Security Threats and Countermeasures for Approximate Arithmetic Computing [2]

Pruthvy Yellu, Mezanur Rahman Monjur, Timothy Kammerer, Dongpeng Xu & Qiaoyan Yu
Department of Electrical and Computer Engineering, University of New Hampshire

University of New Hampshire

Introduction

- **Approximate Computing(AC) is a technique which trades accuracy for providing better energy efficiency.**
- **Approximate Computing can be applied at four different levels.**
- **However, Approximate Computing techniques are vulnerable to different security vulnerabilities.**

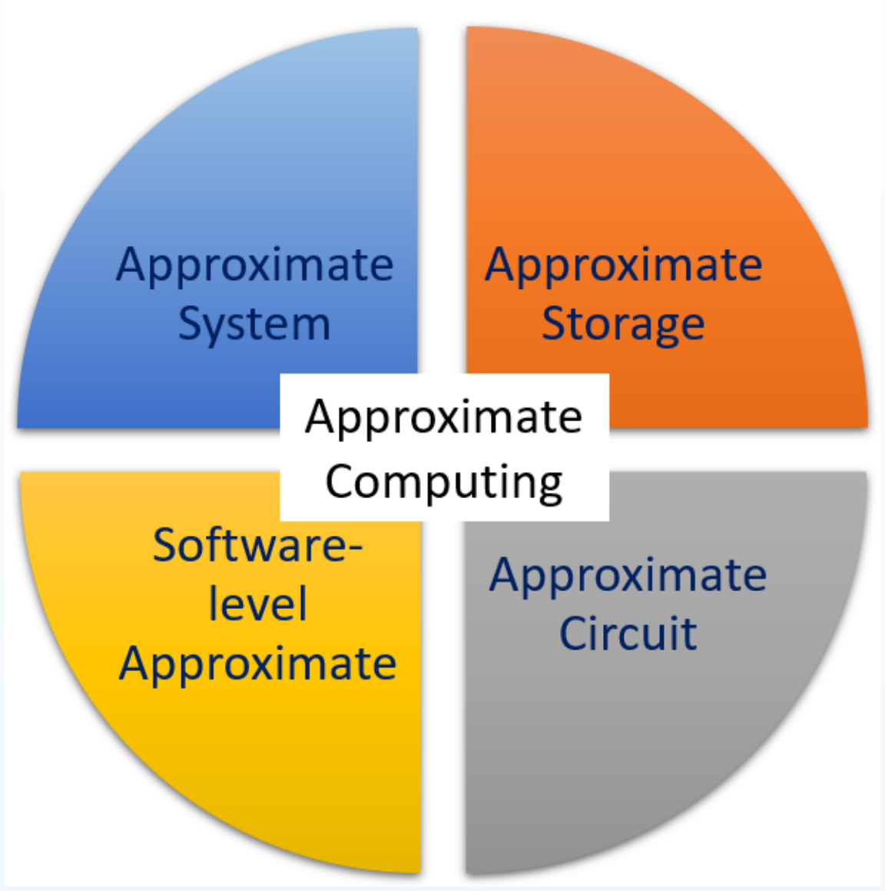

Fig.1. Strategies of AC [1].

- **In our work, we used approximate arithmetic adders to show the security vulnerabilities.**

Motivation Example

- **Approximate Floating-Point Adder**
  - **A 64-bit approximate floating point adder is used to examine the effect of approximate adder.**
  - **One of the adder inputs is a random number and the other input is fed from the adder output.**

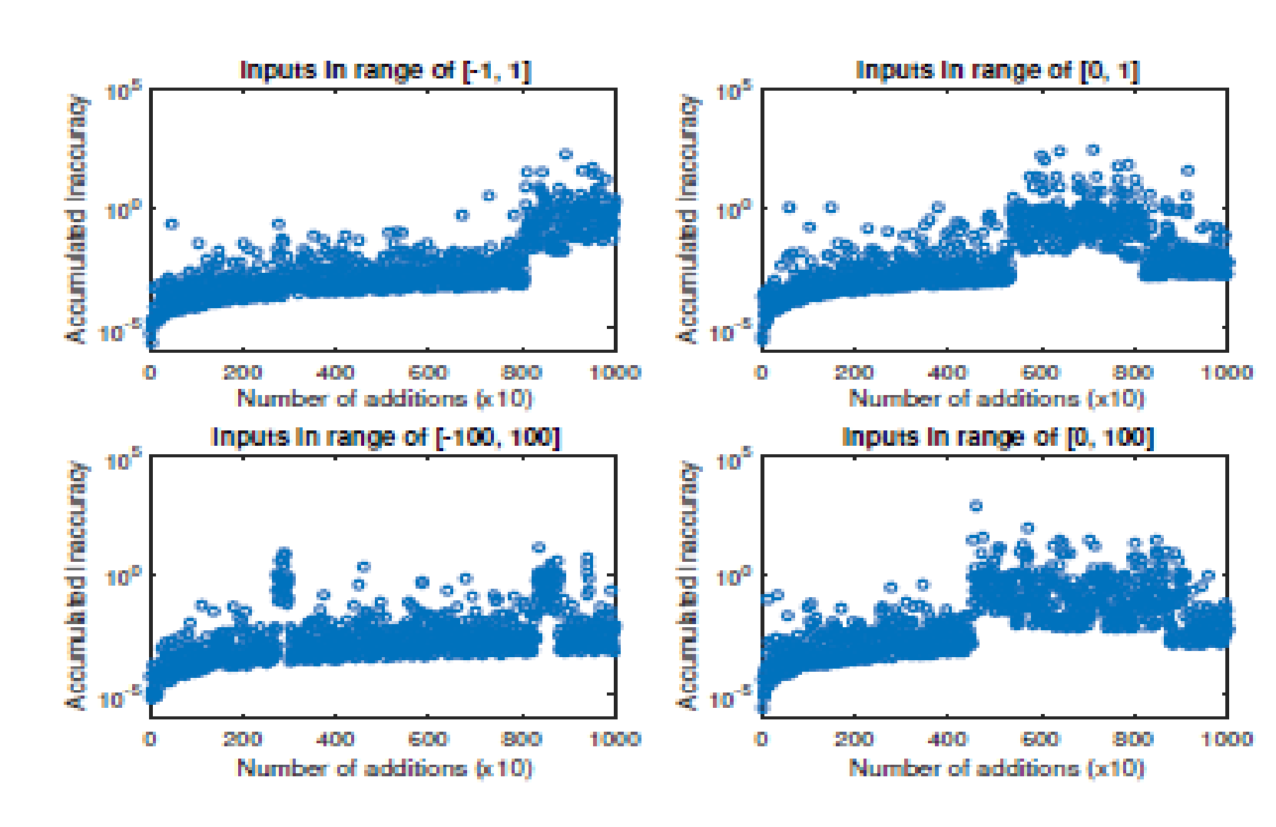

Fig.2. Accumulated inaccuracy of a 64-bit approximate adder.

Proposed Attack Model

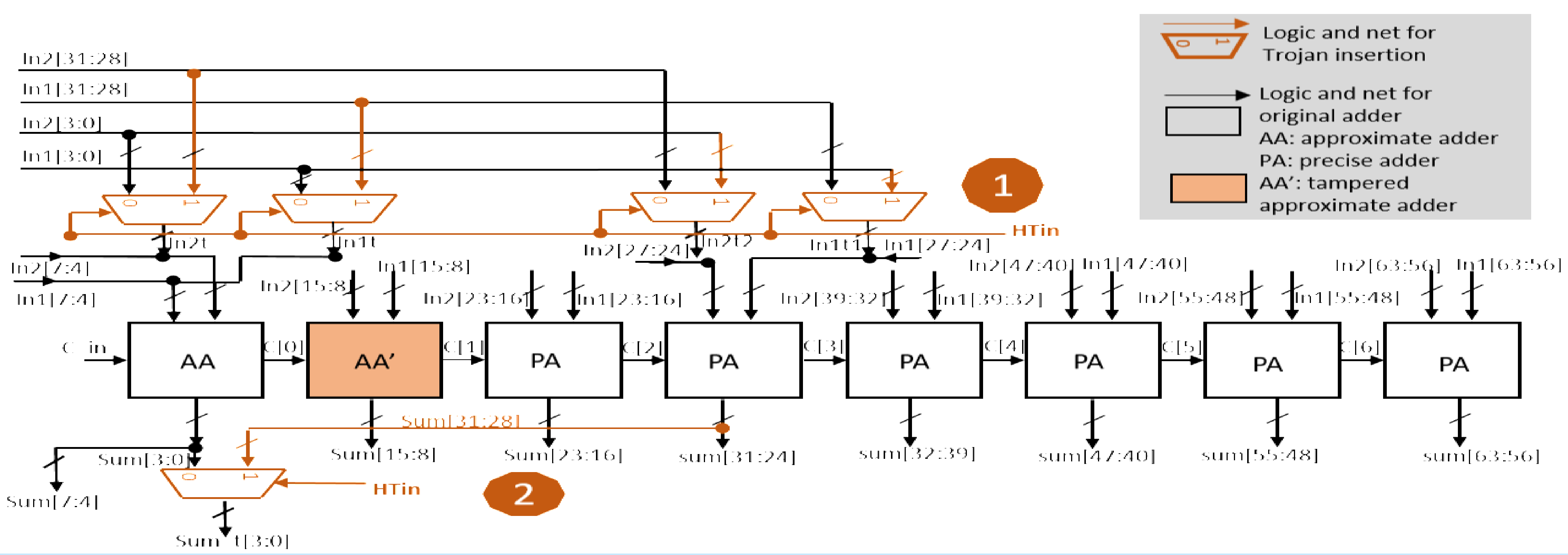

Fig.3. Hybrid adder with hardware Trojan payloads.

Proposed Countermeasures

- **The Input Integrity Check and Output Integrity Check modules examines if the inputs to the precise and approximate IP's are compromised.**
- **The exclusive logic based attack detection module, selectively examines if the outputs are compromised.**

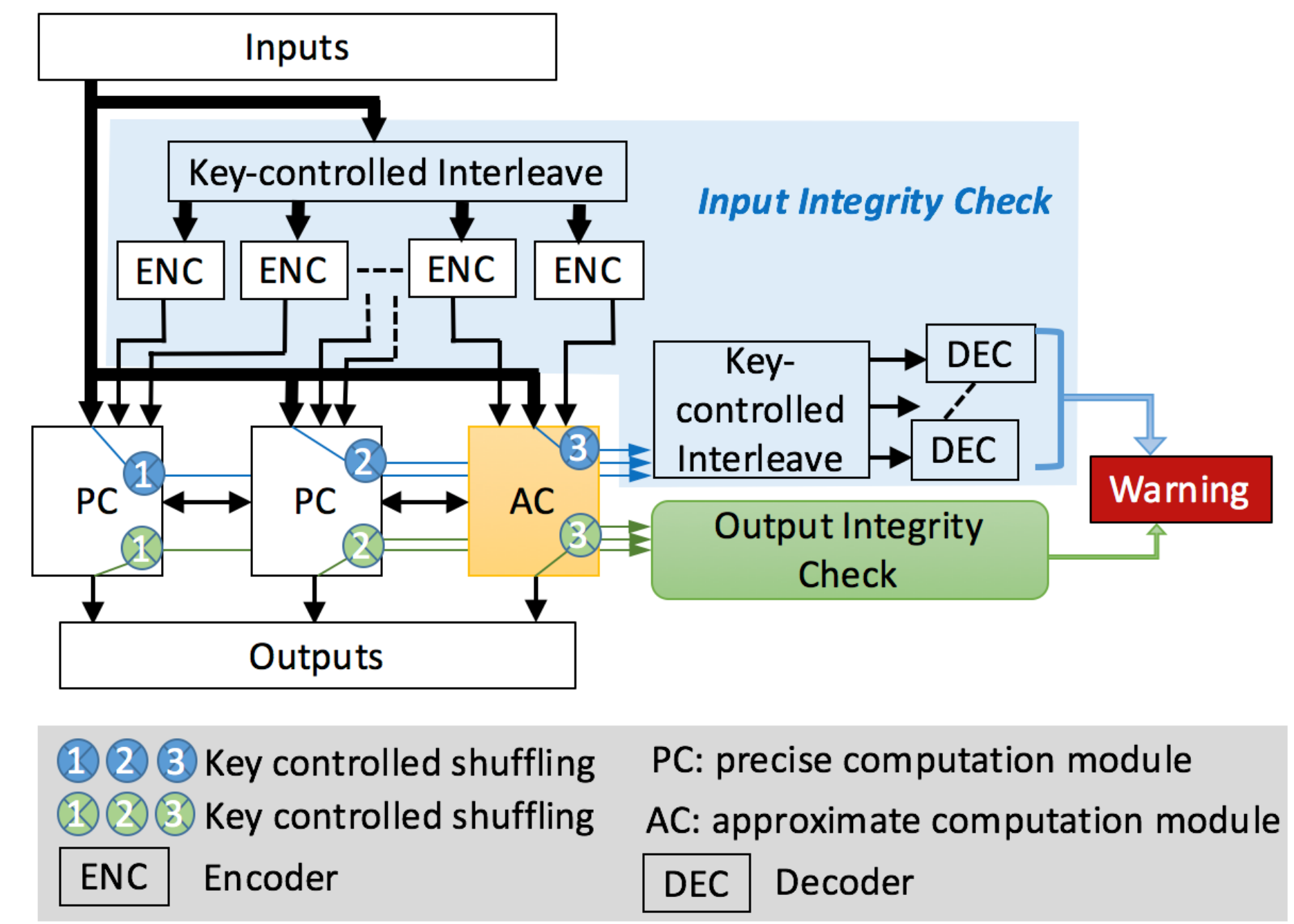
Inputs
Key-controlled Interleave
Input Integrity Check
ENC
ENC
ENC
ENC
Key-controlled Interleave
DEC
DEC
PC
PC
AC
Warning
Output Integrity Check
Outputs
1 2 3 Key controlled shuffling
1 2 3 Key controlled shuffling
ENC Encoder
PC: precise computation module
AC: approximate computation module
DEC Decoder

Fig.4. Proposed countermeasure.

Experimental Results

- **Attack on DCT-IDCT application**
  - **The case (a) shows the original image.**
  - **Case (b) is the sabotaged output after DCT (2 bits swapped).**
  - **Case (c) is the sabotaged output when 4 bits are swapped after DCT.**

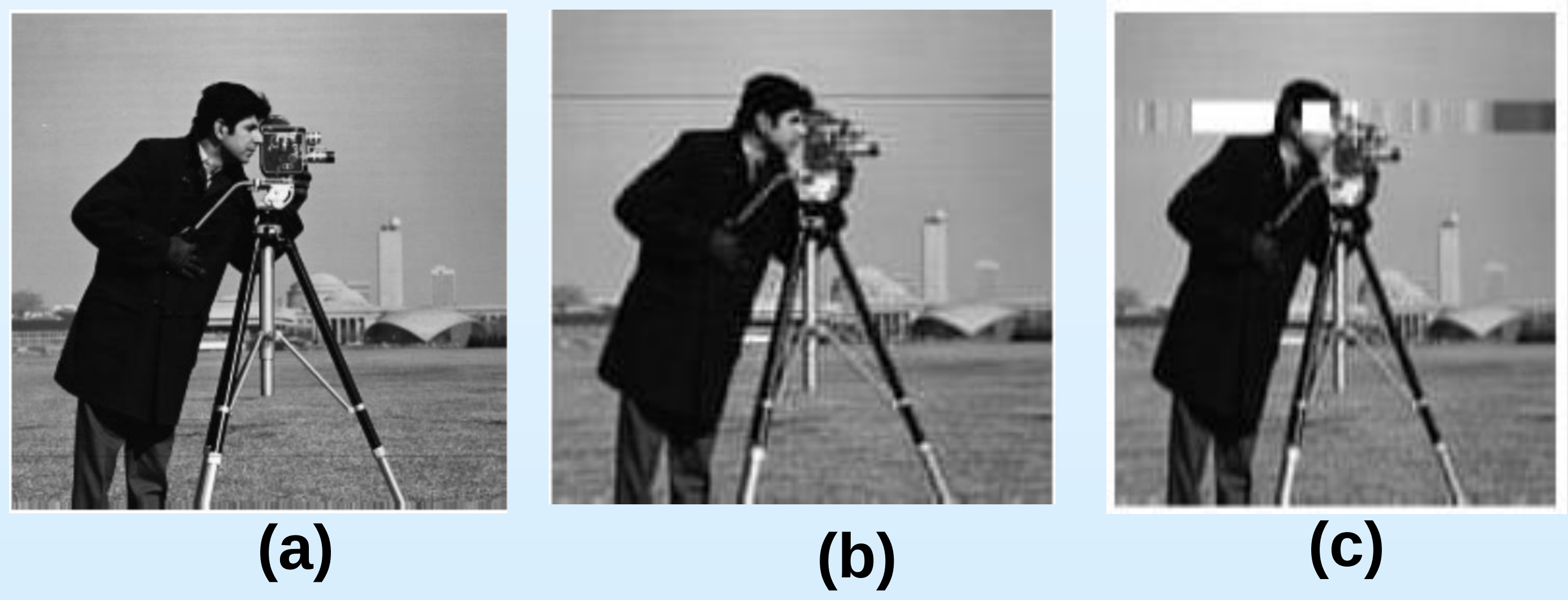

Fig.5. Attack Example.

- **Assessment of proposed countermeasures**
  - **Case (a) shows the attack detection failure rate is in the range of 2.2 ∗ 10^−3 and 8.087 ∗ 10^−4.**
  - **Case (b) shows that our attack detection failure rate remains consistent for the block size of 8 and 32.**

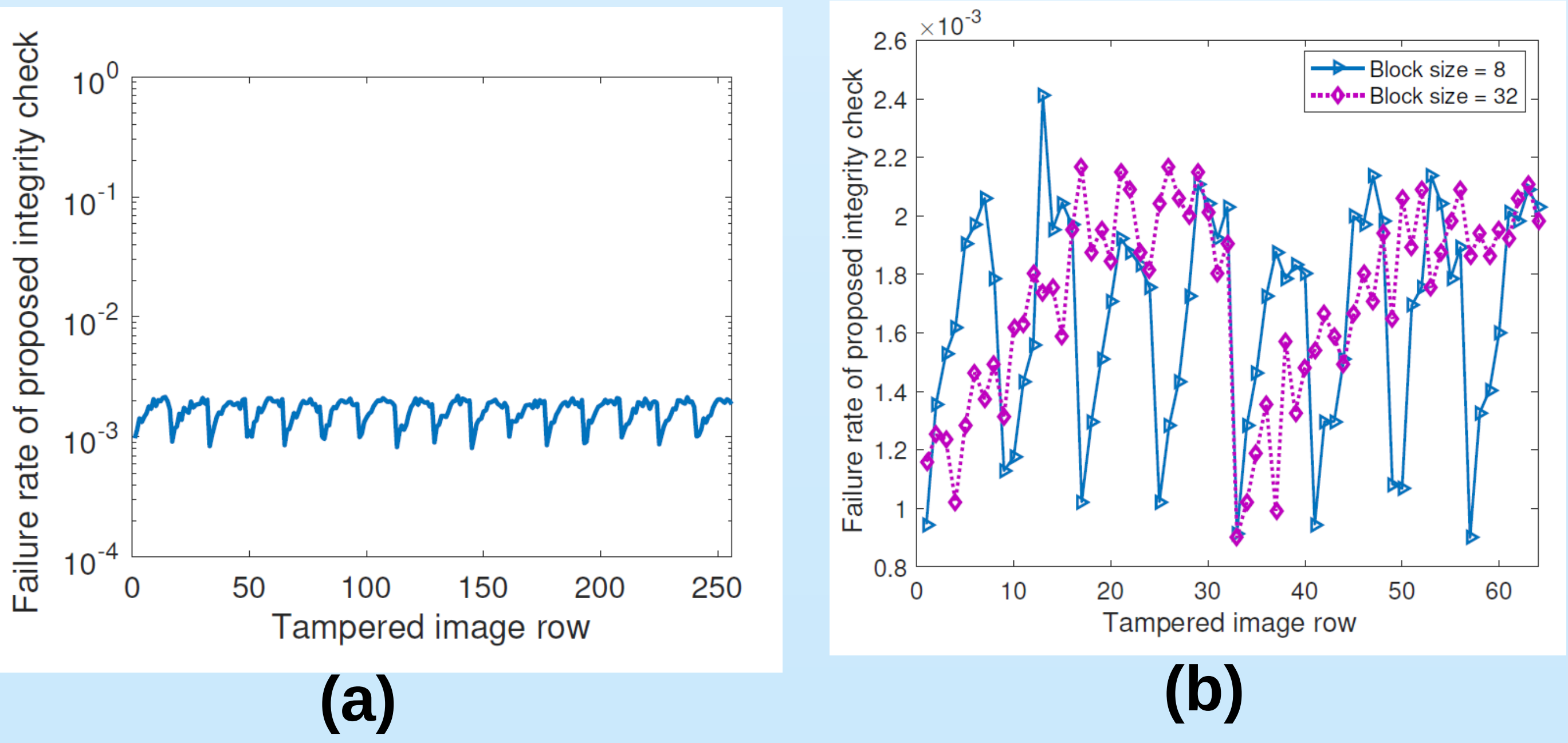

Fig.6. Failure rate of proposed countermeasure.

Conclusion

- **Our work is a preliminary effort to investigate and propose counter measures for security issues in Arithmetic AC.**

[1] P.Yellu et.al., "Security Threats in Approximate Computing Systems", GLSVLSI, 2019. [2] P.Yellu et.al., "Security Threats and Countermeasures for Approximate Arithmetic Computing ", ASPDAC, 2020.