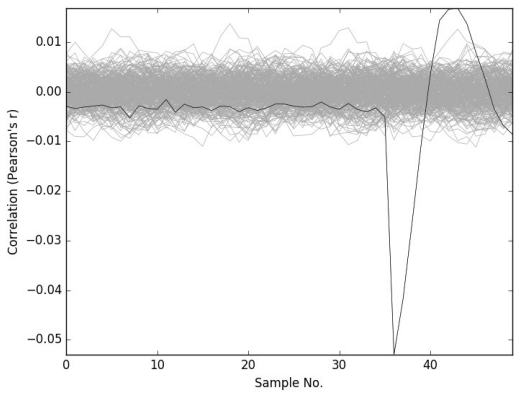
- Generate test vectors using:
  - Supplied *blockCipherTVGen.py* for block ciphers
  - *aeadTVgen* from the CAESAR Development Package
  - *lwcTVgen* from the LWC Development Package
- Test vector format example

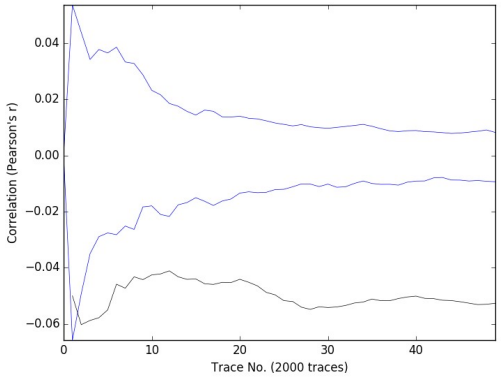
00c00010220b01d...**00c1**001029e5...**0081**0010**0080**0001

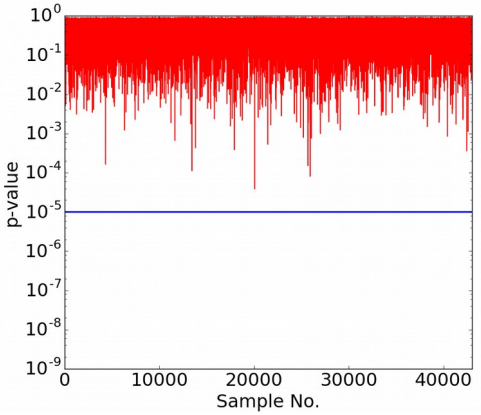
pdi, length, plaintext **SDI**, length,key, **exp_len**,len, **cmd**, start

FOBOS2 -Data Analysis module

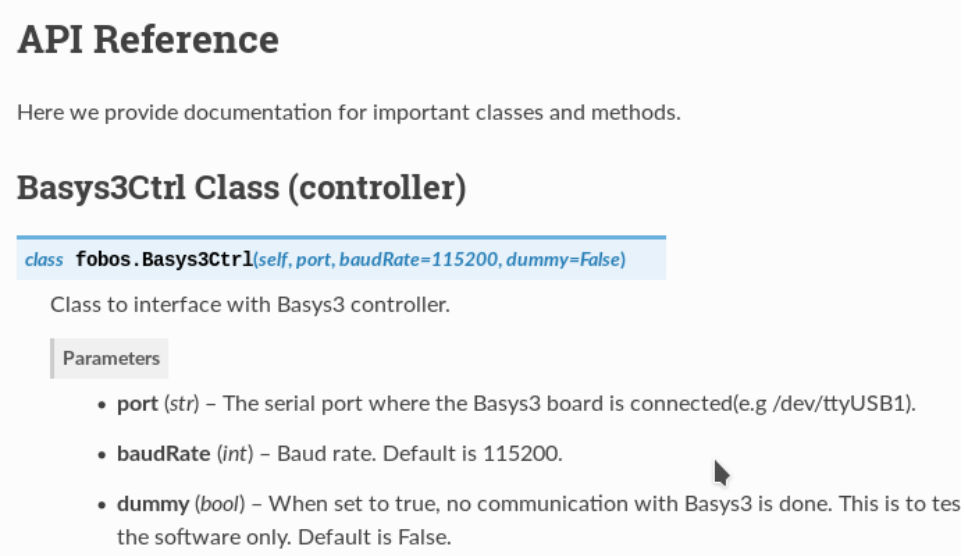
- Pre-processing
- Correlation Power Analysis (CPA)
- Leakage Assessment
  - T-test
  - Chi-square test
- Produces various graphs
  - Correlation
  - MTD
  - TVLA graph, chi-squared graph







Documented Software API



API Reference

Here we provide documentation for important classes and methods.

Basys3Ctrl Class (controller)

class **fobos.Basys3Ctrl**(*self, port, baudRate=115200, dummy=False*)

Class to interface with Basys3 controller.

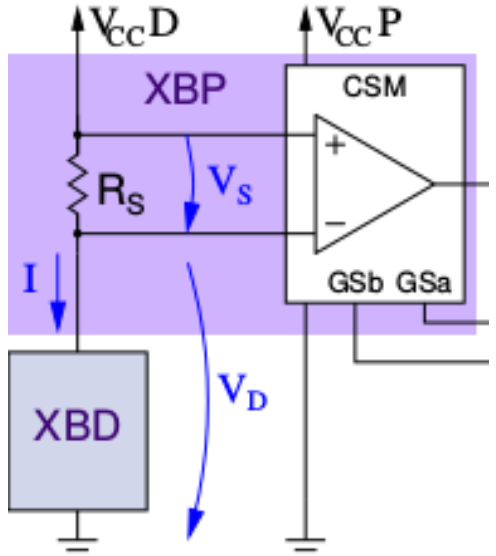
Parameters

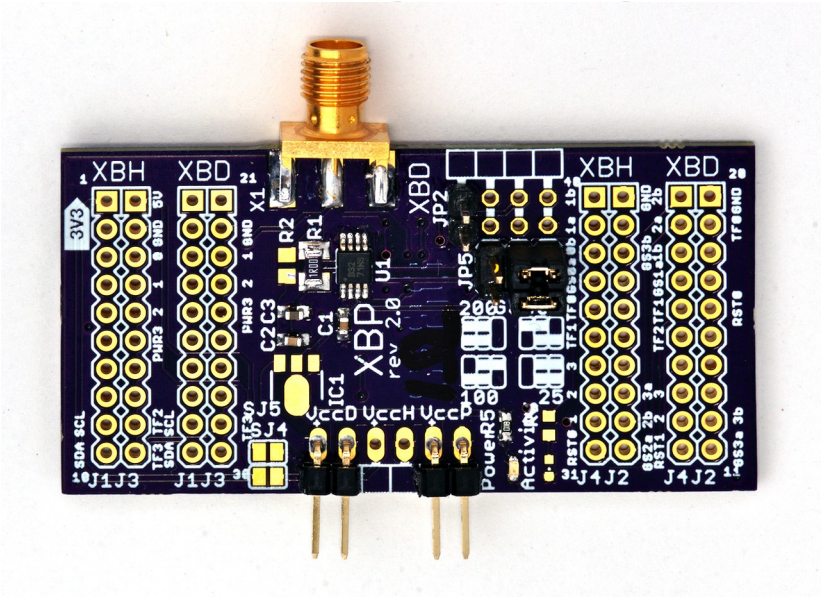
- **port** (*str*) – The serial port where the Basys3 board is connected(e.g /dev/ttyUSB1).
- **baudRate** (*int*) – Baud rate. Default is 115200.
- **dummy** (*bool*) – When set to true, no communication with Basys3 is done. This is to tes the software only. Default is False.

- Written in Python
  - Easy to use
  - Portable
  - Simple array manipulation (Numpy)

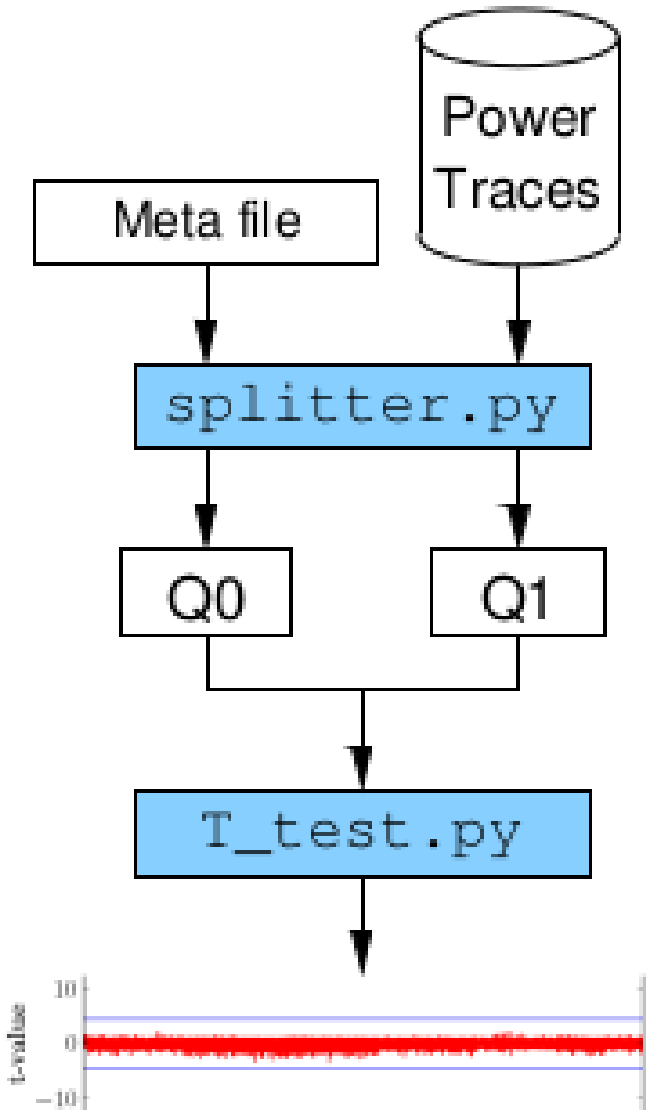
Power Measurements

- Uses XBP power shim
  - From the eXtended eXternal Benchmarking eXtension project
- Measure amplified Voltage across a shunt resistor
- We use a python script to calculate power in mW.

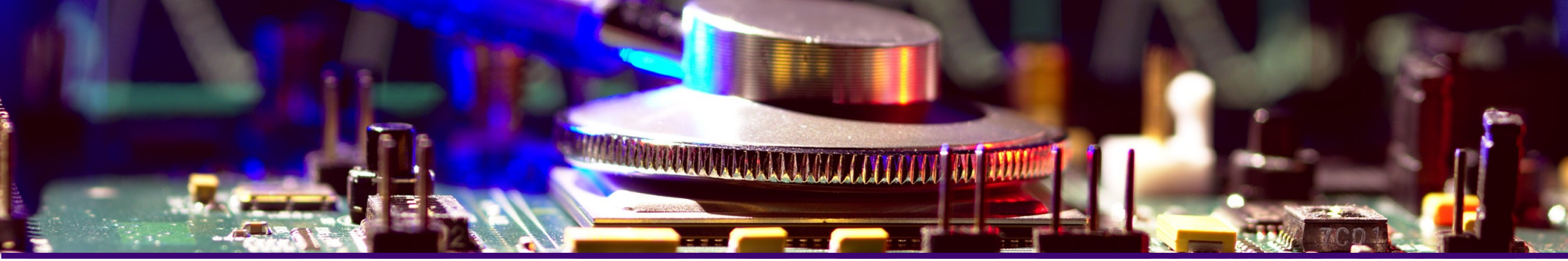




T-test leakage assessment



- Test vectors and meta file are generated.
- Traces collected.
- Analysis is provided with traces and meta file.
- Splitter.py splits power traces to Q0 and Q1.
- Chi-squared test flow is similar.



Results

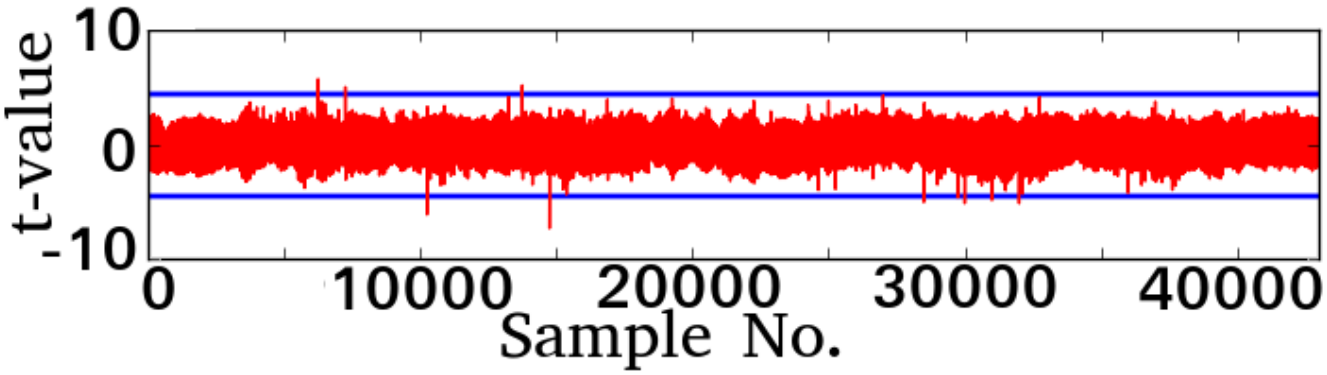
Results - TVLA

- We performed TVLA on:
  - Unprotected FPGA implementations of **Ascon** and **AES-GCM**
  - Protected (threshold implementation) of same ciphers.
- Collected 2000 traces (fixed-vs-random).
- DUT ran at 1 MHz.
- Sampled traces at 125 M Sample/sec.

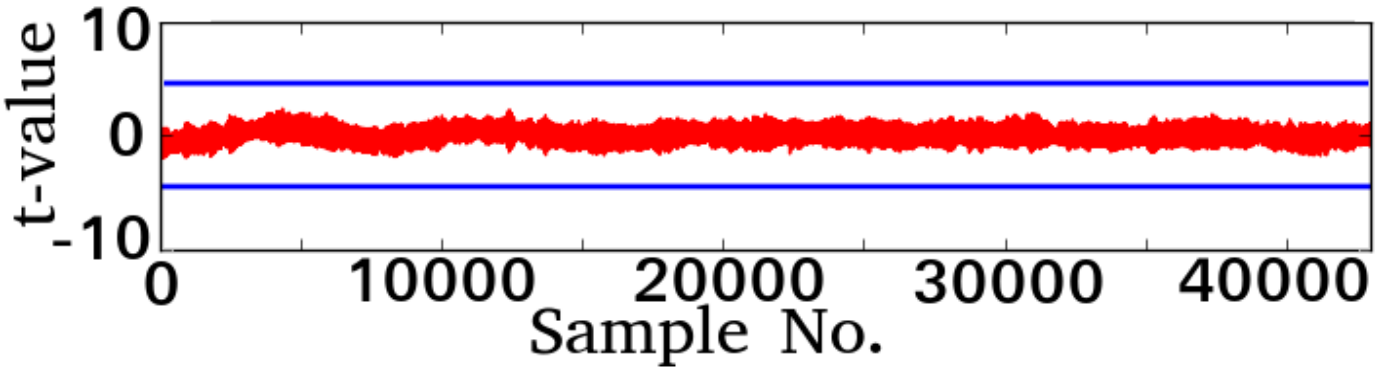
Results- T-test result in Artix7

- TVLA on Ascon unprotected and Ascon - protected (TI)
- Threshold selected at |t| = 4.5

Ascon - Unprotected



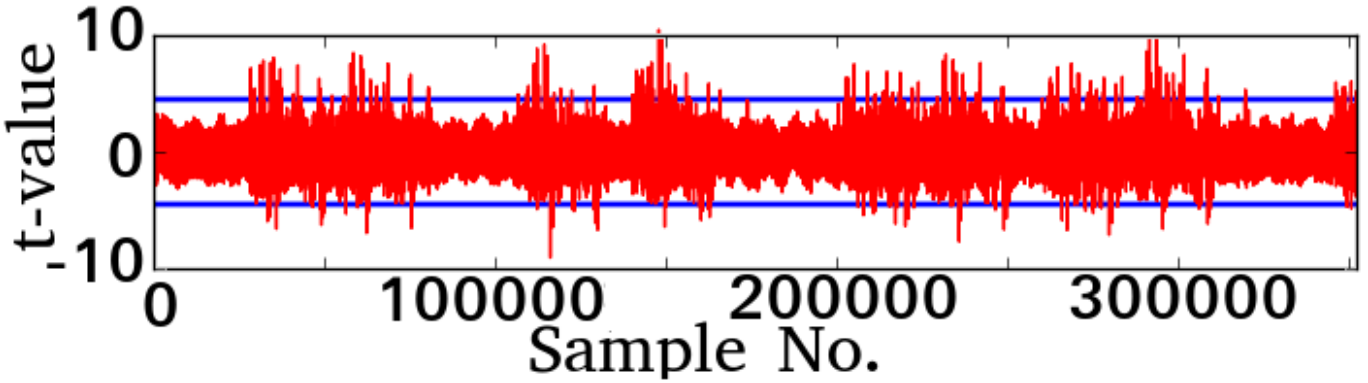
Ascon - Protected



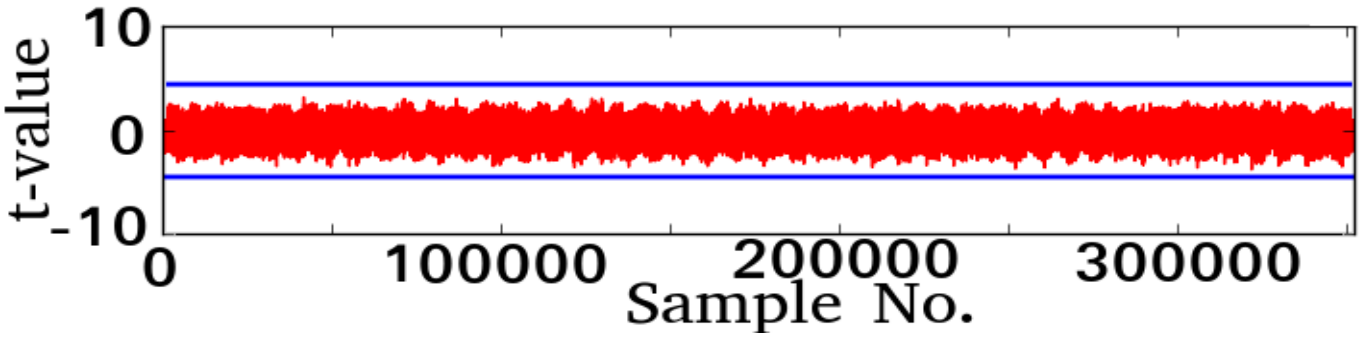
Results- T-test result in Artix7

- TVLA on AES-GCM unprotected and AES-GCM - protected (TI)
- Threshold selected at |t| = 4.5

AES-GCM - Unprotected

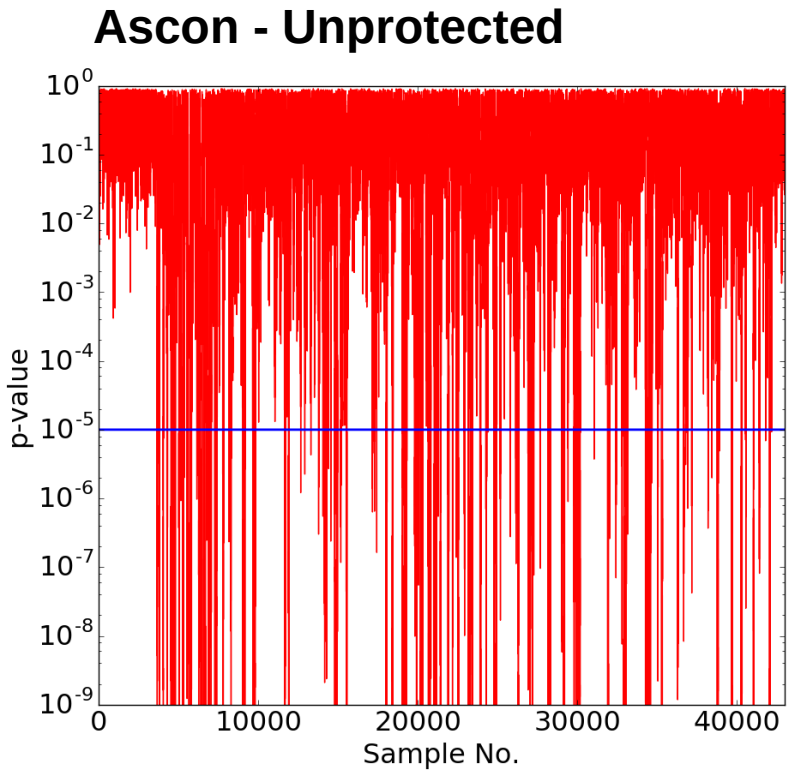


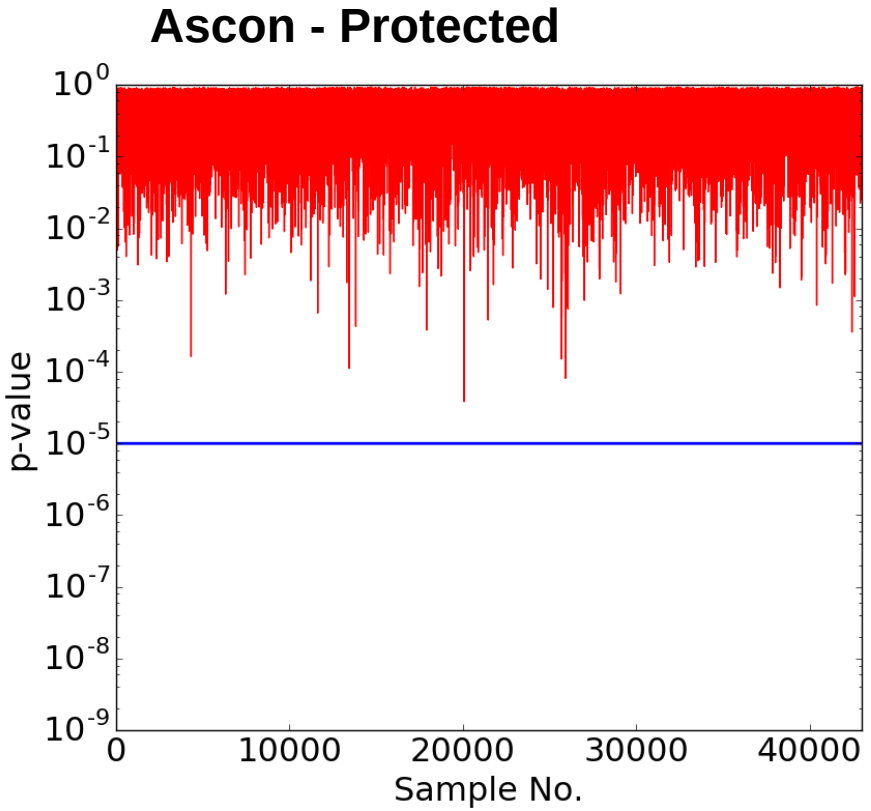
AES-GCM - Protected



Results -Chi-squared test-Spartan 6

- TVLA on Ascon unprotected and Ascon - protected (TI)
- Threshold selected at p = 10^-5
- Results confirm TVLA



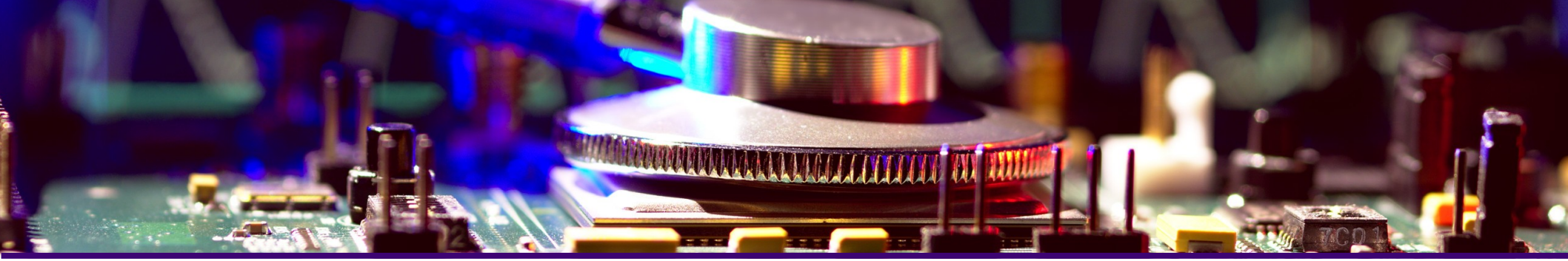


Results-Power and E/bit measurement

- Recently used to measure power an E/bit for 4 NIST LWC round-2 candidates
  - **Ascon**
  - **Spoc**
  - **Spook**
  - **GIFT-COFB**
- AES-GCM as benchmark.
- XBP was used for power measurements on NewAE CW305 (Artix7).

Conclusion

- FOBOS 2 is an efficient SCA platform for FPGA.
- Performs both acquisition and analysis.
- Uses commercially available boards when possible.
- Used for leakage assessment and power measurements.
- Download form https://cryptography.gmu.edu/fobos/



Thank you for listening

?

FOBOS 2 will be available at
https://cryptography.gmu.edu/fobos/