

# The SMAesH dataset

## Power leakage of a masked AES hardware implementation

Gaëtan Cassiers  and Charles Momin

UCLouvain, Louvain-la-Neuve, Belgium

**Abstract.** Datasets of side-channel leakage measurements are widely used in research to develop and benchmarking side-channel attack and evaluation methodologies. Compared to using custom and/or one-off datasets, widely-used and publicly available datasets improve research reproducibility and comparability. Further, performing high-quality measurements requires specific equipment and skills, while also taking a significant amount of time. Therefore, using publicly available datasets lowers the barriers to entry into side-channel research. This paper introduces the SMAesH dataset. SMAesH is an optimized masked hardware implementation of the AES with a provably secure arbitrary-order masking scheme. The SMAesH dataset contains power traces of the first-order SMAesH on two FPGAs of different generations, along with key, plaintext and masking randomness. A part of the dataset use uniformly random key and plaintext to enable leakage profiling, while another part uses a fixed key (still with uniformly random plaintext) to enable attack validation or leakage assessment in a fixed-versus-random setting. We document the experimental setup used to acquire the dataset. It is built from components that are widely available. We also discuss particular methods employed to maximize the information content in the leakage traces, such as power supply selection, fine-grained trace alignment and resolution optimization.

**Keywords:** Side-channel · Power leakage · Dataset · Masking · FPGA

## 1 Introduction

Physical side-channel attacks have been a threat to the security of embedded devices since their introduction [KJJ99]. Protecting a device against such attacks, as well as evaluating its security, is a challenging task, which led to the continuous improvement of countermeasures and attacks. In order to evaluate and compare attacks, the side-channel research community has been using public datasets of side-channel traces. Besides avoiding duplication of the measurement work, the usage of such datasets presents multiple advantages. First, this improves replicability of results, and therefore eases subsequent improvements of attacks. Further, the widespread use of public datasets in works designing new attacks makes their result much easier to compare. This also lowers the barrier to entry for the design of state-of-the-art attacks, sidestepping the need for equipment and skills to perform the measurements.

Many side-channel datasets have been introduced over the years, often in the context of side-channel attacks contests or challenges. Among these datasets, most of them correspond to fairly simple leakage structures such as non-protected implementations (such as AES\_HD [BJP20]), or present very strong leakage due to being software implementations (e.g., DPA contest v4<sup>1</sup>, ASCAD v1 [BPS<sup>+</sup>20] and v2 [MS21]). However, none of the widely

---

E-mail: [gaetan.cassiers@uclouvain.be](mailto:gaetan.cassiers@uclouvain.be) (Gaëtan Cassiers), [charles.momin@uclouvain.be](mailto:charles.momin@uclouvain.be) (Charles Momin)

<sup>1</sup><https://dpacontest.telecom-paris.fr/home/>



used datasets covers a protected hardware implementations the only ones covering this design corner being (to the best of our knowledge) the AES\_HD\_MM [Fei14] dataset and the Spook CTF [BBC<sup>+</sup>21] (CHES 2020 challenge) datasets. These datasets are not widely used, and it has been hypothesized that this is due to AES\_HD\_MM not containing random key traces (preventing profiled attacks) [PPM<sup>+</sup>23] and being broken within a low number of traces [WHJ<sup>+</sup>21], while the Spook dataset contains traces of an uncommon cipher (Clyde-128) [BBC<sup>+</sup>20, BBB<sup>+</sup>20].

**Contributions** We propose a new power leakage dataset of a masked hardware implementation of the AES: the SMAesH dataset. This dataset aims at being useful to the research community by being a realistic benchmark for attacks, while being not excessively difficult to attack. While the latter goal may seem surprising, we believe that very hard datasets have a limited interest to the research community: attacking such dataset requires many traces and large computational resources, which increases attack development iteration time and raises the barrier of entrance. In particular, care has been taken to ensure that, beyond the intrinsic security brought by masking, the dataset is as easy as possible to attack. Indeed, attacks requiring fewer traces are faster to run and require less computational resources (all other parameters being equal), and are therefore easier to work on.

These design goals are translated into the following dataset characteristics. First, the target is SMAesH, an open-source masked hardware implementation of the AES which is thoroughly documented. The implementation is reasonably simple to understand, but not artificially simple or unoptimized. Second, the security is concentrated on the masking countermeasure: we ensured that the masking has no big flaw, but did not add any other countermeasure such as added noise or clock jitter. Third, we optimized the acquisition setup parameters to maximize the amount of leakage, including power supply, clock signal, measurement device. Fourth, the acquisition setup uses devices that are easy to procure and is fully documented, with the aim of enabling its reproduction.

The SMAesH dataset contains measurements for two devices: a Xilinx Artix-7 FPGA (XC7A100T) on a Chipwhisperer CW305 board (next denoted A7), and a Xilinx Spartan-6 (C6SLX75) on a Sakura-G board (next denoted S6). The dataset contains a total of  $6 \times 2^{24}$  traces of 4250 samples each, covering the execution of the first round of the AES. The challenge associated to CHES 2023 was a side-channel analysis contest based on the SMAesH dataset. The attacks against the A7 target were in a worst-case setting (i.e., the design was fully public, including masking randomness for the profiling and validation data), and improved down to 290k traces. On the other hand, for the S6 target, the attack setting was more restricted (only the source code was public and not the bitstream, while the masking randomness was not public), which led to the best attack requiring 901k traces.

**Organization** We first introduce our target implementation SMAesH. We then discuss the high-level dataset structure, the configuration of the target chips, and the acquisition setup.

## 2 SMAesH

SMAesH<sup>2</sup> is an open-source masked hardware implementation of the AES. In version 1.0.0 (the one used for this dataset), the implementation is based on the Hardware Private Circuits (HPC2) masking scheme, which provides state-of-the-art guarantees in terms of resistance against physical defaults (e.g., glitches) and composability. It is therefore

<sup>2</sup><https://github.com/simple-crypto/SMAesH>

**Table 1:** SMAesH v2 datasets

Name <sup>a</sup>	Target	Original Role	Number of traces	Trace length ( $n_s$ )
SMAesH-A7_d2-vk0	Artix-7 ( $d = 2$ )	Profiling	$2^{24}$	4250
SMAesH-A7_d2-fk0		Validation		
SMAesH-A7_d2-fk1		Test		
SMAesH-S6_d2-vk0	Spartan-6 ( $d = 2$ )	Profiling	$2^{24}$	4400
SMAesH-S6_d2-fk0		Validation		
SMAesH-S6_d2-fk1		Test		

<sup>a</sup> The names are intended as unique dataset identifiers. Future dataset versions will not reuse the same dataset names (except when re-packaging the exact same data).

provably secure at arbitrary masking order and has been formally verified by the fullVerif tool. SMAesH relies on a pipelined 32-bit architecture: 4 masked S-boxes are instantiated. In each round, they are used to process the key bytes, then to process the 16 AES state bytes (in four chunks corresponding to MixColumns operations), for a total encryption latency of 105 clock cycles per block. The randomness needed for masking is generated by a PRNG based on an instance of Trivium. SMAesH has been designed to be independent of the implementation technology: its RTL representation uses only flip-flops (clocked on the positive edge of a clock) and combinational logic, it is therefore suited to both ASIC and FPGA implementations.

The synthesis parameters of SMAesH are  $D$ , the number of shares and `PRNG_MAX_UNROLL`, a limit on the unrolling of the Trivium PRNG (which allows adjusting the critical path vs area trade-off), as proposed in [CMM<sup>+</sup>24]. We used  $D=2$  and `PRNG_MAX_UNROLL=128` (leading to the instantiation of two Trivium cores).

### 3 Dataset

Two datasets have been acquired for each target:

- A training dataset that uses a fresh random key for each trace.
- A validation dataset that uses a single key for the whole dataset.

All datasets use a fresh random plaintext for each trace and make a correct use of the SMAesH core: for each trace, the sharing of the key and of the plaintext is fresh. Moreover, although not necessary for security, we reseed the core before each trace with a fresh seed (the reseeding is not included in the trace), in order to ease the simulation of the computations performed by the target. The dataset characteristics as summarized in Table 1, The dataset is available at [CM23].

For each trace, the datasets contain the power leakage trace, as well as all the data required to exactly replicate the measured execution, as shown in Table 2 (the `umsk_plaintext` and `umsk_key` can be derived from `msk_plaintext` and `msk_key` respectively, they are provided for convenience only).

## 4 Target designs

### 4.1 Artix-7

The FPGA bitstream used to perform the acquisitions has been generated using the Xilinx Vivado Toolset (v2022.1 64-bit) and the following modifications have been applied

**Table 2:** Dataset fields

Label	Type	Length	Description
<code>traces</code>	<code>int16</code>	$n_s$	Power trace.
<code>umsk_plaintext</code>	<code>uint8</code>	16	Non-masked plaintext.
<code>umsk_key</code>	<code>uint8</code>	16	Non-masked key.
<code>msk_plaintext</code>	<code>uint8</code>	$16d$	Plaintext shares (each 16-byte chunk is a share).
<code>msk_key</code>	<code>uint8</code>	$16d$	Key shares (each 16-byte chunk is a share).
<code>seed</code>	<code>uint8</code>	10	PRNG seed.

compared to the default toolflow parameters:<sup>3</sup>

- HDL annotation:
  - attribute `DONT_TOUCH` set for every module.
  - attribute `KEEP_HIERARCHY` set for every module.
- Synthesis parameters:
  - `flatten_hierarchy` set to `none`
  - `gated_clock_conversion` set to `off`
  - `bufg` set to 12
  - `directive` set to `Default`
  - `no_retiming` checked
  - `fsm_extraction` set to `auto`
  - `keep_equivalent_registers` checked
  - `ressource_sharing` set to `off`
  - `no_lc` checked
  - `no_srlextract` checked
- Implementation parameters:
  - `opt_design` related: `is_enabled` unchecked
  - `phys_opt_design` related: `is_enabled` unchecked

The acquisition setup programs the microcontroller of the CW305 board with a tweaked version of the newAE-provided firmware. The changes increase the flexibility in data sent to the FPGA (such as seeds, shared values, etc.) and add the ability to perform quick interleaved acquisition of multiple datasets. In more details, the microcontroller can run a full “batch” of traces without interaction with the control computer. At each trace, the microcontroller selects randomly a dataset to which the trace belongs (in our case, `-fk0` or `-vk0`), generates the required inputs and sends them to the FPGA. All the operations are performed in constant time, such that the FPGA traces are as similar to each other as possible. The randomized interleaving ensures that there is no systematic bias in the datasets, which allows using them for a fixed-vs-random TVLA. All the randomness of the microcontroller is drawn from a PRNG seeded by the control computer, which allows it to reconstitute the data sent to the FPGA.

<sup>3</sup>The Vivado project used to generate the bitstream is available at [https://github.com/simple-crypto/SMAesH-challenge/tree/main/fpga\\_designs/A7](https://github.com/simple-crypto/SMAesH-challenge/tree/main/fpga_designs/A7).

## 4.2 Spartan-6

# 5 Experimental setup

## 5.1 Artix-7

The power traces have been acquired by measuring the signal at the X4 point (directly connected to the oscilloscope with a SMA cable), which corresponds to the voltage drop across the  $100\text{ m}\Omega$  shunt resistor R27 amplified with a low-noise amplifier. The target FPGA is powered through the dedicated banana connectors (with the SW1 accordingly set) by an external low noise power supply Keysight E36102B set to a 1 V DC voltage. This setup reduces the noise level compared to the on-board power supply derived from the 5 V USB supply by a switching voltage converter.

The leakage is measured by a PicoScope 6242E digital oscilloscope. The target FPGA and oscilloscope clocks are synchronized in order to reduce the level of noise induced by clock jitter. This is achieved by configuring the CDCE906 PLL of the CW305 board to generate two clocks signals based (derived from the 12 MHz crystal of the CW305). The first is the FPGA clock, running at 1.5625 MHz generated by the PLL1 and fed to the port N13 on the FPGA. The second is a 10 MHz signal generated by the PLL0 and fed routed to the X6 SMA connector. It is then forwarded to the PicoScope 10 MHz clock reference input port. A single measurement channel (channel A) is used to perform the measurement and the trigger signal is fed from the onboard test point TP1 to the oscilloscope AUX trigger port with a PicoScope probe (Picotech TA386, 1:1 ratio, 200MHz of bandwidth). The power traces are sampled at 5 GS/s using a vertical resolution of 10 bits.

The clock configuration aims at minimizing the jitter between the target FPGA clock and the oscilloscope sampling clock. In particular, the configured frequencies result in exactly 3200 samples per target clock cycle, and the relative phase of the clocks are fixed by the PLLs. While this setup has been observed to give a low clock jitter and therefore an excellent alignment of consecutive traces, the relative phase of the clocks may drift over time, resulting in slight misalignments of the traces when measured over a long time span. This drift has been mitigated by increasing the sampling frequency to 5 GS/s, which is beyond the useful signal bandwidth, but limits the possible clock drift to 200 ps. The signal is then down-sampled with a moving average filter, with a decimation factor of 16, leading to a final sampling frequency of 312.5 MS/s, an a vertical resolution of 14 bits.

The target clock frequency has been selected as the largest sub-multiple of the oscilloscope sample rate for which the leakage caused by both edges of the clock can be clearly identified in the leakage trace, as recommended by [BUS21]. Then, the oscilloscope sample rate is selected based on the perceived information [BHM<sup>+</sup>19, MCHS23] on the shares: in order to minimize the dataset size, we selected the lowest sample rate that does not result in significantly degraded information content.

The acquisition has been performed in a room without accurate temperature control, but with relatively stable temperature (air conditioning) and no direct sunlight. The total duration for the acquisition of the Artix-7 datasets is 21 hours each (224 traces/s). In order to minimize the dataset size, the samples selected in the final dataset cover only the leakage of the first round of the AES (this has been determined by means of SNR computation on all the variables of the first round: all the SNR peaks are kept in the trace).

## 5.2 Spartan-6

The acquisition setup for the Spartan-6 target is similar to the one for the Artix-7 target. We discuss the few differences below.

The datasets contain power traces that have been acquired by measuring the voltage drop across a  $2\Omega$  shunt resistor placed at on JP2. This voltage drop is amplified by the on-board amplifier, and measured at the J3 point through a SMA cable. The whole Sakura-G board is powered by an external low noise power supply Keysight E36102B set to a 5 V DC voltage.

Regarding the oscilloscope configuration, the external trigger signal is connected to a GPIO header of the board. Further, the clock synchronization is achieved by generating the clock signal with the signal generator of the oscilloscope (1.5625 MHz square wave, 2V peak-to-peak amplitude, 1V offset) and feeding it directly to the SMA connector J6P on the board (i.e., clock-capable SMA for controller FPGA). This signal is directly used as a clock by the target (clock buffers are used in the design, but no PLL is used). Finally, the oscilloscope performs acquisition at 1.25 GS/s (resulting in 800 samples per target clock cycle) using a vertical resolution of 12 bits, and the post-processing uses a decimation factor of 4 to reduce this to 312.5 MS/s with 14-bit vertical resolution.

The clock generation method used for the Spartan-6 eliminates the drift observed on the Artix-7 acquisitions, which can be explained by the absence of any PLL between the oscilloscope acquisition clock and the target clock. Therefore, a very high sample rate was not needed, and a lower sample rate was used (along with a higher vertical resolution) in the interest of a faster acquisition. After downsampling, both targets' datasets have the same sample rate and resolution.

**Acknowledgements** Gaëtan Cassiers is a Postdoctoral Researcher of the Belgian Fund for Scientific Research (FNRS-F.R.S.). This work has been funded in parts by European Union through the ERC Advanced Grant 101096871 (BRIDGE). Views and opinions expressed are those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

## References

- [BBB<sup>+</sup>20] Davide Bellizia, Francesco Berti, Olivier Bronchain, Gaëtan Cassiers, Sébastien Duval, Chun Guo, Gregor Leander, Gaëtan Leurent, Itamar Levi, Charles Momin, Olivier Pereira, Thomas Peters, François-Xavier Standaert, Balazs Udvarhelyi, and Friedrich Wiemer. Spook: Sponge-based leakage-resistant authenticated encryption with a masked tweakable block cipher. *IACR Trans. Symmetric Cryptol.*, 2020(S1):295–349, 2020.
- [BBC<sup>+</sup>20] Davide Bellizia, Olivier Bronchain, Gaëtan Cassiers, Vincent Grosso, Chun Guo, Charles Momin, Olivier Pereira, Thomas Peters, and François-Xavier Standaert. Mode-level vs. implementation-level physical security in symmetric cryptography - A practical guide through the leakage-resistance jungle. In *CRYPTO (1)*, volume 12170 of *Lecture Notes in Computer Science*, pages 369–400. Springer, 2020.
- [BBC<sup>+</sup>21] Davide Bellizia, Olivier Bronchain, Gaëtan Cassiers, Charles Momin, François-Xavier Standaert, and Balazs Udvarhelyi. Spook SCA CTF, 2021. [doi: 10.14428/DVN/W2SV5G](https://doi.org/10.14428/DVN/W2SV5G).
- [BHM<sup>+</sup>19] Olivier Bronchain, Julien M. Hendrickx, Clément Massart, Alex Olshevsky, and François-Xavier Standaert. Leakage certification revisited: Bounding model errors in side-channel security evaluations. In *CRYPTO (1)*, volume 11692 of *Lecture Notes in Computer Science*, pages 713–737. Springer, 2019.

- [BJP20] Shivam Bhasin, Dirmanto Jap, and Stjepan Picek. AES HD dataset - 500 000 traces. AISyLab repository, 2020. [https://github.com/AISyLab/AES\\_HD\\_2](https://github.com/AISyLab/AES_HD_2).
- [BPS<sup>+</sup>20] Ryad Benadjila, Emmanuel Prouff, Rémi Strullu, Eleonora Cagli, and Cécile Dumas. Deep learning for side-channel analysis and introduction to ASCAD database. *J. Cryptogr. Eng.*, 10(2):163–188, 2020.
- [BUS21] Davide Bellizia, Balazs Udvarhelyi, and François-Xavier Standaert. Towards a better understanding of side-channel analysis measurements setups. In *CARDIS*, volume 13173 of *Lecture Notes in Computer Science*, pages 64–79. Springer, 2021.
- [CM23] Gaëtan Cassiers and Charles Momin. Power leakage traces of smaesh, October 2023. doi:10.3217/bk4fx-rbh46.
- [CMM<sup>+</sup>24] Gaëtan Cassiers, Loïc Masure, Charles Momin, Thorben Moos, Amir Moradi, and François-Xavier Standaert. Randomness generation for secure hardware masking - unrolled trivium to the rescue. *IACR Commun. Cryptol.*, 1(2):4, 2024.
- [Fei14] Yunsi Fei. Northeastern university tescase dataset, 2014. URL: [https://chest.coe.neu.edu/?current\\_page=POWER\\_TRACE\\_LINK&software=ptmasked](https://chest.coe.neu.edu/?current_page=POWER_TRACE_LINK&software=ptmasked).
- [KJJ99] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In *CRYPTO*, volume 1666 of *Lecture Notes in Computer Science*, pages 388–397. Springer, 1999.
- [MCHS23] Loïc Masure, Gaëtan Cassiers, Julien M. Hendrickx, and François-Xavier Standaert. Information bounds and convergence rates for side-channel security evaluators. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2023(3):522–569, 2023.
- [MS21] Loïc Masure and Rémi Strullu. Side channel analysis against the anssi’s protected AES implementation on ARM. *IACR Cryptol. ePrint Arch.*, page 592, 2021.
- [PPM<sup>+</sup>23] Stjepan Picek, Guilherme Perin, Luca Mariot, Lichao Wu, and Lejla Batina. Sok: Deep learning-based physical side-channel analysis. *ACM Comput. Surv.*, 55(11):227:1–227:35, 2023.
- [WHJ<sup>+</sup>21] Yoo-Seung Won, Xiaolu Hou, Dirmanto Jap, Jakub Breier, and Shivam Bhasin. Back to the basics: Seamless integration of side-channel pre-processing in deep neural networks. *IEEE Trans. Inf. Forensics Secur.*, 16:3215–3227, 2021.