Friedrich-Alexander-Universität
Technische Fakultät

Semi-Automatic Locating of Cryptographic Operations in Side-Channel Traces

CHES 1/2022

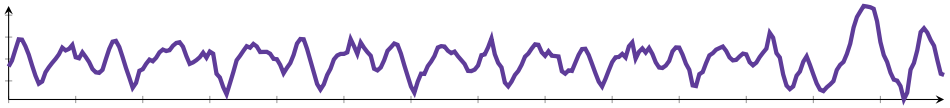
Jens Trautmann[1], Arthur Beckers[2], Lennert Wouters[2], Benedikt Gierlichs[2], Stefan Wildermann[1], Ingrid Verbauwhede[2], and Jürgen Teich[1]
[1]Friedrich-Alexander-Universität Erlangen-Nürnberg, Hardware-Software-Co-Design, [2]imec-COSIC, KU Leuven

September 19, 2022

Motivation

Side-Channel Analysis

- **Side-Channel Attacks**: iPhones [Lis+21], Single Board Computers [Bal+15], IoT devices [Ron+17]
- **Goals**: Retrieve keys, implementation details, ...
- **Requirements**: Side-channel recordings of Cryptographic Operations (COs), associated plaintext/ciphertext

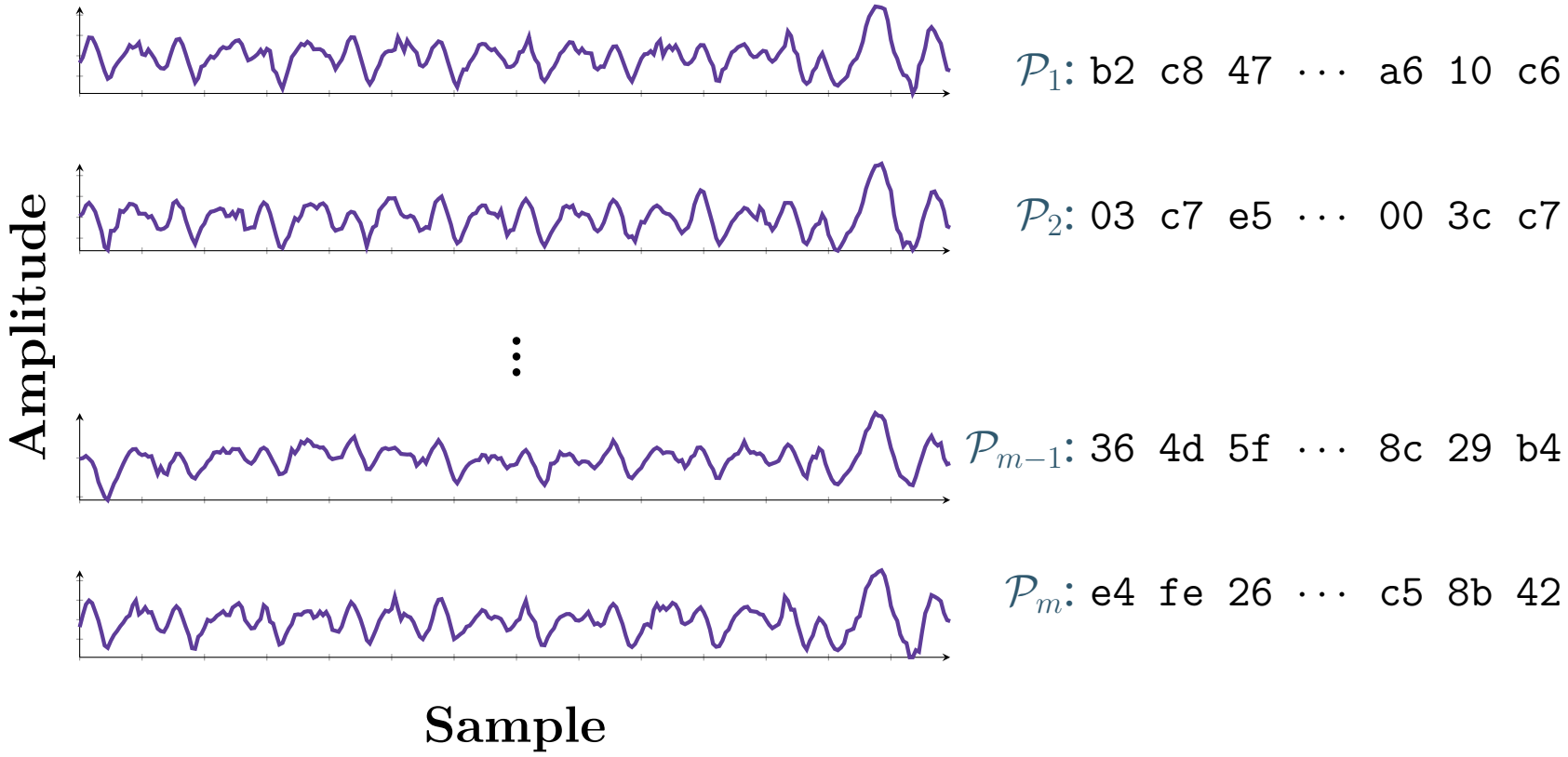


$\mathcal{P}_1$: `b2 c8 47 ··· a6 10 c6`

Motivation

Side-Channel Analysis

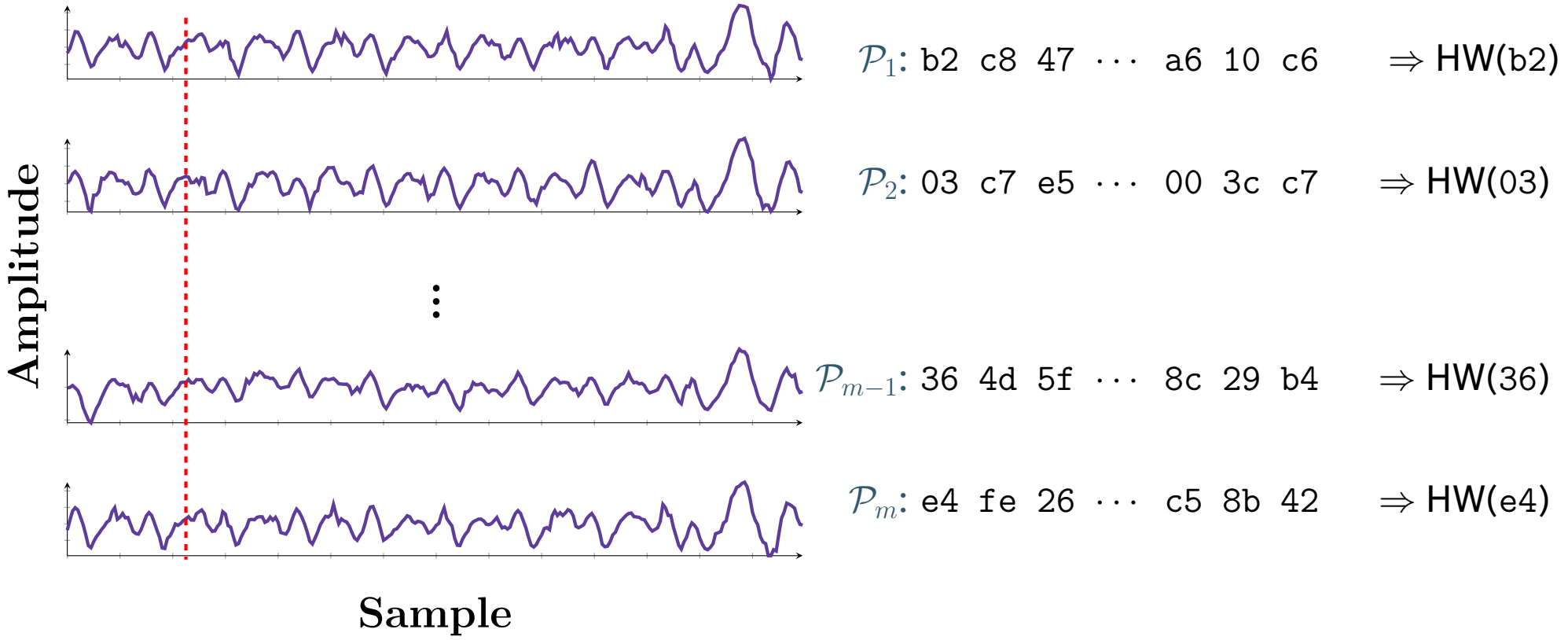
- **Side-Channel Attacks**: iPhones [Lis+21], Single Board Computers [Bal+15], IoT devices [Ron+17]
- **Goals**: Retrieve keys, implementation details, ...
- **Requirements**: Side-channel recordings of Cryptographic Operations (COs), associated plaintext/ciphertext



Motivation

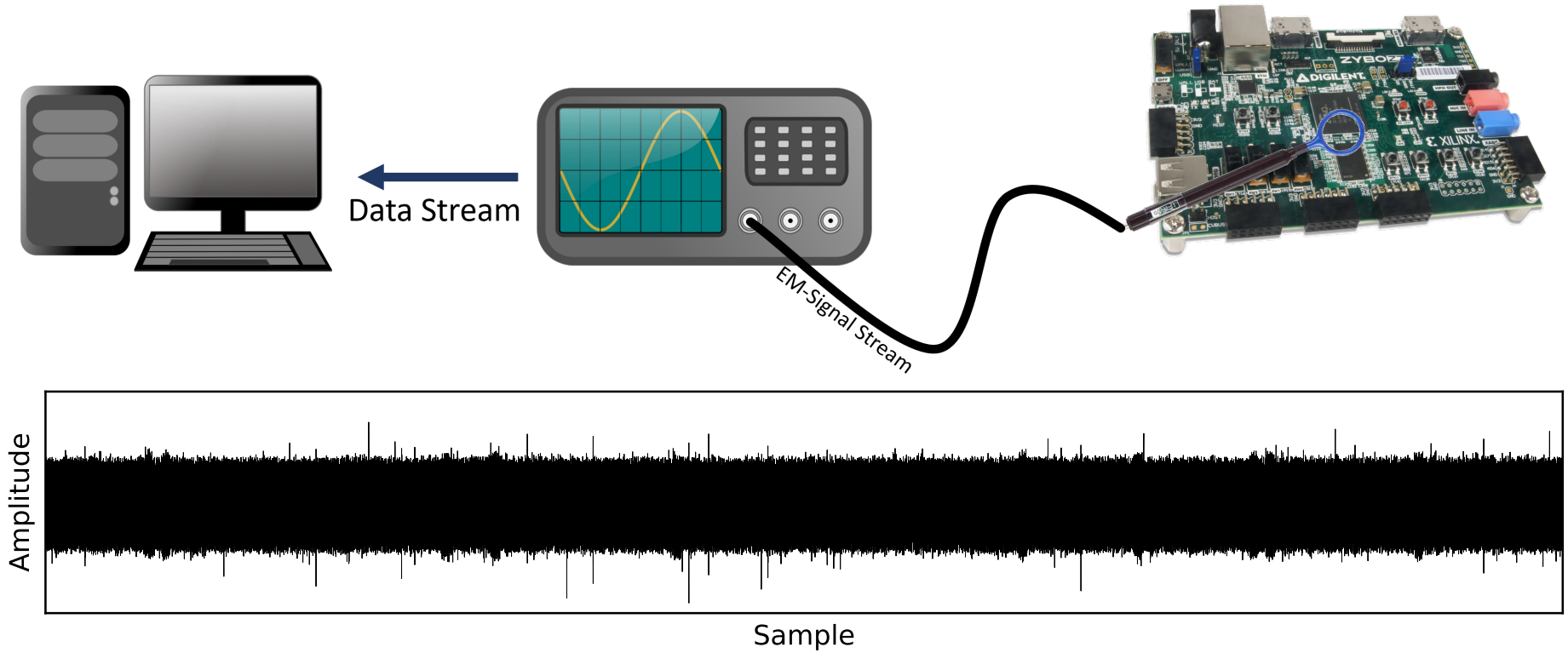
Side-Channel Analysis

- **Side-Channel Attacks**: iPhones [Lis+21], Single Board Computers [Bal+15], IoT devices [Ron+17]
- **Goals**: Retrieve keys, implementation details, ...
- **Requirements**: Side-channel recordings of Cryptographic Operations (COs), associated plaintext/ciphertext



Motivation

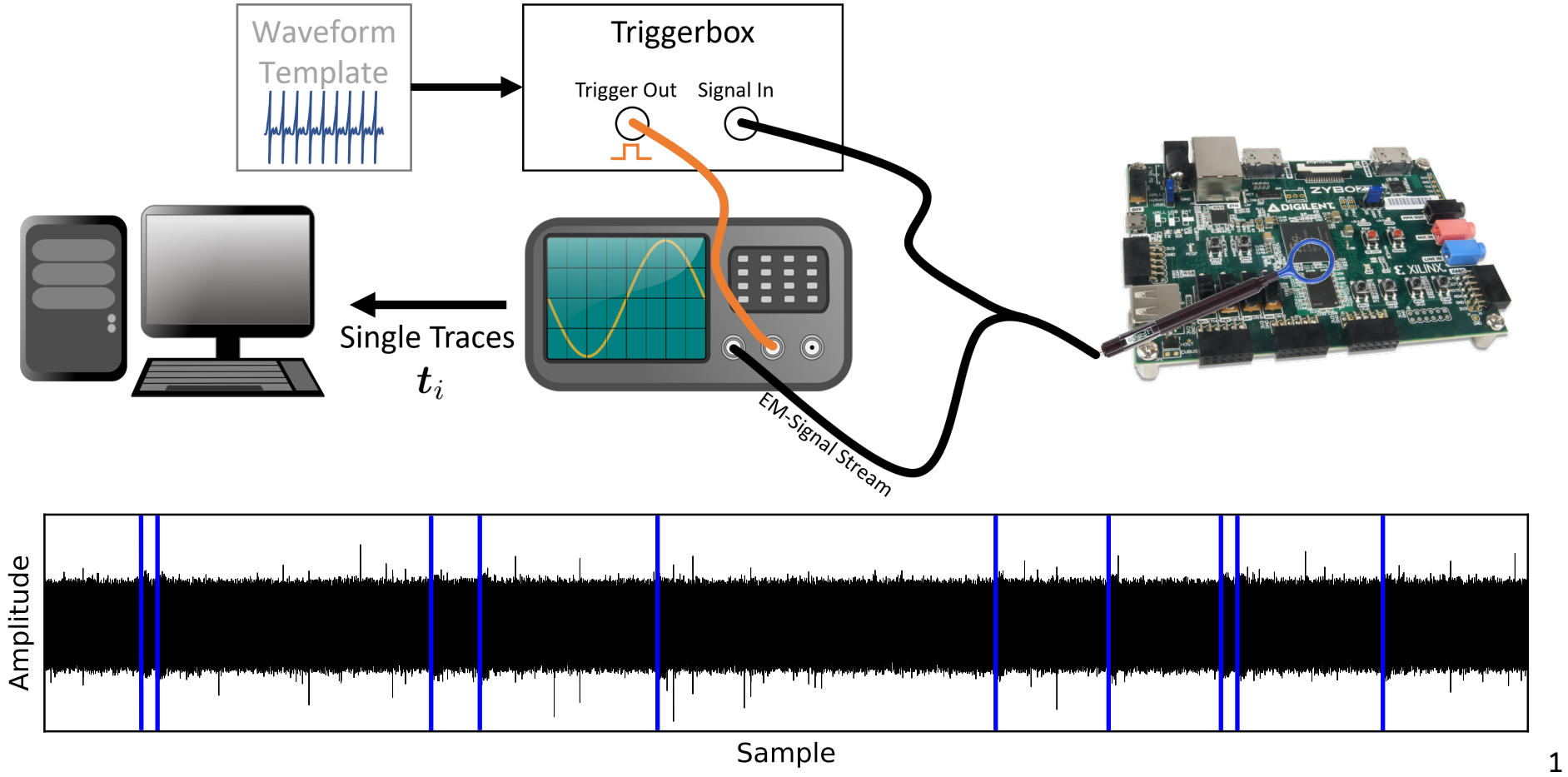
Waveform-Matching Triggering Systems



[1] A. Beckers, J. Balasch, B. Gierlichs, and I. Verbauwhede. "Design and Implementation of a Waveform-Matching Based Triggering System". In: *Constructive Side-Channel Analysis and Secure Design*. Ed. by F.-X. Standaert and E. Oswald. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 184–198. DOI: 10.1007/978-3-319-43283-0_11

Motivation

Waveform-Matching Triggering Systems



[1]

[1] A. Beckers, J. Balasch, B. Gierlichs, and I. Verbauwhede. "Design and Implementation of a Waveform-Matching Based Triggering System". In: *Constructive Side-Channel Analysis and Secure Design*. Ed. by F.-X. Standaert and E. Oswald. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 184–198. DOI: 10.1007/978-3-319-43283-0_11

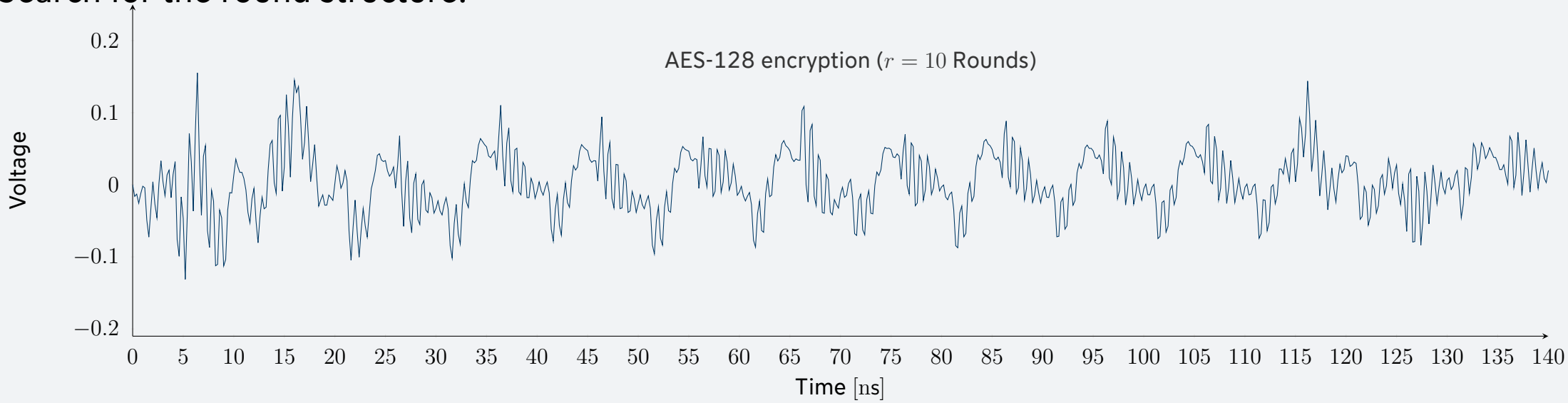
Method

High Level Overview

1. Identify the most probable location of one CO (Cryptographic Operation).

Concept

Search for the round structure:



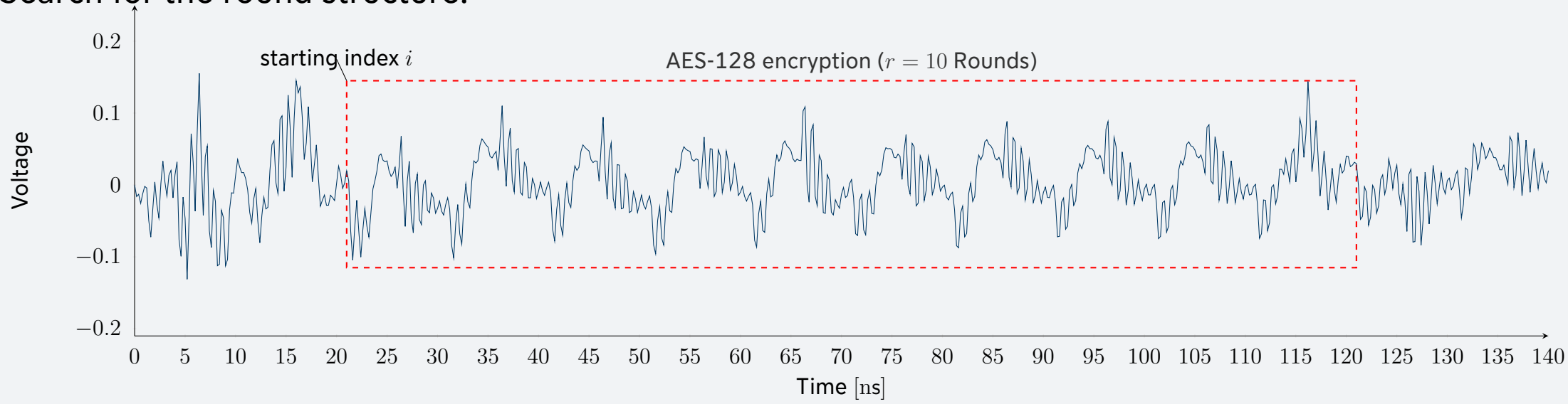
Method

High Level Overview

1. Identify the most probable location of one CO (Cryptographic Operation).

Concept

Search for the round structure:



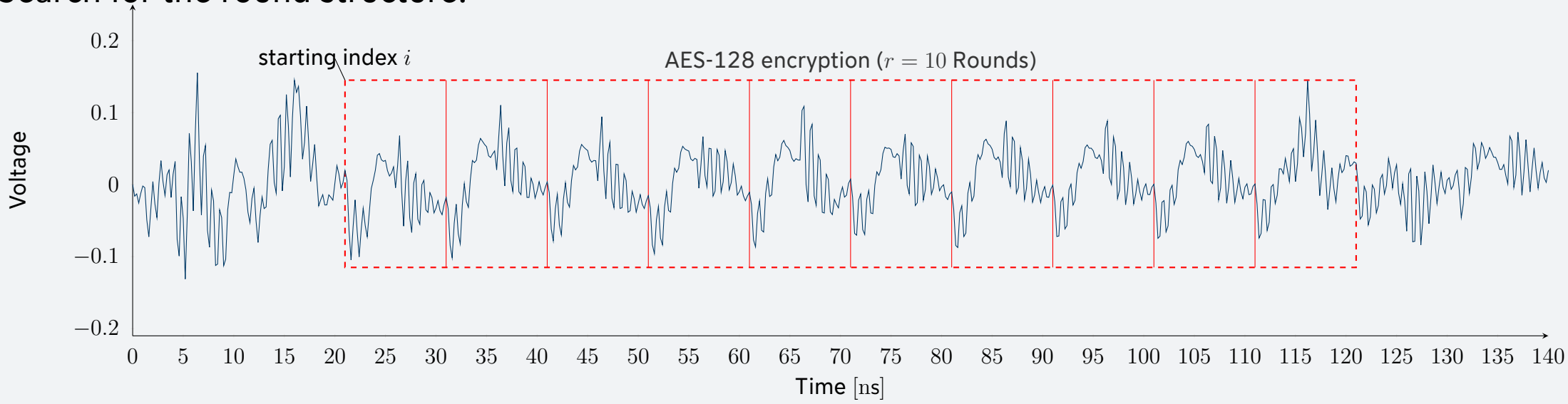
Method

High Level Overview

1. Identify the most probable location of one CO (Cryptographic Operation).

Concept

Search for the round structure:



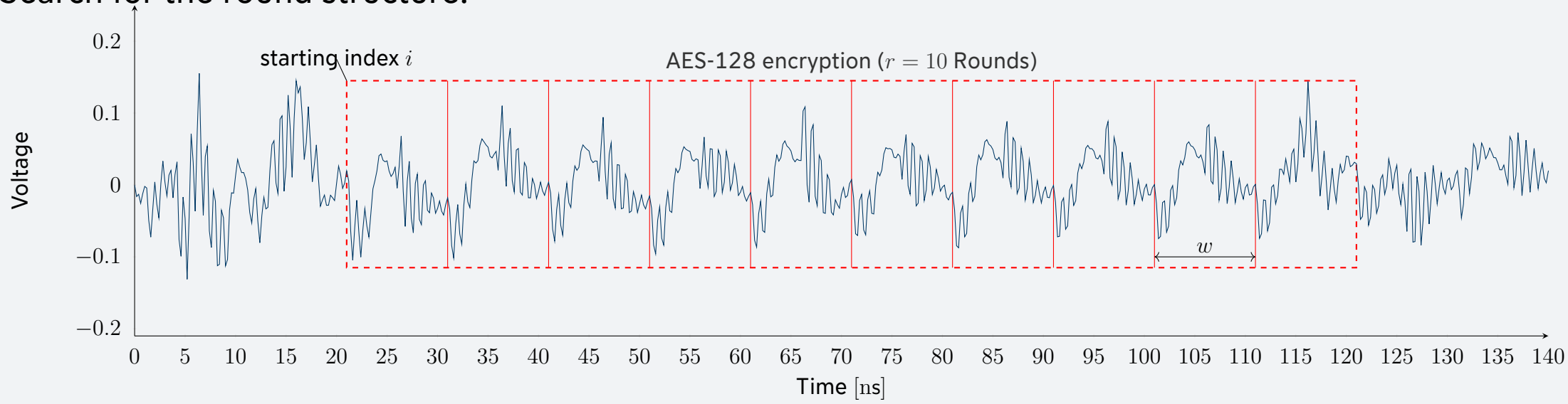
Method

High Level Overview

1. Identify the most probable location of one CO (Cryptographic Operation).

Concept

Search for the round structure:



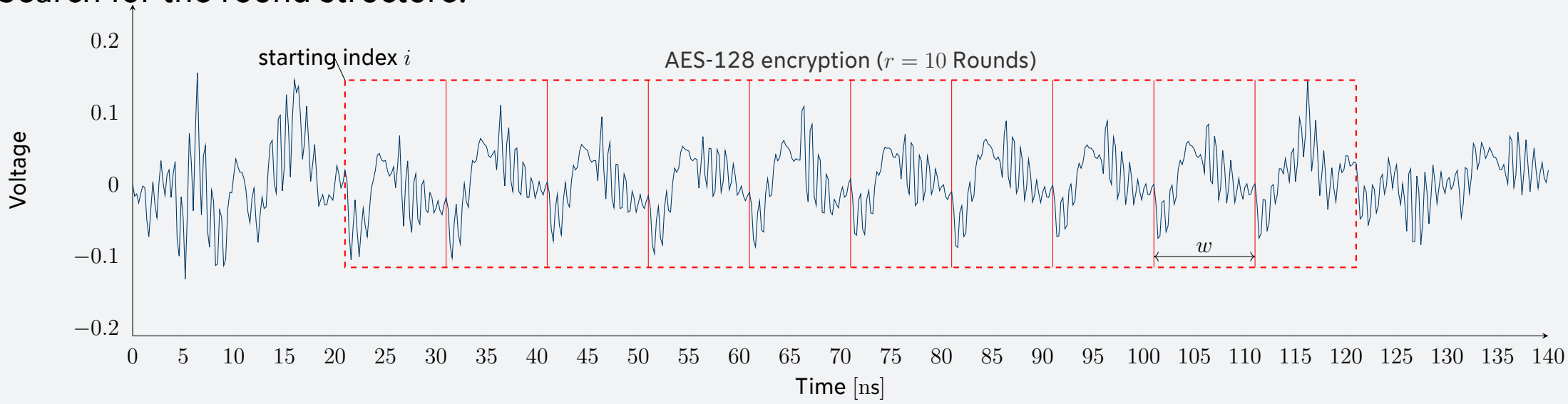
Method

High Level Overview

1. Identify the most probable location of one CO (Cryptographic Operation).
2. Generate a CO template.
3. Use the CO template to find all trace segments which correspond to other occurrences of the same CO.

Concept

Search for the round structure:



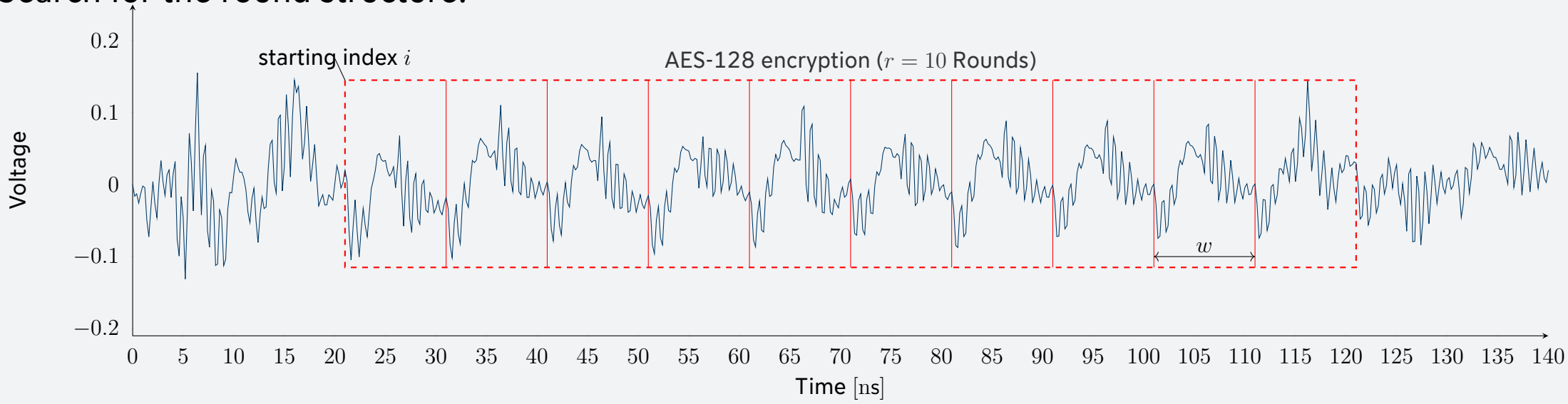
Method

High Level Overview

1. Identify the most probable location of one CO (Cryptographic Operation).
2. Generate a CO template.
3. Use the CO template to find all trace segments which correspond to other occurrences of the same CO.
4. Use those trace segments to refine the CO template (optional).

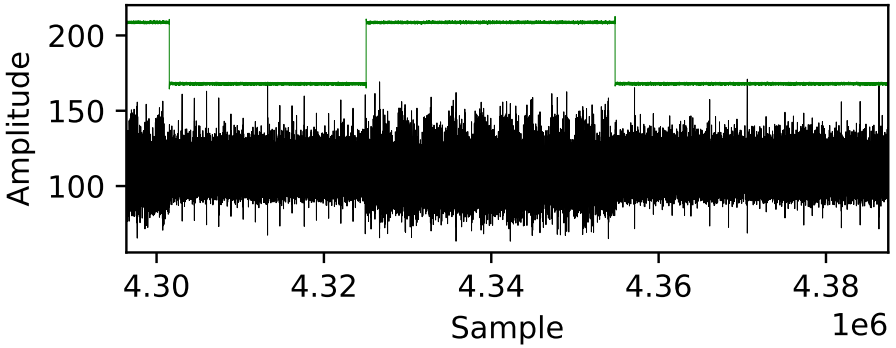
Concept

Search for the round structure:



Method

Locating COs if the round pattern length w is known



How to locate COs in a trace t with known w and r

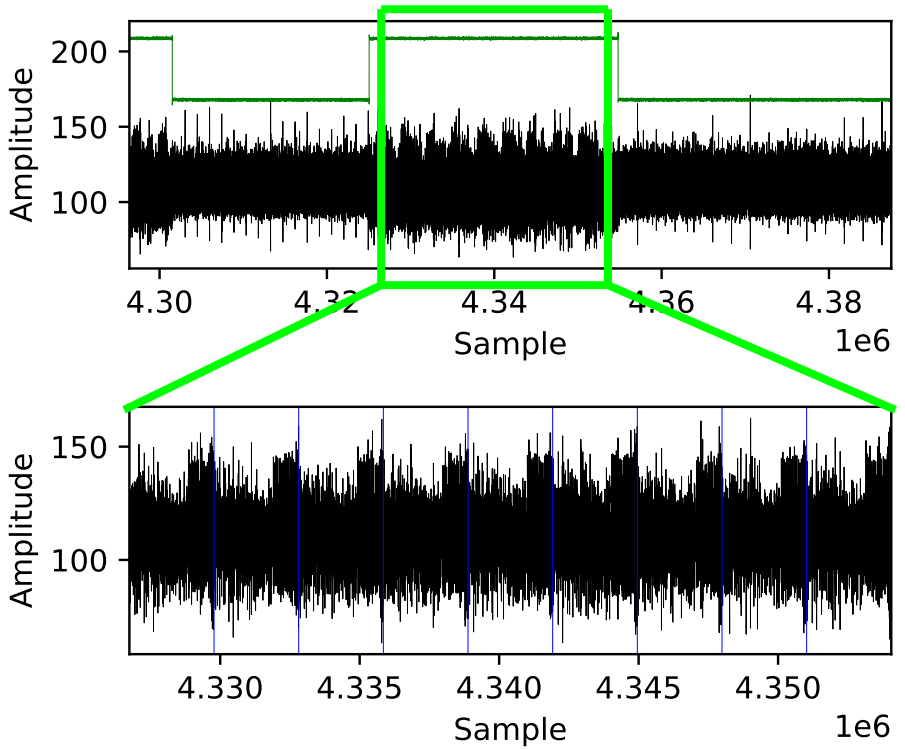
- Go through each index in trace t

Method

Locating COs if the round pattern length w is known

How to locate COs in a trace t with known w and r

- Go through each index in trace t
  - Calculate similarity between potential rounds

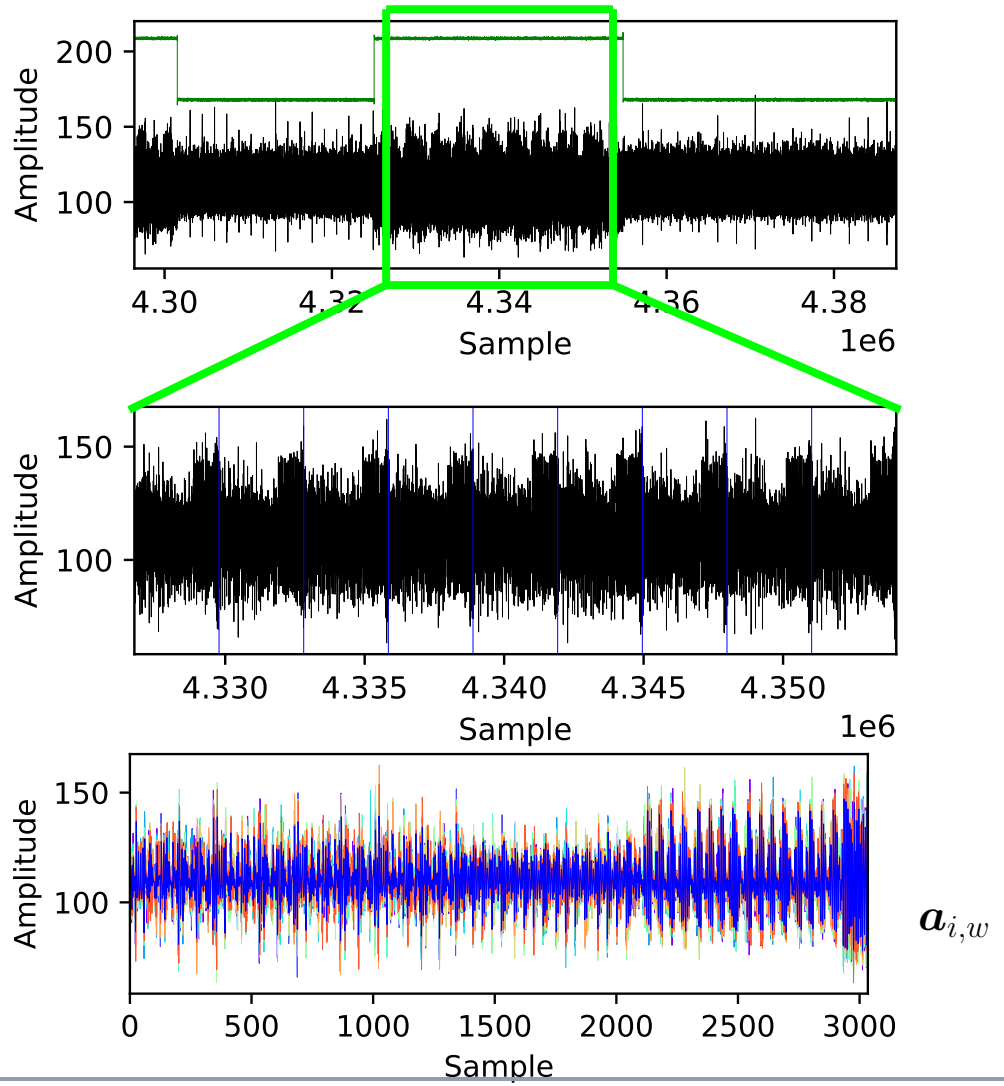


Method

Locating COs if the round pattern length w is known

How to locate COs in a trace t with known w and r

- Go through each index in trace t
  - Calculate similarity between potential rounds

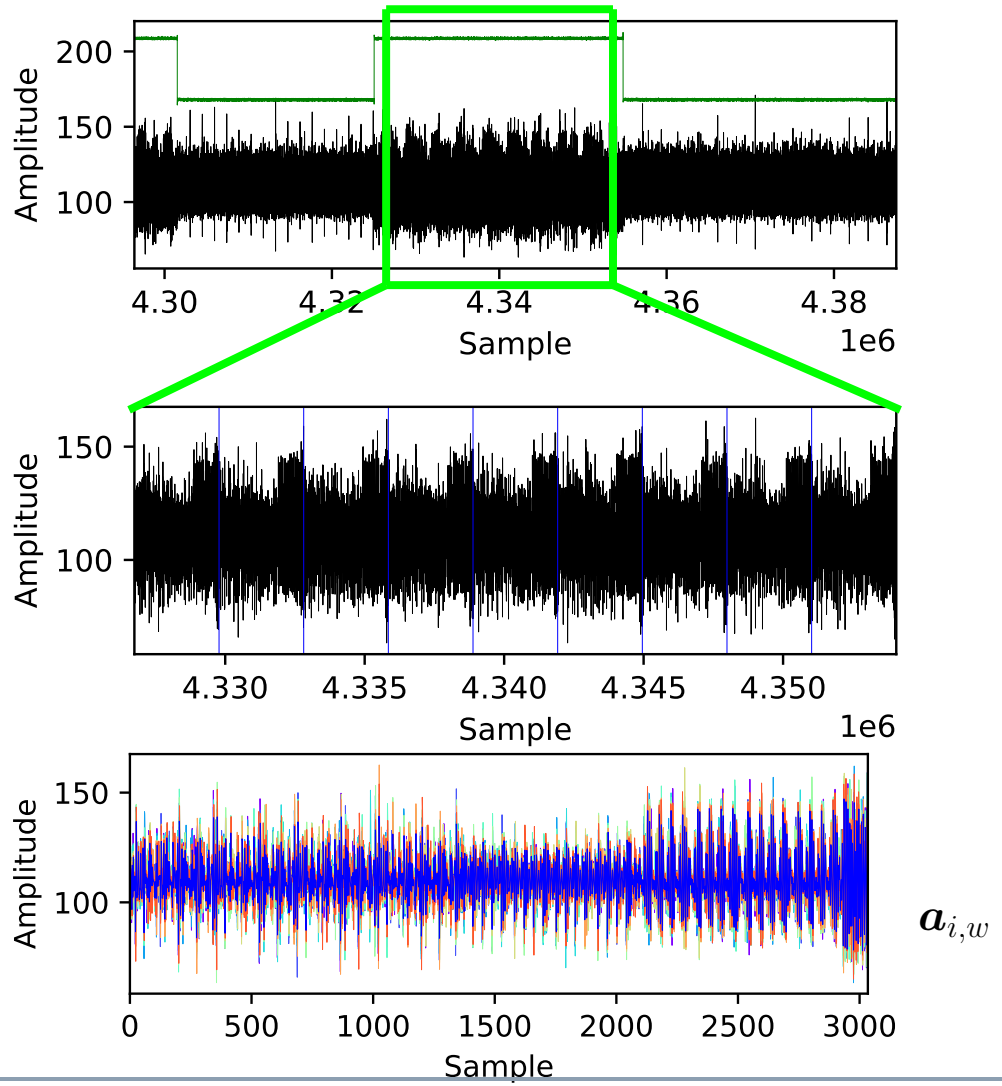


Method

Locating COs if the round pattern length w is known

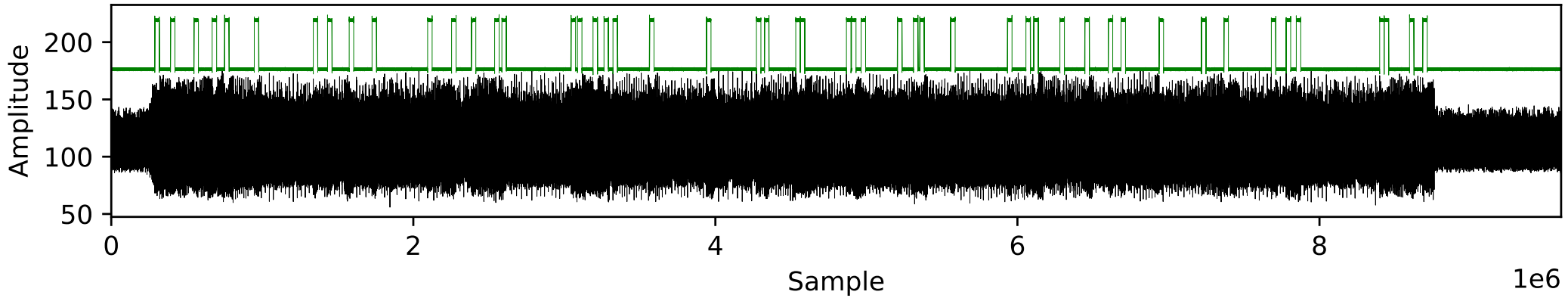
How to locate COs in a trace t with known w and r

- Go through each index in trace t
  - Calculate similarity between potential rounds
- Select point with the highest similarity as most probable starting point of a CO
- Create CO template and find all occurrences of this CO



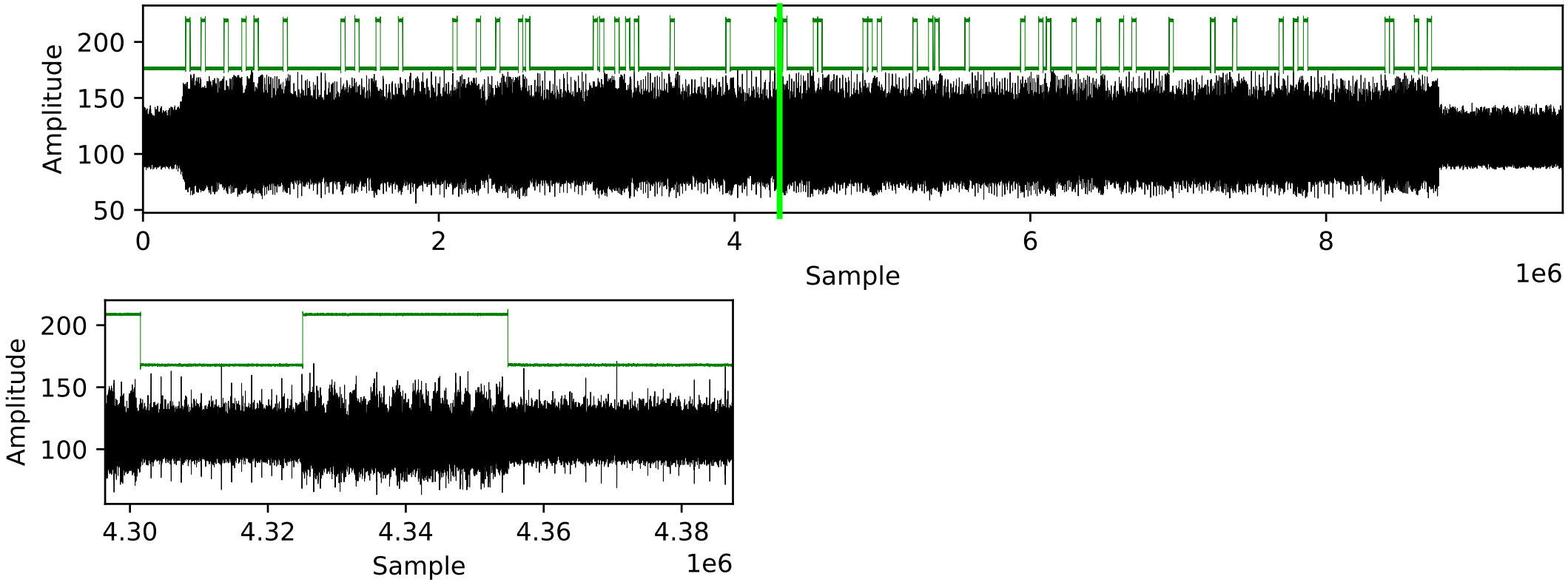
Method

Locating COs if the round pattern length w is known



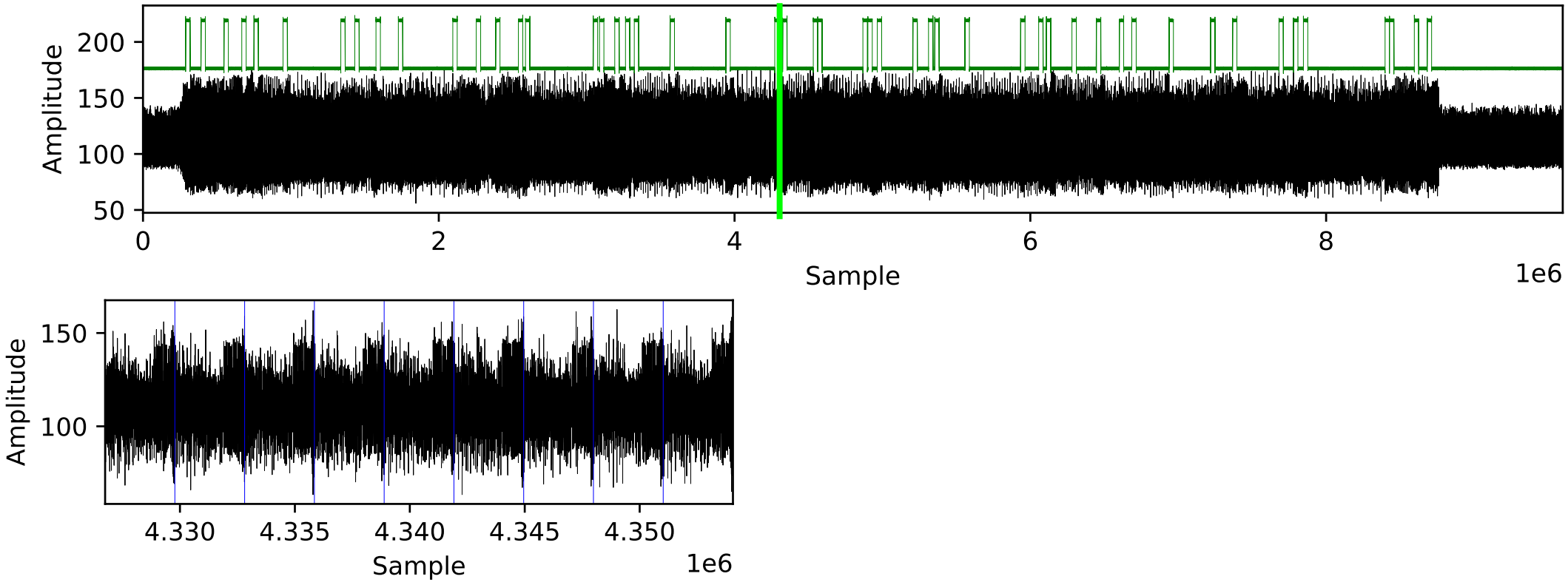
Method

Locating COs if the round pattern length w is known



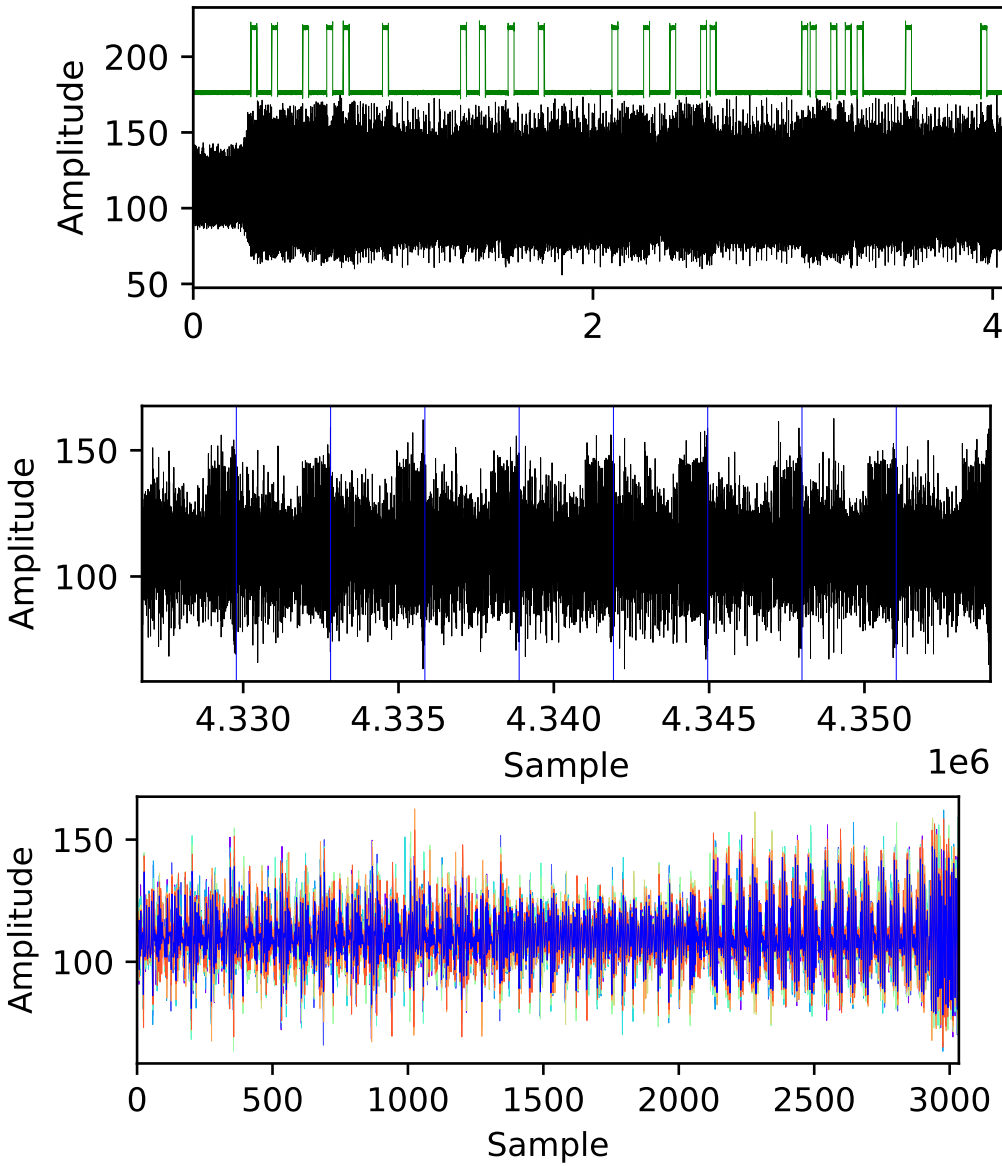
Method

Locating COs if the round pattern length w is known



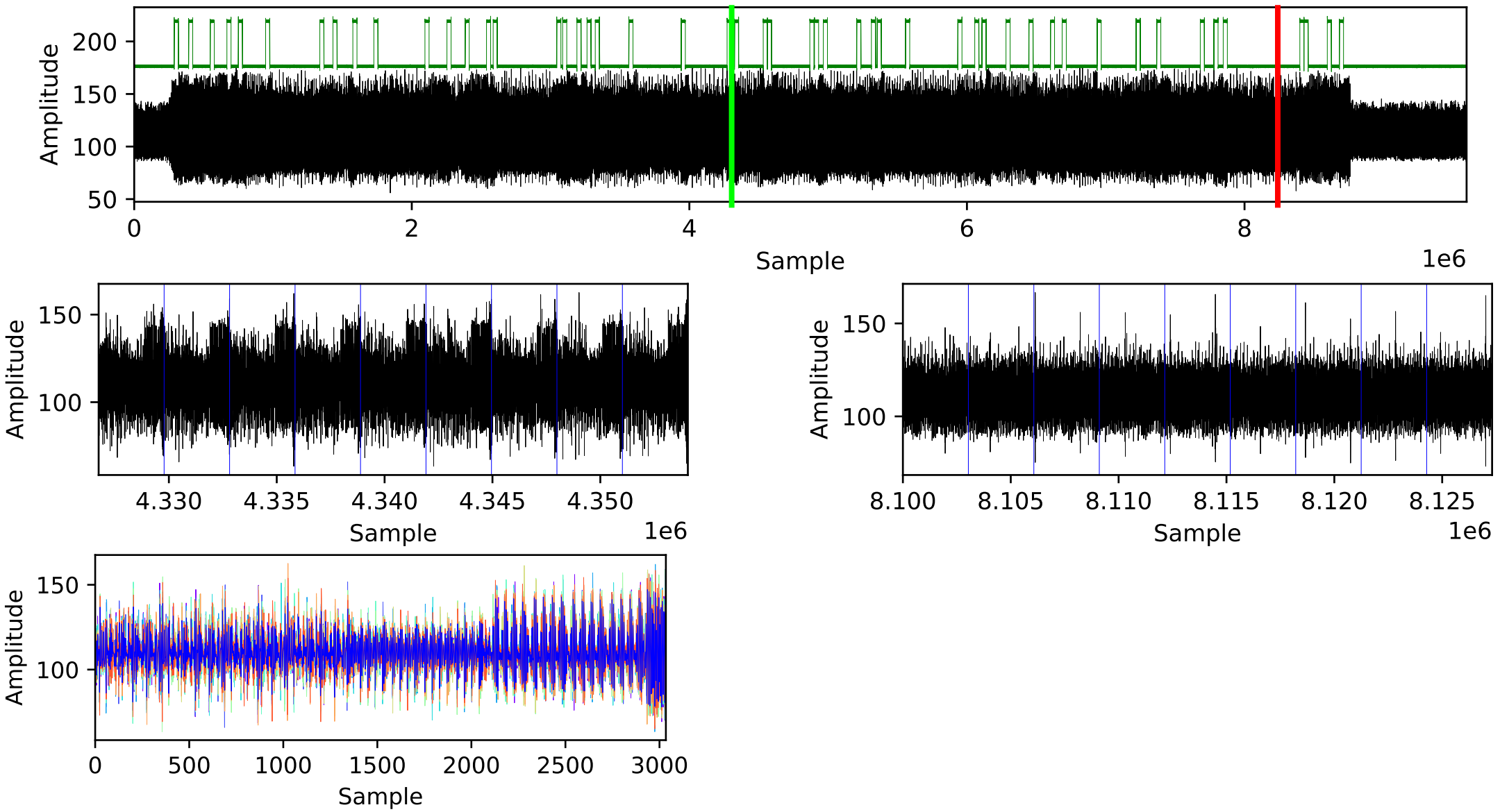
Method

Locating COs if the round pattern length w is known



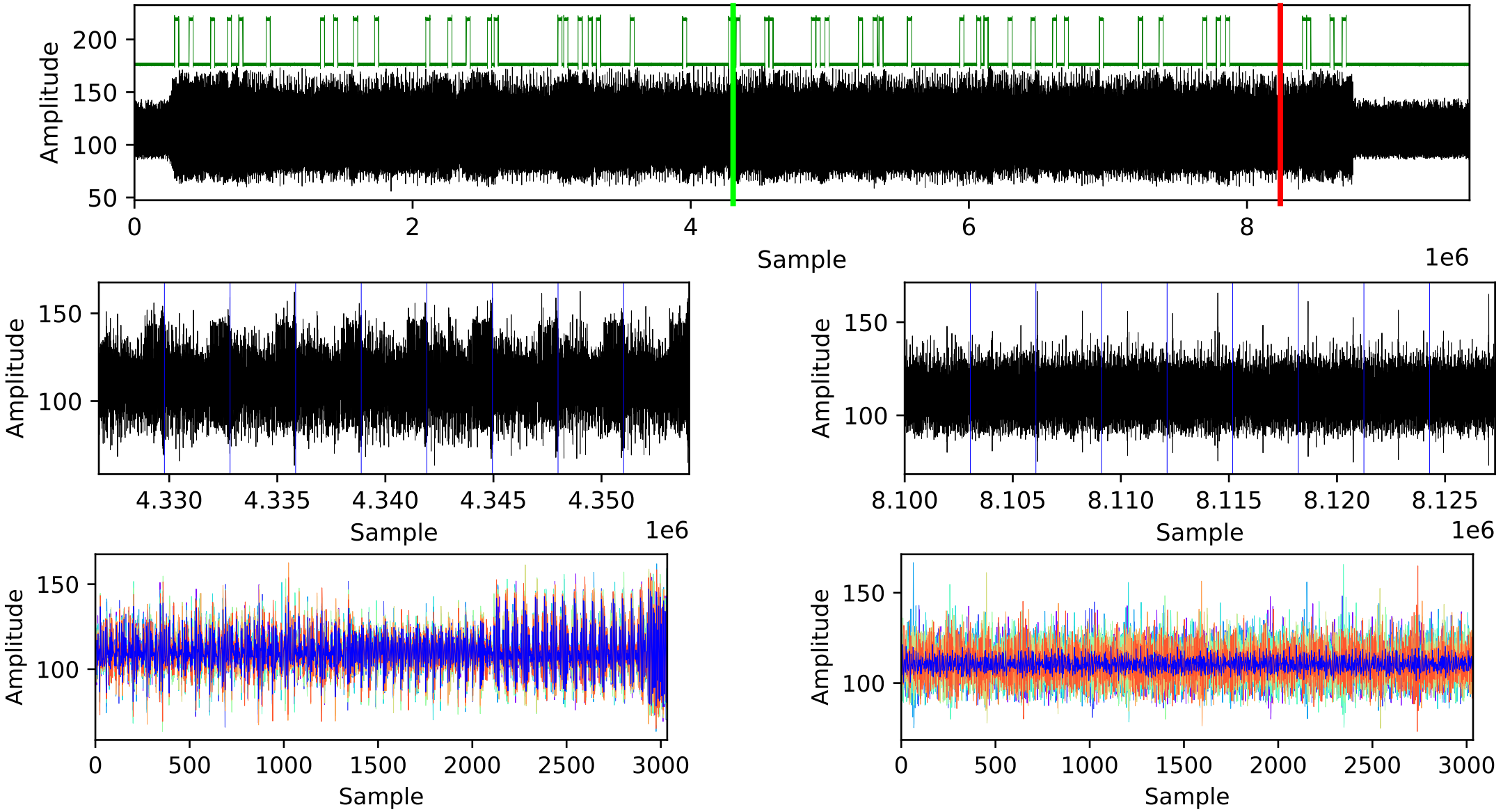
Method

Locating COs if the round pattern length w is known



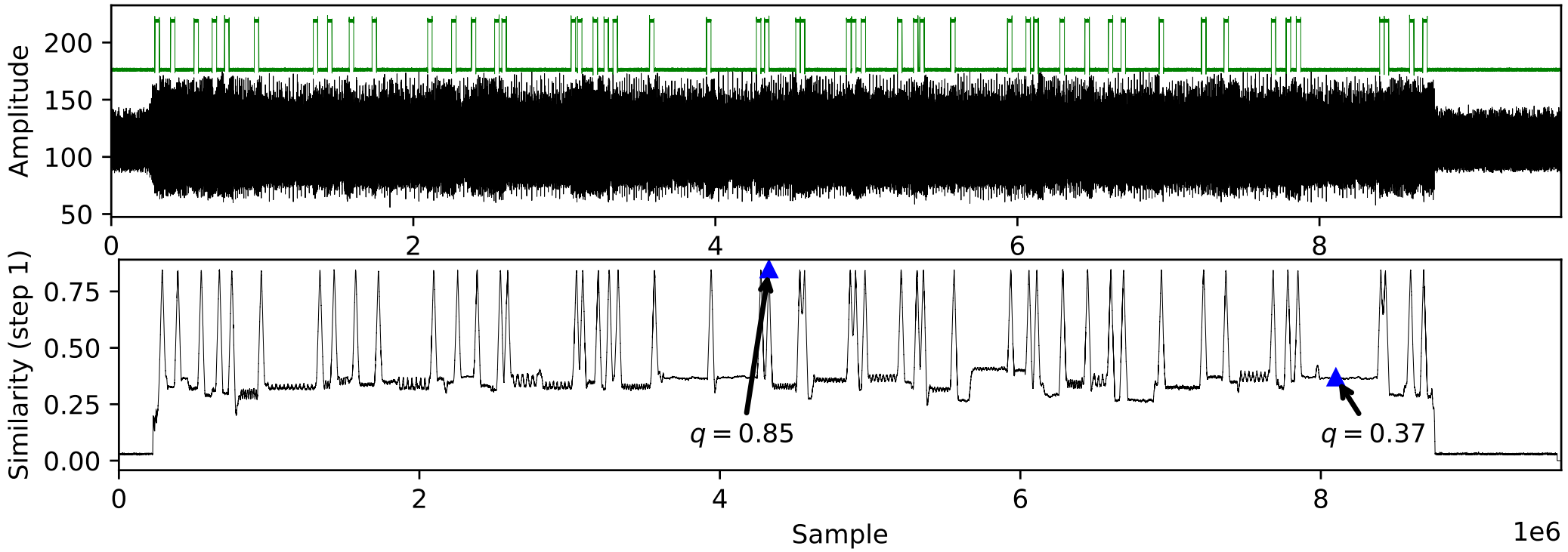
Method

Locating COs if the round pattern length w is known



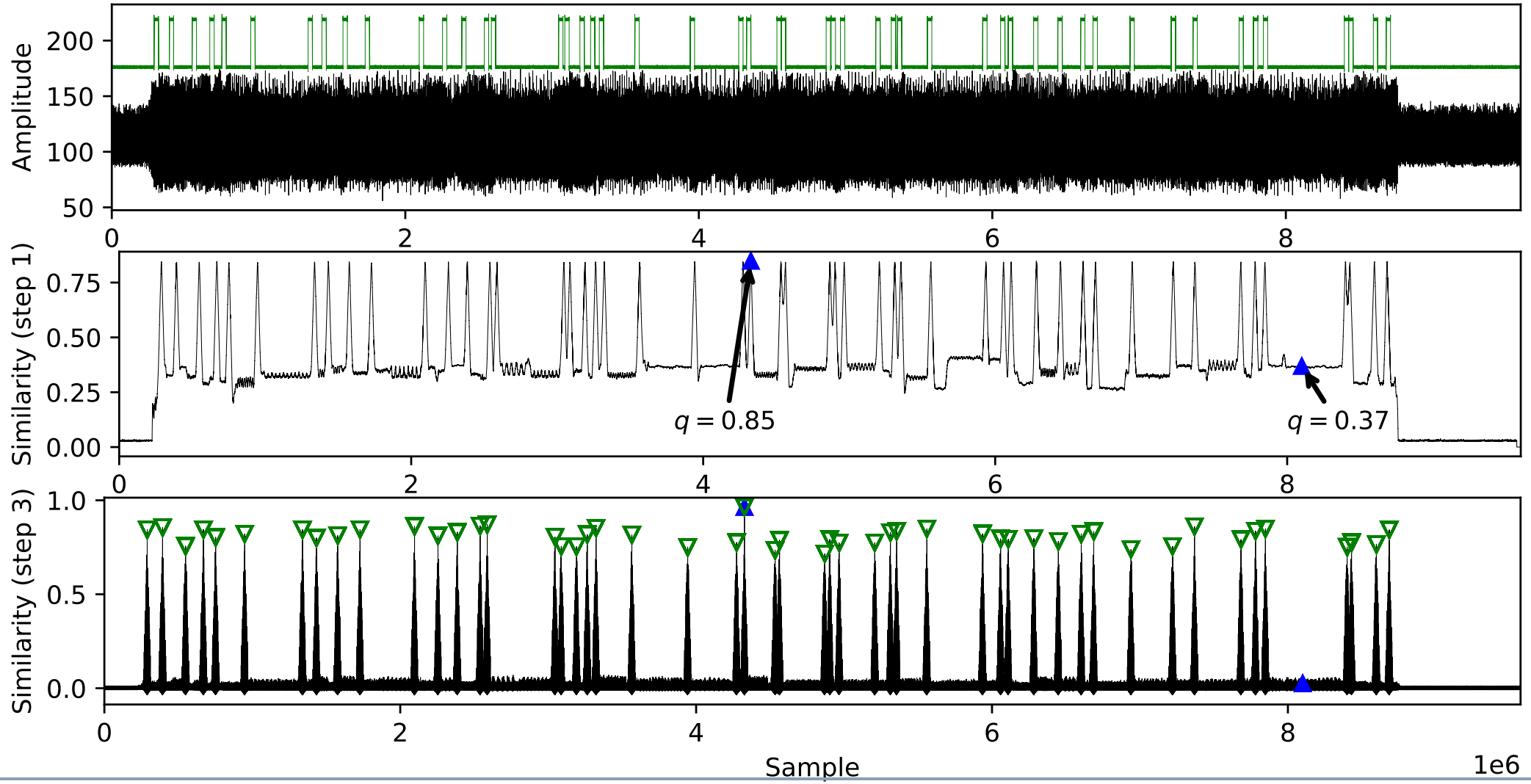
Method

Locating COs if the round pattern length w is known



Method

Locating COs if the round pattern length w is known



Method

Locating COs if the round pattern length w is **un**known

How to locate COs in a trace t with known r and unknown w

- For all reasonable widths w
  - Determine CO template with width w as shown before

Requirements:

Method

Locating COs if the round pattern length w is **un**known

How to locate COs in a trace t with known r and unknown w

- For all reasonable widths w
  - Determine CO template with width w as shown before
  - Calculate similarity between template and trace

Requirements:

- Side-channel trace $\boldsymbol{t}$ recorded with sampling rate f_{sample}
- Knowledge of the device clock frequency f_{device} (calculation of possible widths $\boldsymbol{w}$)
- Knowledge of the number r of *successive* and *identical* rounds that are part of the CO

Method

Locating COs if the round pattern length w is **un**known

How to locate COs in a trace t with known r and unknown w

- For all reasonable widths w
  - Determine CO template with width w as shown before
  - Calculate similarity between template and trace
  - Determine possible CO starting points based on shape of similarity

Requirements:

- Side-channel trace $\boldsymbol{t}$ recorded with sampling rate f_{sample}
- Knowledge of the device clock frequency f_{device} (calculation of possible widths $\boldsymbol{w}$)
- Knowledge of the number r of *successive* and *identical* rounds that are part of the CO

Method

Locating COs if the round pattern length w is **un**known

How to locate COs in a trace t with known r and unknown w

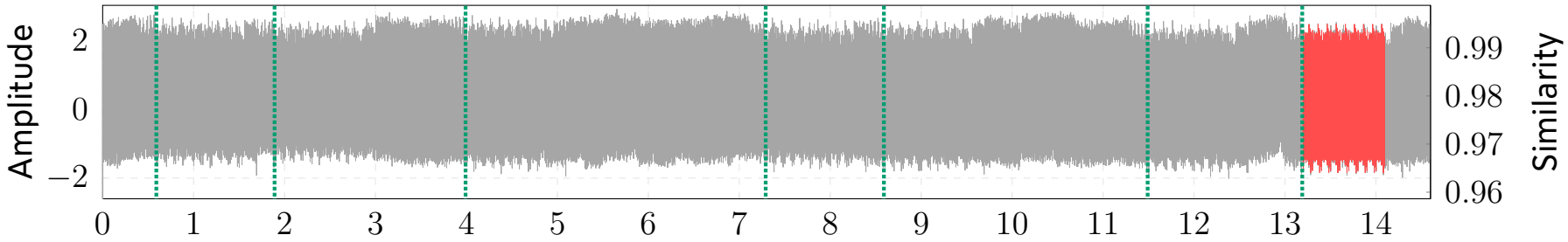
- For all reasonable widths w
  - Determine CO template with width w as shown before
  - Calculate similarity between template and trace
  - Determine possible CO starting points based on shape of similarity
  - Success if number of found COs = n

Requirements:

- Side-channel trace $\boldsymbol{t}$ recorded with sampling rate f_{sample}
- Knowledge of the device clock frequency f_{device} (calculation of possible widths $\boldsymbol{w}$)
- Knowledge of the number r of *successive* and *identical* rounds that are part of the CO
- Trace $\boldsymbol{t}$ includes a *known* number $n \geq 1$ of COs

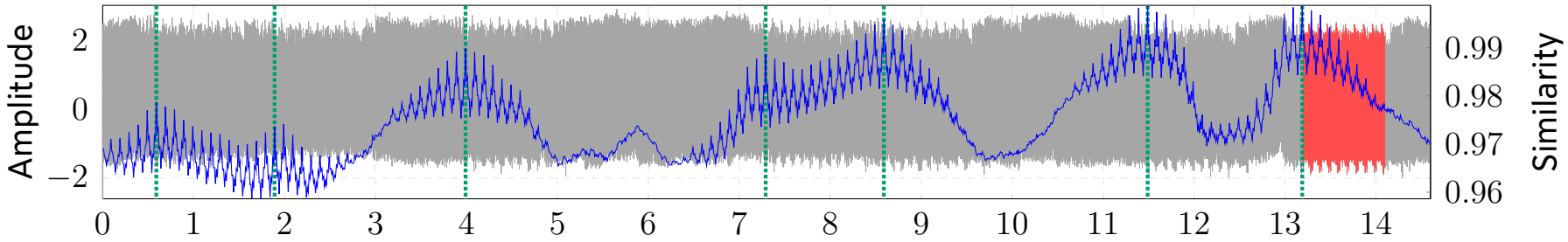
Method

Locating COs if the round pattern length w is **un**known



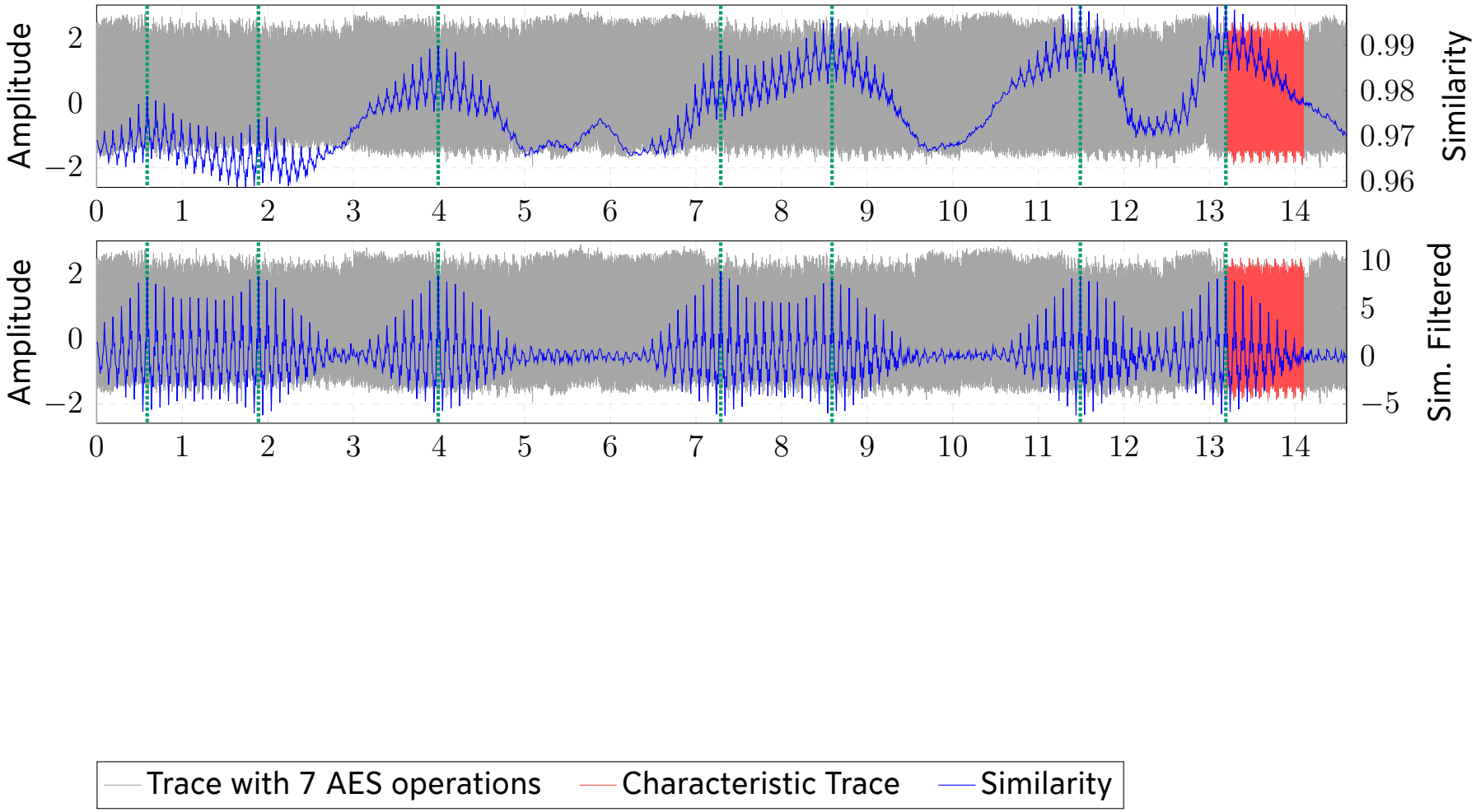
Method

Locating COs if the round pattern length w is **un**known



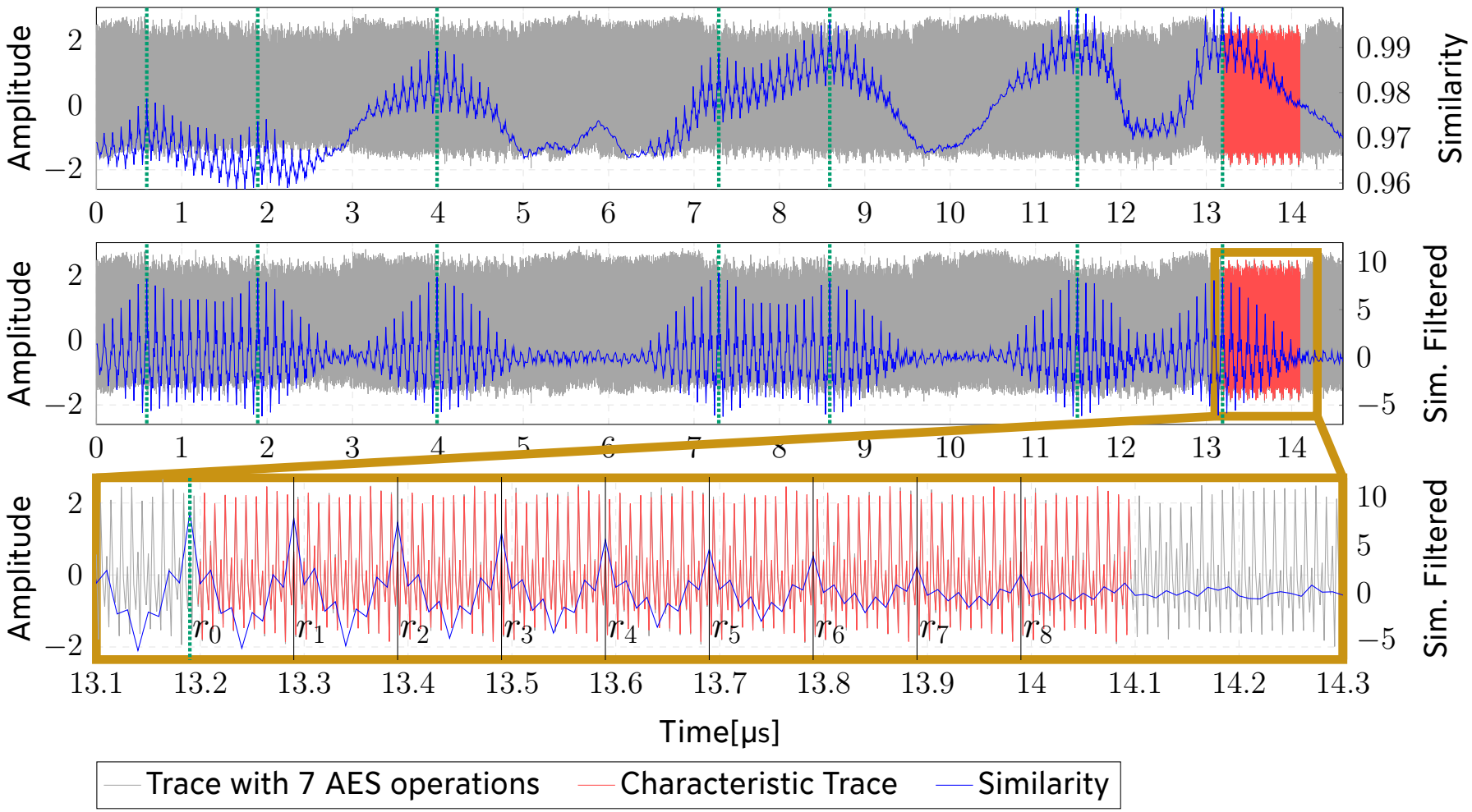
Method

Locating COs if the round pattern length w is **un**known



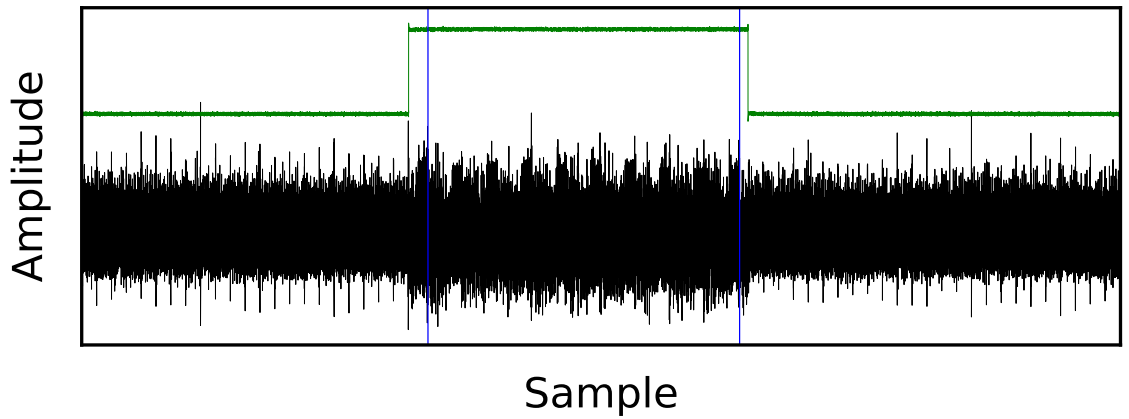
Method

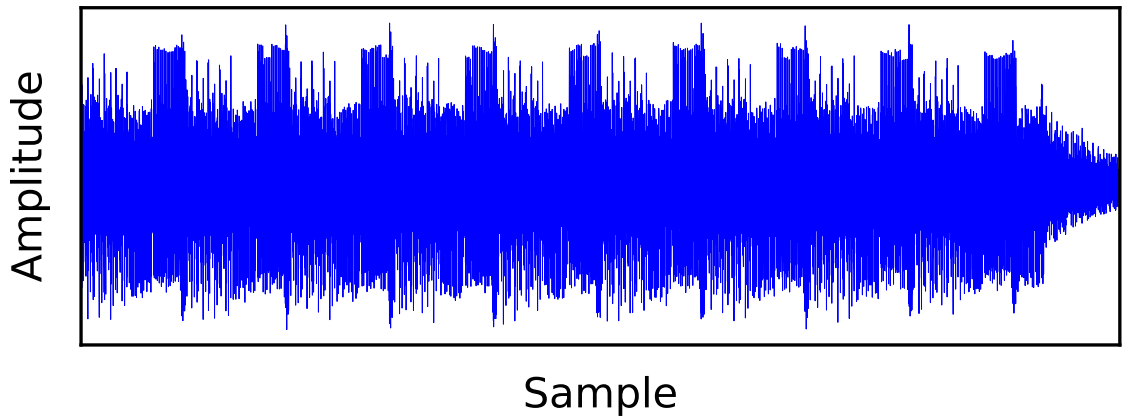
Locating COs if the round pattern length w is **un**known



Evaluation

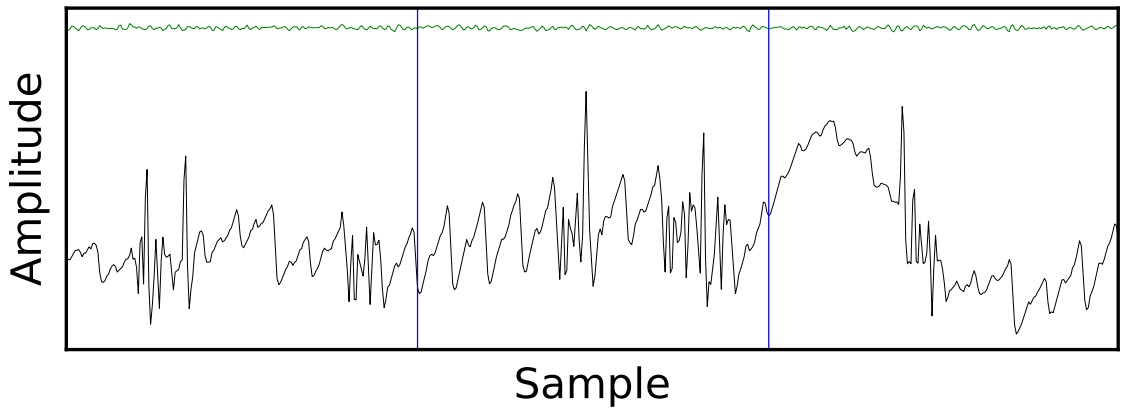
- Proof of Concept with 100% accuracy for different Systems:
  - STM32F415@8MHz (ARM Cortex-M4) with TinyAES

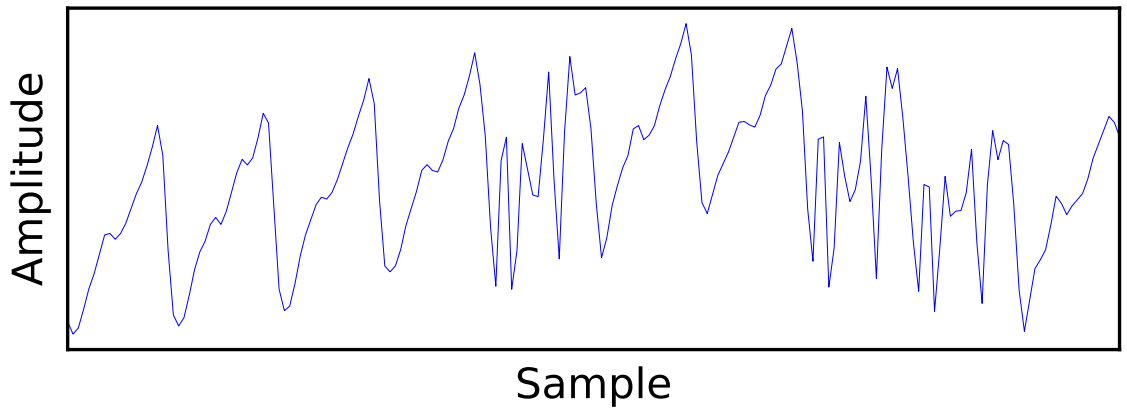




Evaluation

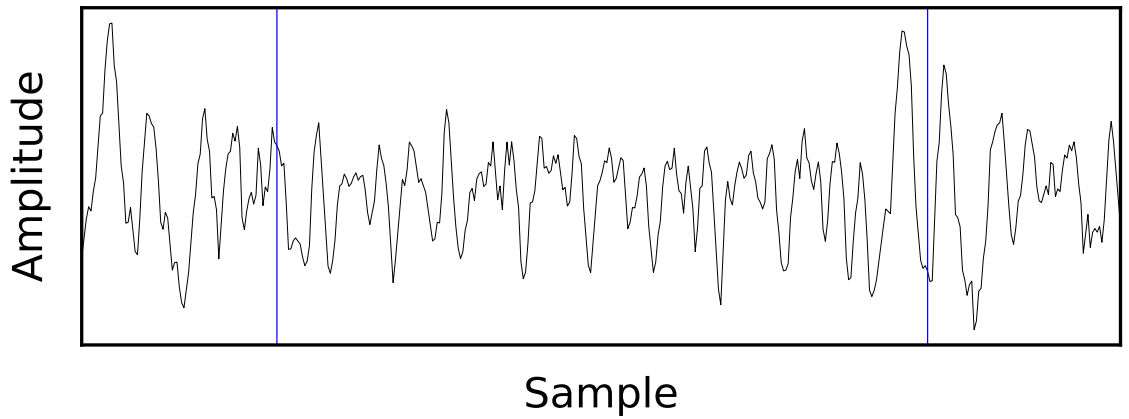
- Proof of Concept with 100% accuracy for different Systems:
  - STM32F415@8MHz (ARM Cortex-M4) with TinyAES
  - STM32F415@8MHz (ARM Cortex-M4) with HWAES

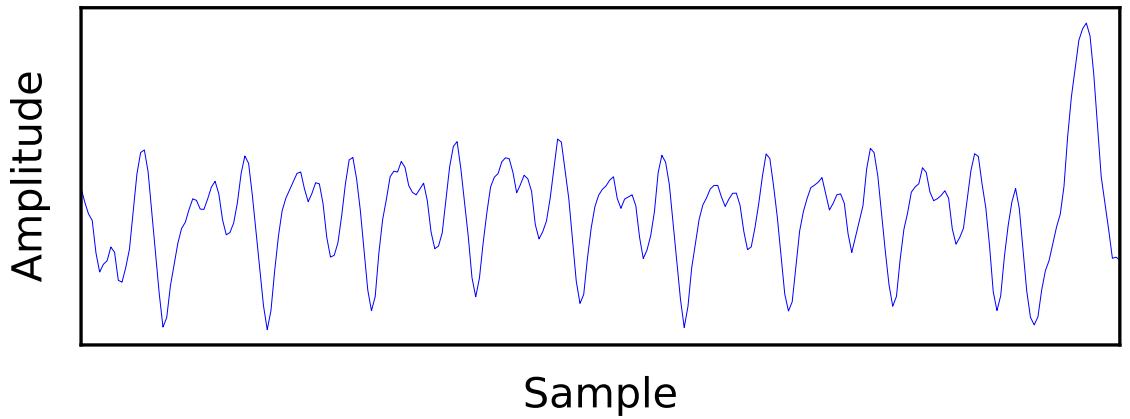




Evaluation

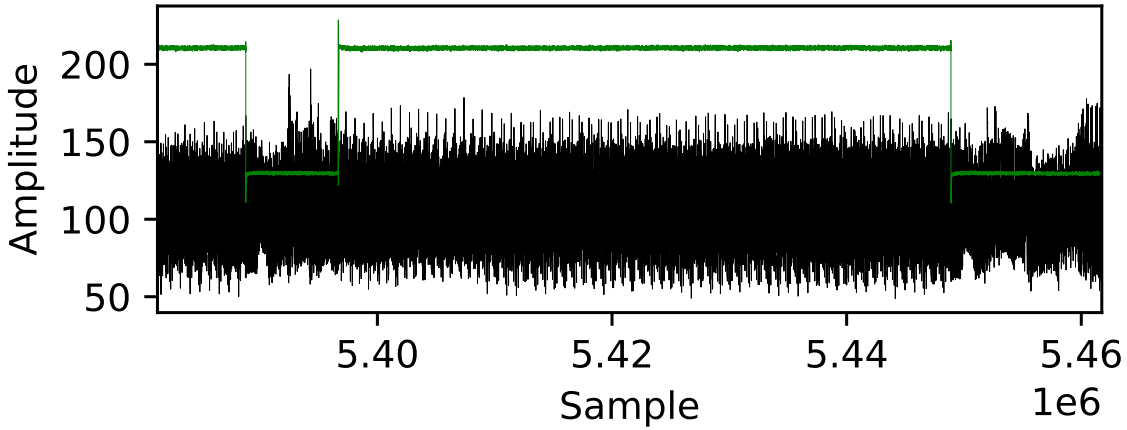
- Proof of Concept with 100% accuracy for different Systems:
  - STM32F415@8MHz (ARM Cortex-M4) with TinyAES
  - STM32F415@8MHz (ARM Cortex-M4) with HWAES
  - AM335x@1GHz (ARM Cortex-A8) on the BeagleBone Black with OpenSSL AES-128

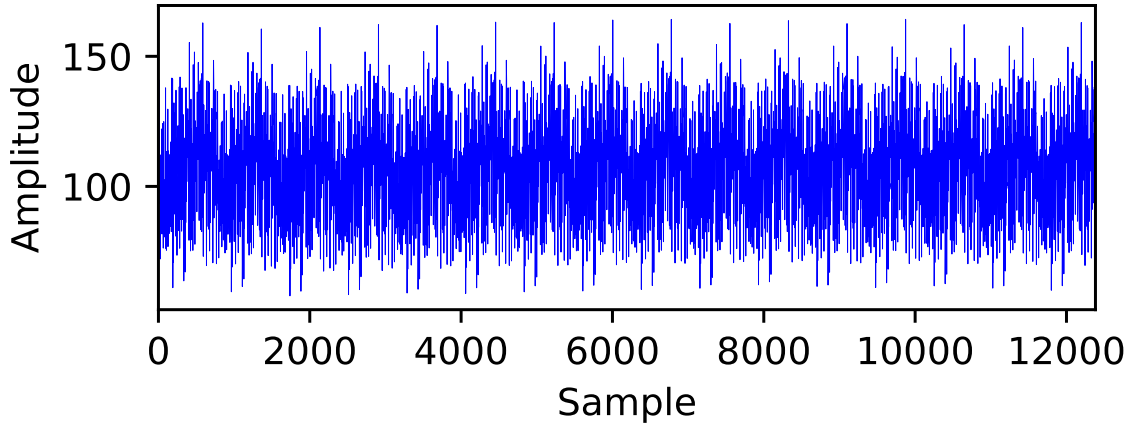




Evaluation

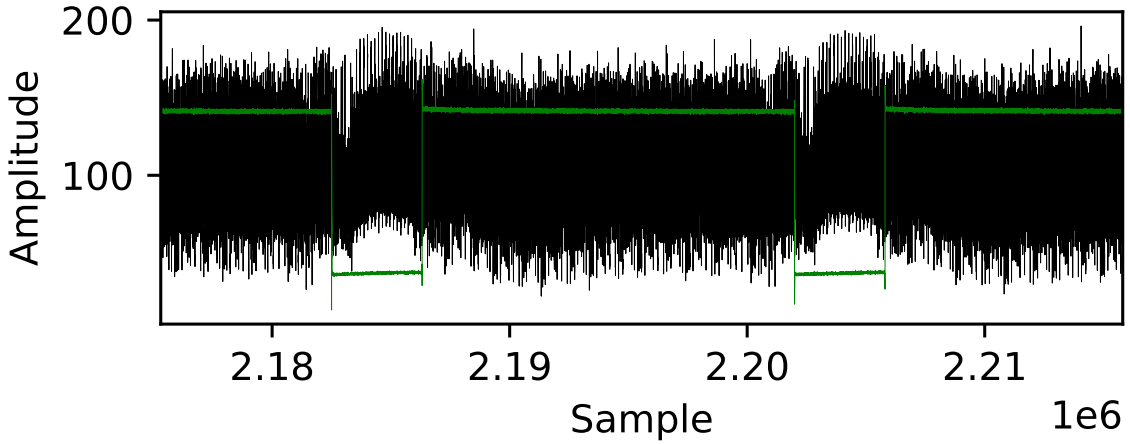
- Proof of Concept with 100% accuracy for different Systems:
  - STM32F415@8MHz (ARM Cortex-M4) with TinyAES
  - STM32F415@8MHz (ARM Cortex-M4) with HWAES
  - AM335x@1GHz (ARM Cortex-A8) on the BeagleBone Black with OpenSSL AES-128
  - STM32F303@72MHz (ARM Cortex-M4) with Mbed TLS SHA-256 (Internal and external clock)

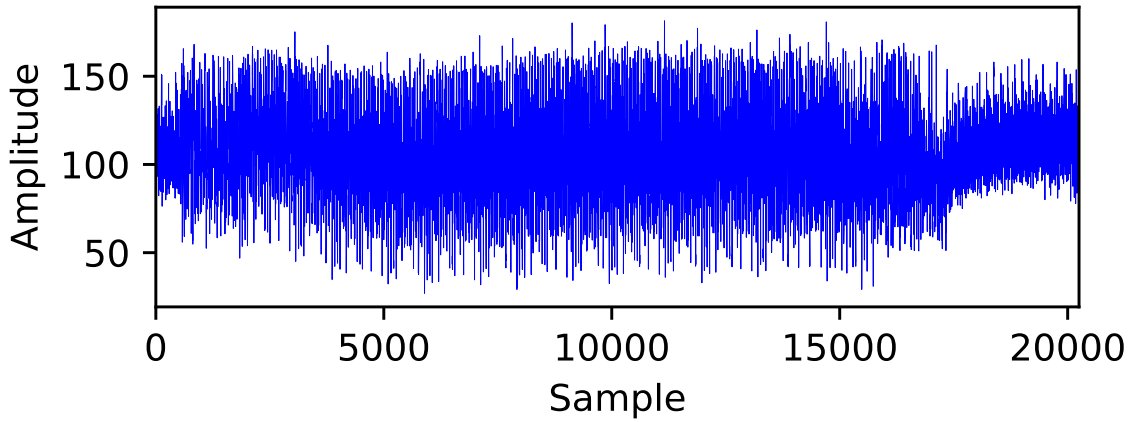




Evaluation

- Proof of Concept with 100% accuracy for different Systems:
  - STM32F415@8MHz (ARM Cortex-M4) with TinyAES
  - STM32F415@8MHz (ARM Cortex-M4) with HWAES
  - AM335x@1GHz (ARM Cortex-A8) on the BeagleBone Black with OpenSSL AES-128
  - STM32F303@72MHz (ARM Cortex-M4) with Mbed TLS SHA-256 (Internal and external clock)
  - STM32F303@72MHz (ARM Cortex-M4) with Mbed TLS AES-128 (Internal and external clock)





Evaluation

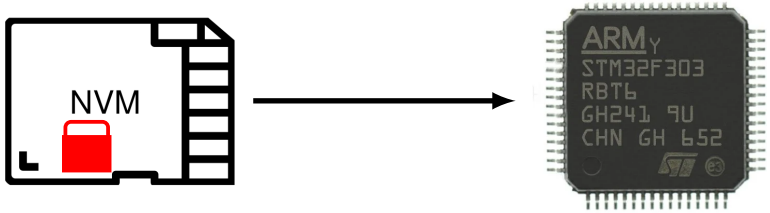
AES-128 in a Secure Boot Scenario

- Secure boot on a ARM Cortex-M4 microcontroller (STM32F303@8MHz):

Evaluation

AES-128 in a Secure Boot Scenario

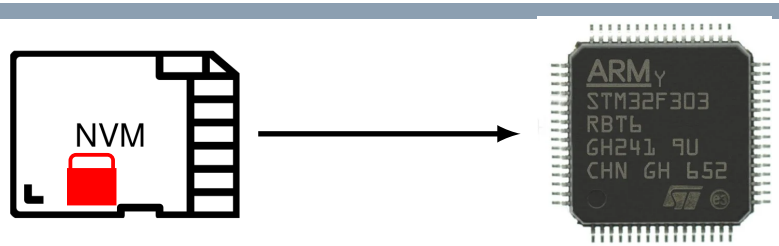
- Secure boot on a ARM Cortex-M4 microcontroller (STM32F303@8MHz):
  - Encrypted boot content (8 kB) on Non-Volatile Memory (NVM)
  - Hashing, ECDSA signature check, 500 AES-128 CBC decryptions



Evaluation

AES-128 in a Secure Boot Scenario

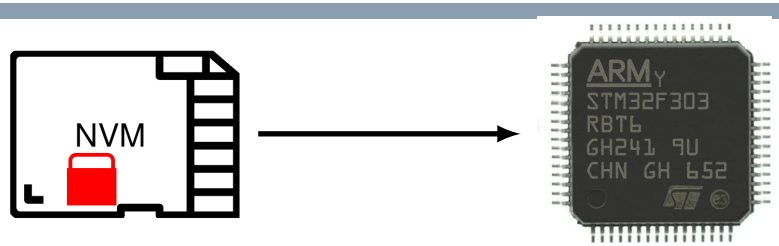
- Secure boot on a ARM Cortex-M4 microcontroller (STM32F303@8MHz):
  - Encrypted boot content (8 kB) on Non-Volatile Memory (NVM)
  - Hashing, ECDSA signature check, 500 AES-128 CBC decryptions
- Attacker capabilities:



Evaluation

AES-128 in a Secure Boot Scenario

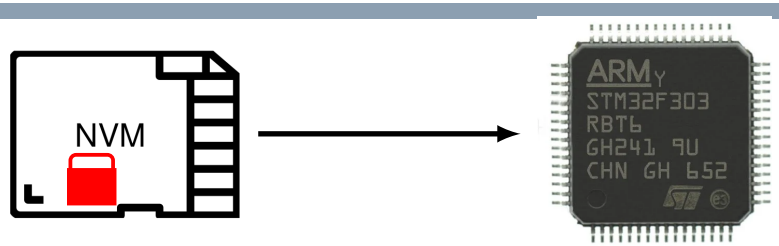
- Secure boot on a ARM Cortex-M4 microcontroller (STM32F303@8MHz):
  - Encrypted boot content (8 kB) on Non-Volatile Memory (NVM)
  - Hashing, ECDSA signature check, 500 AES-128 CBC decryptions
- Attacker capabilities:
  - Read out encrypted boot image from NVM



Evaluation

AES-128 in a Secure Boot Scenario

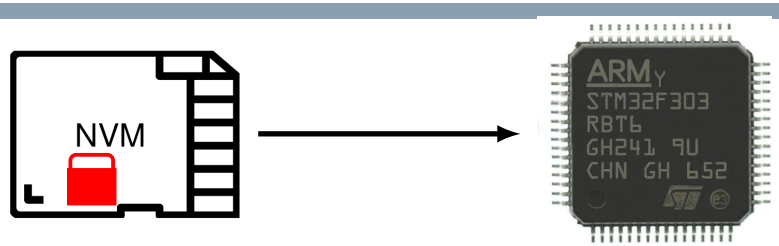
- Secure boot on a ARM Cortex-M4 microcontroller (STM32F303@8MHz):
  - Encrypted boot content (8 kB) on Non-Volatile Memory (NVM)
  - Hashing, ECDSA signature check, 500 AES-128 CBC decryptions
- Attacker capabilities:
  - Read out encrypted boot image from NVM
  - Knowledge of the different boot stages and specifics about the AES decryption:
    - ► AES mode and implementation: AES-CBC

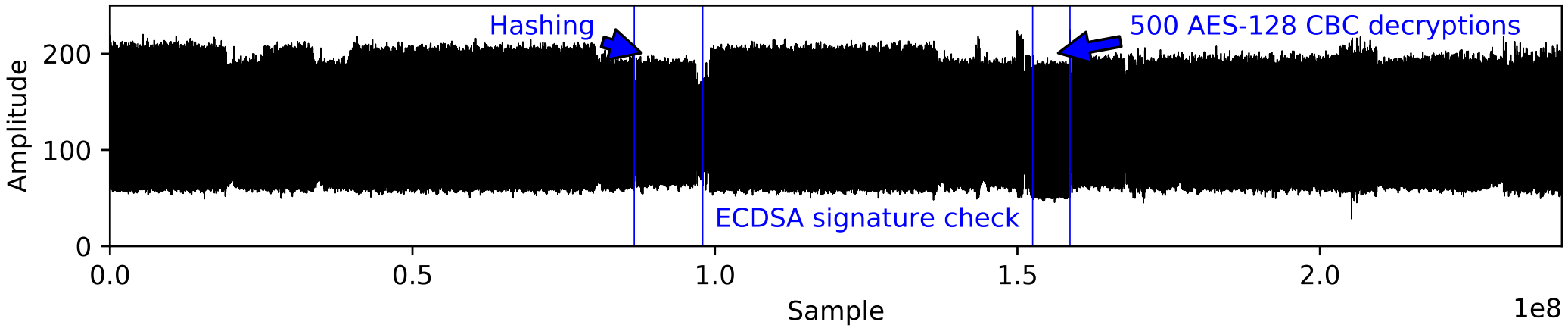


Evaluation

AES-128 in a Secure Boot Scenario

- Secure boot on a ARM Cortex-M4 microcontroller (STM32F303@8MHz):
  - Encrypted boot content (8 kB) on Non-Volatile Memory (NVM)
  - Hashing, ECDSA signature check, 500 AES-128 CBC decryptions
- Attacker capabilities:
  - Read out encrypted boot image from NVM
  - Knowledge of the different boot stages and specifics about the AES decryption:
    - ► AES mode and implementation: AES-CBC
  - Record a power trace of the entire boot process (250M samples)

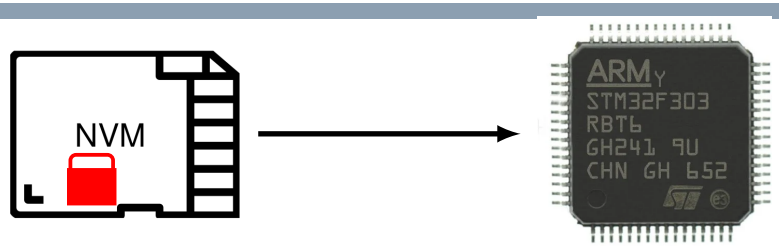


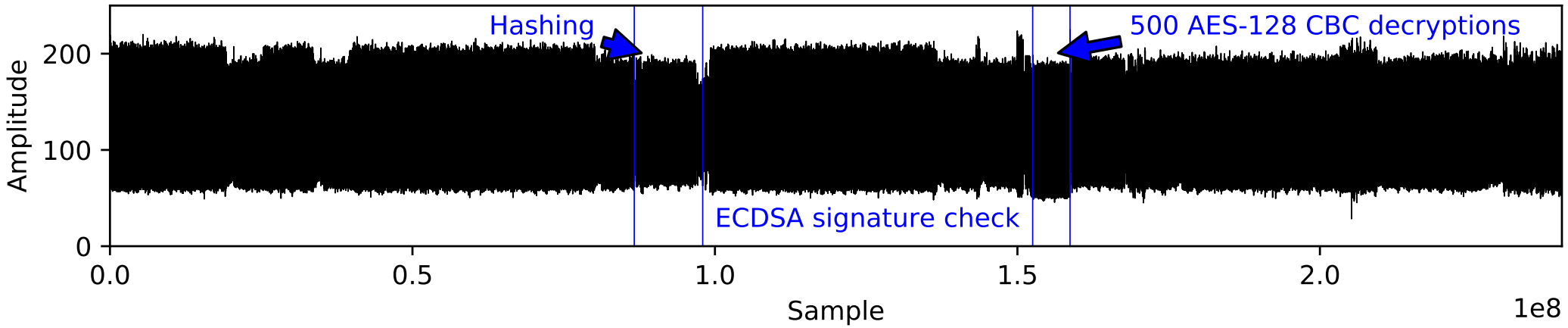


Evaluation

AES-128 in a Secure Boot Scenario

- Secure boot on a ARM Cortex-M4 microcontroller (STM32F303@8MHz):
  - Encrypted boot content (8 kB) on Non-Volatile Memory (NVM)
  - Hashing, ECDSA signature check, 500 AES-128 CBC decryptions
- Attacker capabilities:
  - Read out encrypted boot image from NVM
  - Knowledge of the different boot stages and specifics about the AES decryption:
    - ► AES mode and implementation: AES-CBC
  - Record a power trace of the entire boot process (250M samples)





- Correlation Power Analysis (**CPA**) of **AES-CBC decryption** is a possible attack vector!

Evaluation

AES-128 in a Secure Boot Scenario - Workflow

- Extract all $n = 500$ ciphertexts $\boldsymbol{ct}$ from external Non-Volatile Memory (NVM)

Evaluation

AES-128 in a Secure Boot Scenario - Workflow

- Extract all $n = 500$ ciphertexts $\boldsymbol{ct}$ from external Non-Volatile Memory (NVM)
- 40 different widths tested, tool considered three plausible, one was selected after looking at the template produced

Evaluation

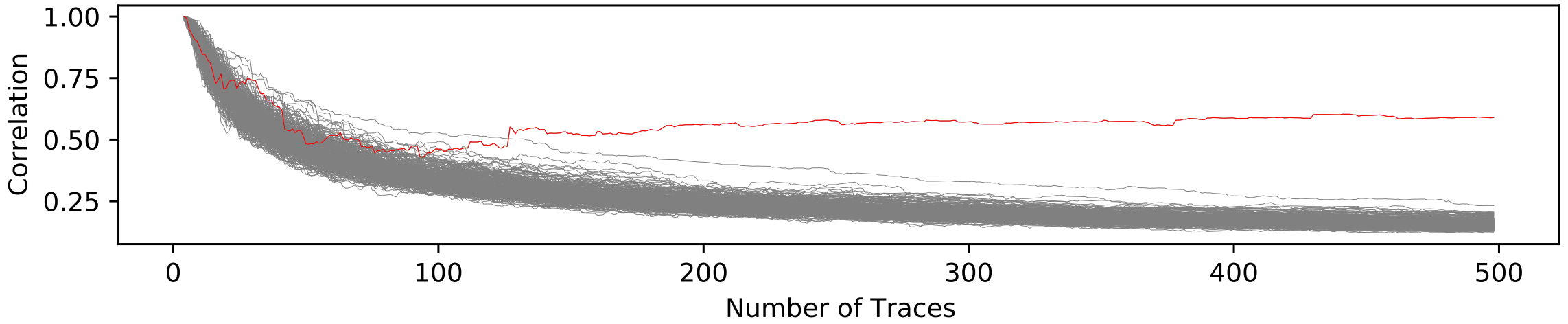
AES-128 in a Secure Boot Scenario - Workflow

- Extract all $n = 500$ ciphertexts $\boldsymbol{ct}$ from external Non-Volatile Memory (NVM)
- 40 different widths tested, tool considered three plausible, one was selected after looking at the template produced
- Each trace $\boldsymbol{t}_i$ with its CO parameters $\Pi_i = \boldsymbol{ct}_i$ was directly used for CPA without post-processing or alignment

Evaluation

AES-128 in a Secure Boot Scenario - Workflow

- Extract all $n = 500$ ciphertexts $\boldsymbol{ct}$ from external Non-Volatile Memory (NVM)
- 40 different widths tested, tool considered three plausible, one was selected after looking at the template produced
- Each trace $\boldsymbol{t}_i$ with its CO parameters $\Pi_i = \boldsymbol{ct}_i$ was directly used for CPA without post-processing or alignment
- Full key recovery was possible after 200 traces
- Unsupervised analysis took two hours, selecting best candidate and attack a few minutes:



Summary and Future Work

Summary

- Presentation of a semi-automatic algorithm that can find COs in side-channel traces
- Partly unsupervised analysis helps to search through large traces
- Functionality based on meta information(n, r, f_{device}) $\rightarrow$ no template necessary
  - Limitations: random delays, (highly) unstable clock frequency
- Implementation that utilizes parallel GPU computation
- Open-source implementation and CHES artifact (https://artifacts.iacr.org/tches/2022/a8/)

Summary and Future Work

Summary

- Presentation of a semi-automatic algorithm that can find COs in side-channel traces
- Partly unsupervised analysis helps to search through large traces
- Functionality based on meta information(n, r, f_{device}) $\rightarrow$ no template necessary
  - Limitations: random delays, (highly) unstable clock frequency
- Implementation that utilizes parallel GPU computation
- Open-source implementation and CHES artifact (https://artifacts.iacr.org/tches/2022/a8/)

Future Work

- Evaluate on other architectures (FPGAs) and limitations in terms of noise [TTW22]

Summary and Future Work

Summary

- Presentation of a semi-automatic algorithm that can find COs in side-channel traces
- Partly unsupervised analysis helps to search through large traces
- Functionality based on meta information(n, r, f_{device}) $\rightarrow$ no template necessary
  - Limitations: random delays, (highly) unstable clock frequency
- Implementation that utilizes parallel GPU computation
- Open-source implementation and CHES artifact (https://artifacts.iacr.org/tches/2022/a8/)

Future Work

- Evaluate on other architectures (FPGAs) and limitations in terms of noise [TTW22]

Thank you for your attention.

Do you have any questions?

jens.trautmann@fau.de

References

Refernces

[Bal+15] J. Balasch, B. Gierlichs, O. Reparaz, and I. Verbauwhede. “DPA, Bitslicing and Masking at 1 GHz”. In: *CHES 2015*. Ed. by T. Güneysu and H. Handschuh. Vol. 9293. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 599–619. DOI: 10.1007/978-3-662-48324-4_30.

[Bec+16] A. Beckers, J. Balasch, B. Gierlichs, and I. Verbauwhede. “Design and Implementation of a Waveform-Matching Based Triggering System”. In: *Constructive Side-Channel Analysis and Secure Design*. Ed. by F.-X. Standaert and E. Oswald. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 184–198. DOI: 10.1007/978-3-319-43283-0_11.

[Lis+21] O. Lisovets, D. Knichel, T. Moos, and A. Moradi. “Let’s Take It Offline: Boosting Brute-Force Attacks on iPhone’s User Authentication through SCA”. In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* (July 2021), pp. 496–519. DOI: 10.46586/tches.v2021.i3.496-519.

[Ron+17] E. Ronen, A. Shamir, A. Weingarten, and C. O’Flynn. “IoT Goes Nuclear: Creating a ZigBee Chain Reaction”. In: *2017 IEEE Symposium on Security and Privacy (SP)*. May 2017, pp. 195–212. DOI: 10.1109/SP.2017.14.

[TTW22] J. Trautmann, J. Teich, and S. Wildermann. “Characterization of Side Channels on FPGA-based Off-The-Shelf Boards against Automated Attacks”. In: *2022 IEEE 30th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*. 2022, pp. 1–9. DOI: 10.1109/FCCM53951.2022.9786190.