



BNL-224755-2023-JAAM

# Contact-Less Integrity Verification of Microelectronics Using Near-Field EM Analysis

J. Huan, S. Mandal

To be published in "IEEE Access"

July 2023

Instrumentation Division  
**Brookhaven National Laboratory**

**U.S. Department of Energy**  
USDOE Office of Science (SC), Basic Energy Sciences (BES) (SC-22)

Notice: This manuscript has been authored by employees of Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy. The publisher by accepting the manuscript for publication acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

## **DISCLAIMER**

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or any third party's use or the results of such use of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof or its contractors or subcontractors. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

# Contact-less Integrity Verification of Microelectronics Using Near-Field EM Analysis

Junjun Huan, Peyman Dehghanzadeh, Soumyajit Mandal, *Senior Member, IEEE*, and Swarup Bhunia, *Senior Member, IEEE*

**Abstract**—Modern microelectronics life-cycle and supply chain ecosystem bring multiple untrusted entities, which can compromise their integrity. A major integrity issue of microelectronics stems from piracy of intellectual properties (IP) and counterfeiting, which causes significant revenue loss to the semiconductor manufacturers. Further, these components often lead to compromised functionality, reliability, security, and safety of an electronic system. This paper presents secure information transmission and probing methods for verifying the integrity of digital integrated circuit (ICs) based on their electromagnetic (EM) near-field emissions and thereby protecting systems against counterfeit components. The proposed method has been tested on both high-level instructions executed by microprocessors or Systems-on-Chip (serving as examples of software), and also logic circuits within FPGA fabrics and ASICs (serving as examples of hardware). The authentication information required by each digital system is generated using a pseudo-random number generator circuit and securely transmitted via near-field magnetic emissions. The authorized party can probe these emissions using a near-field probe, process the acquired signals to improve the signal-to-noise ratio (SNR), and then recover the secure information through matched filtering. Experimental results from commercial SoCs are used to demonstrate the proposed technique. Methods for reducing EM interference during integrity verification of both FPGAs and ASICs are also described.

**Index Terms**—Near-field electromagnetic emission, hardware integrity verification, FPGA fabric, System on Chip, linear-feedback shift register, Counterfeit electronics.

## I. INTRODUCTION

**H**ARDWARE and software integrity verification are both highly complex tasks, which have been explored for decades by researchers in the area of cryptography [1]–[5]. Counterfeit and substandard microelectronic components in the modern supply chain poses significant threats to protection of hardware intellectual property (IP), and functionality, reliability, security, and safety of an electronic system. There is a critical need to identify and maintain integrity of the electronic components used to build electronic systems, which are deployed in diverse sectors. Current mainstream hardware or software authenticating methods include inserting a fingerprint or digital signature into a target and using a sensor to read the fingerprint information after it has been assembled on a printed circuit board (PCB). This scheme can be implemented by using a test pin or interface, such as the JTAG debugging and programming port that is often found on targets such as

Field Programmable Gate Arrays (FPGA) or hard processor system (HPS) chips [6]. However, even if all the required test interfaces are available, this signature identification technique is time-consuming and frequently inaccessible on consumer products.

Unintentional side-channel leakage of secure information poses severe threats to the integrity of both hardware and software IP designs, since adversaries tend to exploit this vulnerability to eavesdrop and tamper important data and information from the design for their own benefits. Therefore, cryptographic countermeasures have been extensively explored for protecting any important information from leaking to an untrustworthy party through power, EM, and cache-based side channel attacks [7]–[9]. By contrast, for more than a decade, side-channel leakage has also been researched and leveraged to intentionally emit secure information to a trusted party with the information encrypted to circumvent unauthorized access. The authors in [10] present the idea of embedding secure information as a fingerprint or watermark of a hard IP in an FPGA logic circuit and using power side-channel measurement and analysis to detect and recover the fingerprint for authentication purposes. This side-channel watermarking technique requires physical contact between the electrical current probe and the power pin of the FPGA chip and is limited to authenticating hardware designs. The work in [11] introduces a design that uses a power side channel to leak secure information from an FPGA. However, this work focuses on the ability of an adversary to securely leak information by introducing a hardware Trojan into the design.

This paper extends the range of applications for intentional side-channel leakage by developing a data transmission methodology based on near-field EM leakage to transfer secure information from either hardware or software to an authorized receiver. The proposed approach provides a novel, effective, and convenient non-contact alternative to traditional electrical probing or readout scan-based techniques for device authentication. In particular, we present the methodology, design, and experimental results of two separate non-contact authentication systems for integrity verification of hardware (typically FPGA logic circuits or ASICs) and software (typically HPS instructions). Both systems use a novel data transmission methodology based on EM side-channel communication to transmit secure information from the information carrier (typically a piece of hardware or software within an IC) to a receiver. Obfuscation mechanisms are included to ensure that only authorized users can access secure information through non-contact sensing and EM side-channel analysis. Several

J. Huan, P. Dehghanzadeh and S. Bhunia are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, USA. Email: junjun.huan@ufl.edu, swarup@ece.ufl.edu.

S. Mandal is with Brookhaven National Laboratory, Upton, NY 11973, USA. Email: smandal@bnl.gov.

techniques for protecting against EM interference (EMI) are also proposed to ensure that the verification systems operate reliably on both custom-made ASICs and commercial FPGAs.

The remainder of the paper is organized as follows. Section II discusses the background and motivation for this work, while Section III presents the design of the proposed hardware and software integrity verification systems. Experimental results are discussed in Section IV, while Section V summarizes our contributions and concludes the paper.

## II. BACKGROUND AND MOTIVATION

The proposed authentication systems can 1) insert secure information (e.g., a unique bit pattern) within each target (typically an IC); and 2) recover this information in the form of near-field magnetic emissions by probing the target without physical contact or access to test interfaces. In general, our methodology can be used for authenticating hardware and software at different stages of their supply chains. In one use-case, the methodology is applicable for reading a physical unclonable function (PUF) output signature [12] from an IC without directly contacting the chip, PCB traces, or other connected components on a PCB retrieved from the field. Another application may see its use purely on an FPGA fabric. Here, the methodology can enable secure information to be embedded as a watermark or a tag [13] and allow for magnetic field emission detection by a reliable party. From the software perspective, this methodology is also highly relevant and can facilitate wireless control flow integrity verification in a HPS during run-time software integrity check like in [14]. Another promising benefit of our proposed methodology is the ability to authenticate a target in an electronic system without taking the system apart - either immediately after its procurement or at run time during field deployment of the system - which is a novel capability in both the research and industrial domains. For example, a user can authenticate the hardware or software core within an embedded system or a device such as a server from its enclosure. More importantly, our methodology enables secure transmission of other types of information from an IC (such as medical sensor outputs or biometric information that is private to the user [15]) without restrictions on the type of authenticating hardware and software. All these proposed applications are based on successfully sensing strong EM emissions from a hardware or software target.

### A. EM Signal Generation

1) *Authentication information generation*: To generate information that can be securely transmitted and is unique to each IC for integrity verification purposes, we use pseudo-random number generation (PRNG) circuits built using logic gates or software functions for hardware or software authentication, respectively [16]. The circuit is designed to generate as many unique and unpredictable electrical signals that contain digital signature information as possible, such that hardware or software in massive quantities can be authenticated and secure information is not susceptible to interception by attackers and potential hardware or software counterfeiting [17]. These

unique digital signatures can then be emitted from the IC in the form of EM emissions.

One example of a PRNG is a linear-feedback shift register (LFSR) circuit. An LFSR is essentially a parallel-in serial-out shift register whose input bits are determined by a linear feedback function of their previous states [18]. The most common feedback function uses a set of exclusive-or (XOR) gates (acting as modulo-2 adders) that define a feedback polynomial; the latter is chosen to maximize the repetition length and randomness of the output bit sequence. Fig. 1(a) presents the schematic of a 32-bit Galois-field LFSR circuit with feedback taps at locations 1, 5, 6, and 31. These taps are chosen to maximize the number of possible states such that an output bit sequence of maximal length is generated. For a 32-bit LFSR, the maximal length of the output stream is  $2^{32} - 1$  bits since the all-zero state is forbidden. The number of possible  $n$ -bit signatures that can be used for optimization is thus given by  $S(n) = (2^{32} - 1)/n$ , where each signature consists of  $n$  LFSR states. For modest values of  $n$ , the value of  $S(n)$  is large enough to prevent attackers from easily guessing the signature. To further improve the security of the system, an longer LFSR circuit can be used to increase the number of possible signatures. The outputs of multiple LFSR circuits can also be combined (via an XOR operation) to increase the complexity of signature decryption by rogue parties. The number of signatures which can be stored in the file system is only limited by the system's storage capacity.

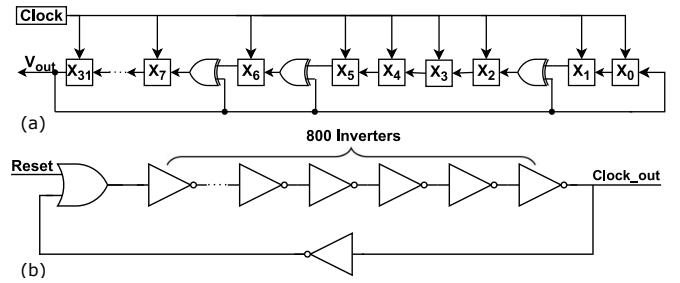


Fig. 1. (a) Schematic of a 32-bit Galois-field LFSR with feedback taps at positions 1, 5, 6, and 31. (b) Schematic of a ring oscillator (RO) circuit with a nominal output frequency of  $\sim 2.4$  MHz when implemented on a Cyclone V SoC.

2) *Security improvement using inter-chip variations*: Ring oscillator (RO) circuits can be implemented within FPGA- or ASIC-based systems to introduce frequency variations to the magnetic field emission signals that carry the signature information. This is an effective approach for allowing more IC chips to be authenticated and confusing attackers with more possible EM signature guesses. The oscillation frequency of an RO is determined by the aggregate propagation delays of all the delay elements (typically CMOS inverters) within the circuit. These delays are sensitive to process, voltage, and temperature (PVT) variations, leading to both intra- and inter-die variations in the oscillation frequency. The latter can serve as 1) a source of entropy for magnetic field emission-based measurements, and 2) additional key bits for bit-stream encryption when authenticating FPGA logic circuits. Fig. 1(b) shows a ring oscillator comprising a total of 801 inverters

that outputs a square-wave clock signal at a frequency of  $\sim 2.4$  MHz under nominal PVT conditions when implemented on a Cyclone V SoC. An active-low reset signal is applied to start or stop the oscillations.

3) *Error correction mechanism*: Even when the proposed magnetic field emissions used for hardware and software authentication are detected with high SNR, it is still possible for external interference to generate bit errors while reconstructing the digital signature. Potential interference sources include 1) other logic circuits operated in the FPGA fabric, and 2) software functions running on the HPS. Thus, an error correction mechanism is needed to detect or correct potential bit errors. For example, here we use a Hamming code that can detect and correct 1-bit errors.

A Hamming code encodes the input data with parity bits (or redundant bits) inserted at certain positions, namely those which are powers of 2 (i.e., positions 1, 2, 4, 8, 16, 32, ...) to generate a final Hamming-encoded vector. The total number of parity bits is determined by the expression:

$$2^N \geq n + N + 1, \quad (1)$$

where  $N$  is the number of parity bits and  $n$  is the length of the input data. For a 32-bit digital signature, a minimum of 6 parity bits are required to detect and correct single-bit errors. Since each parity bit  $P_x$  governs different data bits  $D_x$ , parity bits  $P_1, P_2, P_4, P_8, P_{16}$  and  $P_{32}$  at positions 1, 2, 4, 8, 16 and 32 of the final (32, 26) Hamming vector can be computed using Eqn. (2), where  $D_1 \sim D_{32}$  are the data bits of a 32-bit signature. Errors in the data bits can then be detected as mismatches between the parity bit values of the original and the reconstructed digital signatures, respectively. Also, a single-bit error can be corrected through the syndrome decoding method [19].

## B. EM Signal Sensing

1) *Near-field EM emissions from ICs*: The encoded signature information can be transmitted from an IC through two possible types of EM emissions, namely magnetic or electric field emissions. Magnetic field emission usually results in higher SNR than its electric field counterparts. Physically, this is because of two factors: i) lower ambient noise for the magnetic field, and ii) the fact that most everyday materials (apart from ferromagnets) are non-magnetic and thus do not affect magnetic fields, while almost all of them are dielectric and thus strongly affect electric fields. Given this observation, here we focus on non-contact probing of magnetic field emissions from the embedded authentication systems after the corresponding ICs have been assembled on a PCB. Generally, the magnetic field emissions include near-field emissions from the internal circuitry, conductive emissions from connected PCB traces, and direct emissions from bond wires within the IC package [20]. However, near-field emissions are the main focus of our work, since PCB traces may not be readily accessible for probing in some cases. Examples include Internet of Things (IoT) applications that use 3D integrated technology for on-chip wireless data links [21], or ingestible sensor networks within the human body [22]. Additionally, the high frequency

of direct emissions (typically in the GHz range) makes it hard to detect embedded signatures with low distortion and noise.

The mechanism of near-field magnetic emissions from an IC is illustrated in Fig. 2 [20]. Near-field emissions consist of two parts: 1) field  $\vec{H}_2$  generated from transient current loops across the internal IC; and 2) field  $\vec{H}_1$  formed around the ground plane of the PCB. The strength of field  $\vec{H}_2$ , which is localized within approximately 10 mm above the surface of the IC package, is much greater than field  $\vec{H}_1$  and is therefore the main source of near-field emissions detected by a magnetic field probe [20], [23]. The proposed system uses a near-field magnetic probe and broadband pre-amplifier to capture these emissions and then recover digital signature from them for hardware and software authentication.

2) *Near-field EM emission source modelling*: To further analyze near magnetic field emissions from an IC with respect to probing distance, we modeled a rectangular current loop on a 2-layer PCB with an FR4 dielectric layer in between as shown in Fig. 3(a). The ground pin of the loop is connected to the bottom copper layer through a via. We applied a current of 150  $\mu\text{A}$  to the loop and simulated the surrounding quasistatic magnetic field using an EM field solver (COMSOL Multiphysics). The amplitude of the magnetic field emissions on the PCB plane is shown in Fig. 3(b) as a color map. The result indicates that the magnetic field strength is the greatest along the traces, which is understandable since the current flowing through these traces acts as the field source.

The field amplitude at the center of the loop is plotted in Fig. 3(c) as a function of distance from the PCB plane,  $d$ . This figure shows that the magnetic flux density is maximal on the plane containing the loop but decays symmetrically with  $d$  both above and below this plane. Theoretically, the field may be approximated by that of a circular loop with the same area,  $A$ , which is given by

$$H_z(d) = \frac{\sqrt{\pi}}{2} \frac{AI}{(\pi d^2 + A)^{3/2}} \quad (2)$$

where  $I$  denotes the current and the loop is assumed to lie in the  $xy$ -plane. Eqn. (2) shows that  $H_z$  decreases  $\propto 1/d^3$  for distances larger than the characteristic size of the loop,  $\sqrt{A/\pi}$ . This dependence limits the maximum sensing distance of the proposed non-contact authentication method.

In addition, we studied the maximum distance at which magnetic field emissions from the loop are detectable as a function of  $I$ , the loop current. The amplitude of the minimally detectable magnetic field emission signal was calculated as about 0.134 A/m based on the following conditions: 1) measurement noise floor of 0.34 mV; 2) a minimum SNR of 2.5 dB for robust signal detection (50% probability of detection at a false alarm rate of 3.2% in Gaussian noise); 3) a magnetic probe sensitivity of 3 mVm/A at a frequency of 2.5 MHz; and 4) a pre-amplifier gain of 30 dB. Fig. 2(d) shows a graph of the maximum distance required to reliably detect the magnetic field emissions as a function of the current applied to the trace loop. The graph shows that the maximum sensing distance grows rapidly for currents  $< 20$  mA before saturating at  $\sim 12.5$  mm. In reality, the emitted field amplitude is typically weaker than in the simulation model due to the

$$\begin{aligned}
P_1 &= D_1 \oplus D_2 \oplus D_4 \oplus D_5 \oplus D_7 \oplus D_9 \oplus D_{11} \oplus D_{12} \oplus D_{14} \oplus D_{16} \oplus D_{18} \oplus D_{20} \oplus D_{22} \oplus D_{24} \oplus D_{26} \oplus D_{27} \oplus D_{29} \oplus D_{31}, \\
P_2 &= D_1 \oplus D_3 \oplus D_4 \oplus D_6 \oplus D_7 \oplus D_{10} \oplus D_{11} \oplus D_{13} \oplus D_{14} \oplus D_{17} \oplus D_{18} \oplus D_{21} \oplus D_{22} \oplus D_{25} \oplus D_{26} \oplus D_{28} \oplus D_{29} \oplus D_{32}, \\
P_4 &= D_2 \oplus D_3 \oplus D_4 \oplus D_8 \oplus D_9 \oplus D_{10} \oplus D_{11} \oplus D_{15} \oplus D_{16} \oplus D_{17} \oplus D_{18} \oplus D_{23} \oplus D_{24} \oplus D_{25} \oplus D_{26} \oplus D_{30} \oplus D_{31} \oplus D_{32}, \\
P_8 &= D_5 \oplus D_6 \oplus D_7 \oplus D_8 \oplus D_9 \oplus D_{10} \oplus D_{11} \oplus D_{19} \oplus D_{20} \oplus D_{21} \oplus D_{22} \oplus D_{23} \oplus D_{24} \oplus D_{25} \oplus D_{26}, \\
P_{16} &= D_{12} \oplus D_{13} \oplus D_{14} \oplus D_{15} \oplus D_{16} \oplus D_{17} \oplus D_{18} \oplus D_{19} \oplus D_{20} \oplus D_{21} \oplus D_{22} \oplus D_{23} \oplus D_{24} \oplus D_{25} \oplus D_{26}, \\
P_{32} &= D_{27} \oplus D_{28} \oplus D_{29} \oplus D_{30} \oplus D_{31} \oplus D_{32},
\end{aligned} \tag{2}$$

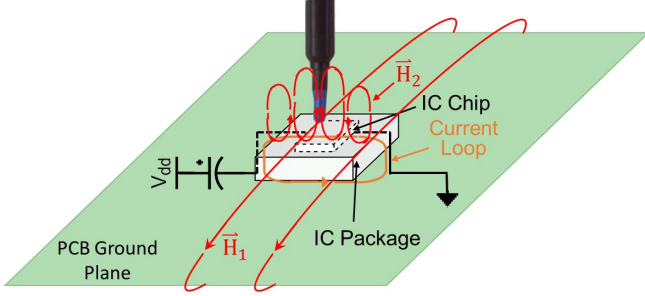


Fig. 2. Mechanism of near-field magnetic emissions from an IC.

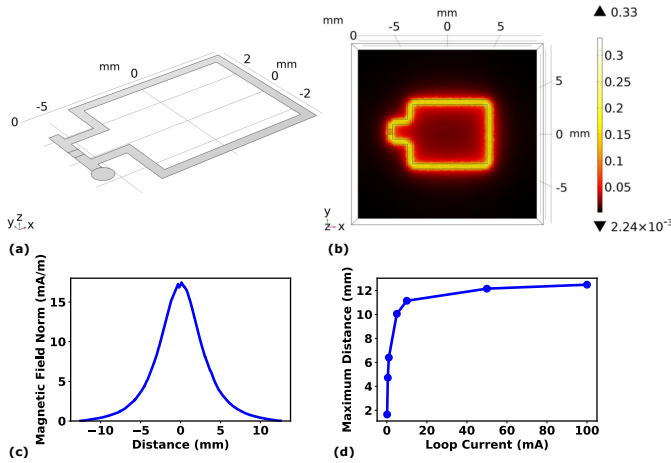


Fig. 3. Quasistatic magnetic field emitted from a  $150 \mu\text{A}$  current loop on a PCB. (a) 3D model of a trace loop on a 2-layer PCB. (b) Color map of the magnetic flux density (in T) across the PCB plane. (c) Magnetic field amplitude (in mA/m) versus distance from the PCB plane, as measured along a line through the center of the loop. (d) Maximum distance for reliable detection of magnetic field emissions as a function of loop current.

shielding provided by additional on-chip metal layers, thus limiting the useful sensing distance to  $\sim 10$  mm.

### C. Accurate Detection of Embedded Signatures

Accurate detection of the embedded signature information requires the SNR of the measured magnetic field emission signal to be maximized. There are two main methods for improving the SNR. The first is to increase the  $\vec{H}$ -field strength by optimizing the on-chip source. For example, the self-inductance of interconnects within gate arrays mapped to an

FPGA fabric can be increased by configuring routing and layout constraints during the floor-planning stage. By contrast, the second method relies on signal processing techniques, such as low-pass filtering, to minimize high-frequency noise in the sensed signal and thus increase SNR.

An additional signal processing technique is matched filtering, which is known to provide optimal accuracy for detection of known signals (i.e., minimal false error rate,  $P_{fa}$ , for a given probability of detection,  $P_d$ ) in white Gaussian noise. The technique can also be extended to situations where the noise is non-white by adding a whitening filter before the matched filter. For concreteness, let us define the known signal of interest in our case (i.e., the embedded digital signature or template) by  $s(t)$  and the noisy received data by  $r(t) = s(t) + n(t)$  where  $n(t)$  is additive white noise. The impulse response of the corresponding matched filter is

$$h_M(t) = s^*(t_0 - t) \tag{3}$$

where  $*$  denotes the complex conjugate and  $t_0$  is the time at which peak output SNR is obtained. In the time domain, the output of the matched filter,  $s_{out}(t)$ , is obtained by convolving  $r(t)$  with  $h_M(t)$ , resulting in

$$s_{out} = r(t) * h_M(t) = s(t) * s^*(t_0 - t) + n(t) * s^*(t_0 - t). \tag{4}$$

The first term in this expression is the desired signal, while the second is the filtered noise. Note that the convolution operation is equivalent to a cross-correlation, which requires  $\mathcal{O}(N^2)$  operations for an length- $N$  signal vector. Alternatively, matched filtering can be performed in the frequency domain, where the convolution becomes a multiplication, i.e.,

$$S_{out}(\omega) = R(\omega)H_M(\omega) = R(\omega)S^*(\omega). \tag{5}$$

The frequency response of the matched filter,  $H_M(\omega)$ , can be pre-computed, so the required operations reduce to 1) using a fast Fourier transform (FFT) to obtain the signal spectrum,  $R(\omega)$ ; 2) performing a point-by-point multiplication to find the output spectrum,  $S_{out}(\omega)$ ; and 3) using the inverse FFT to obtain the time-domain output,  $s_{out}(t)$ . Due to the efficiency of the FFT, this process only requires  $\mathcal{O}(N \log(N))$  operations, making it the preferred choice for real-time implementations.

The amount of improvement in SNR due to matched filtering depends on the bandwidth-time product of  $s(t)$ , the known signal or template. Consider a pulse-like template of length  $T_p$  and amplitude  $A$ . After matched filtering, this waveform is “compressed” to a duration  $\sim 1/B$  where  $B$  is its bandwidth (thus, this process is known as pulse compression

in radar systems). The filtering conserves signal energy, so the amplitude of the compressed pulse,  $A'$ , must satisfy

$$A^2 T_p = (A')^2 / B \Rightarrow A' = A \sqrt{B T_p}$$

where  $B T_p$  is the bandwidth-time product. Since the noise is uncorrelated with  $h_M(t)$ , its rms amplitude is unaffected by the matched filter, such that the output SNR improves by a factor of

$$\frac{SNR_{out}}{SNR} = \left( \frac{A'}{A} \right)^2 = B T_p. \quad (6)$$

The signals of interest in our integrity verification system (i.e., the embedded signatures) are pseudorandom bit streams, for which 1)  $T_p = n_{bit} f_{clk}$  where  $f_{clk}$  is the clock frequency and  $n_{bit}$  is the length of the signature; and 2)  $B \approx 1/f_{clk}$ . Thus, such waveforms have a bandwidth-time product of  $B T_p \approx n_{bit}$ , implying that the amount of SNR improvement provided by matched filtering is proportional to the length of the signature. Thus, if the sensing distance  $d$  is fixed, matched filtering allows the amplitude of the on-chip field source to be reduced by a factor of  $n_{bit}$  while preserving SNR. Alternatively, the fact that field amplitude decreases  $\propto 1/d^3$  implies that matched filtering can also be used to increase the maximum usable sensing distance by a factor of  $\sqrt[3]{n_{bit}}$  when the field source is kept fixed. For example, using a signature of length  $n_{bit} = 32$  bits improves SNR for a given field source by 15.1 dB. Assuming a loop current of 50 mA and the same sensing parameters as before, matched filtering also allows 32-bit signatures to be reliably detected at distances up to  $\sim 39.7$  mm (compared to the  $\sim 12.5$  mm shown in Fig. 2(d)).

#### D. Protection Against EM Interference (EMI)

In this section, we discuss several approaches to both i) protect a target IC against external EMI attacks, and ii) also prevent the target IC from generating its own EMI. These methods can be classified into two categories based on circuit design at the EM transmitter (i.e., the target IC) and signal processing at the EM receiver. Some EMI removal or prevention methods are implemented at the FPGA or SoC level using logic circuits or software instructions, respectively. For example, resistance to external EMI attacks on the hardware authentication system can be improved by using differential signaling both within the FPGA fabric and for the I/O pins. Most FPGA families feature built-in modules for converting single-ended I/O to differential protocols such as low-voltage differential signaling (LVDS) or current-mode logic (CML). Such differential signals are robust to common-mode noise, i.e., noise that appears with the same polarity at both the non-inverting and inverting input terminals of a differential amplifier, as shown in Fig. 4. Another method for improving resistance to EMI or EM-based attacks on an FPGA fabric is to copy the target circuit to different regions of the IC. As a result, this technique can help protect against localized EMI that affects only a subset of these copies. However, this method is ineffective against attacks that affect the entire chip.

ASICs offer designers additional options for eliminating and/or tailoring EMI. In particular, ASICs can include custom on-chip metalization patterns that are optimized to block

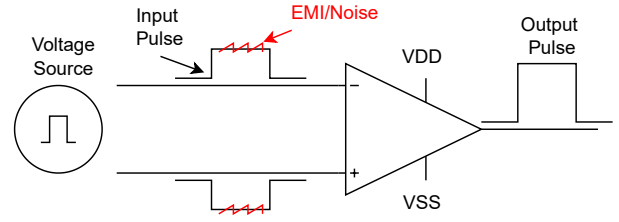


Fig. 4. The concept of using differential signaling to reduce the impact of common-mode noise, such as external EMI.

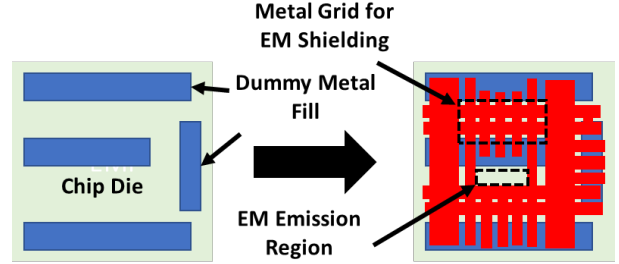


Fig. 5. Summary of the design flow for fabricating EM-shielding layers using dummy metal fills.

external EMI but still allow for EM emissions of the desired electrical signals. For example, unconnected “dummy” metal fills, which are typically used to planarize the chip surface, can be reconfigured to serve as an EMI shielding layer. Specifically, EMI shielding regions are formed by connecting squares of the (normally floating) dummy metals together to form metal grids with hole dimensions much smaller than the EMI wavelength, thus blocking external EMI. On the other hand, fill blocking layers are used to remove dummy metals from regions where desired EM emissions take place. The resulting design flow is summarized in Fig. 5.

At the receiver end, signal processing methods can be used to minimize noise and EMI within the signals recorded by the data acquisition (DAQ) system. Out-of-band EMI can be removed by using a band-pass filter, while in-band noise can be minimized by using a matched filter. As discussed in the previous section, a matched filter enhances signal components that match the selected template while suppressing unmatched components such as those due to EMI.

### III. HARDWARE AND SOFTWARE INTEGRITY VERIFICATION SYSTEMS

Fig. 6 summarizes the process used for secure non-contact transfer of information to/from an IC by using near-field emissions. A security-critical signal  $S$  (such as a digital watermark or signature) is chosen for transmission and mapped to a PRNG seed value. The PRNG, which can be implemented either as logic gates in hardware (an FPGA fabric or ASIC) or high-level instructions in software (a HPS), now generates the corresponding electrical signature signal,  $S_E$ . An error correction code is added to  $S_E$  and the result encrypted by a cipher function,  $C$ , with a key,  $K$ . The final encrypted electrical

signal is serialized and converted to an EM (mainly magnetic) signal,  $S_{EM}$ , in a bit-wise fashion by an EM transmitter.

An authorized party can use a near-field magnetic probe (EM receiver) to 1) detect the emitted  $\vec{H}$ -field signal,  $S_{EM}$ ; and 2) recover signature information by using a matched filter. We now describe the various steps within this integrity verification process in more detail.

#### A. FPGA fabric-based system

The proposed hardware integrity verification system is implemented on an FPGA fabric by using 1) a PRNG circuit to generate digital signatures, and 2) near-field emission and sensing. As shown in Fig. 7, the system contains two main modules: 1) a transmitter consisting of a file system, an HPS, and an FPGA fabric; and 2) a receiver consisting of a  $\vec{H}$ -field probe, a DAQ system (high-speed oscilloscope), and a signature detection module. The file system in the transmitter stores two lists of millions of pairs of signature values that are generated offline by a PRNG circuit and the corresponding seed values. The HPS acquires an intended digital signature value to be written to the hardware from a user input, searches for a possible match in the signature list, and outputs the corresponding seed value to a PRNG in the FPGA fabric that generates the final  $\vec{H}$ -field signal. Hence any digital signature value inputted by the user can be mapped to an  $\vec{H}$ -field signature, as summarized in Fig. 8.

The receiver integrates a  $\vec{H}$ -field probe with a DAQ (high-speed oscilloscope) for measurement of magnetic field emission signals. The oscilloscope uses an external trigger signal to activate each measurement and define one signature period during which real-time signal averaging can be implemented on the acquired signal to improve the SNR. The signature detection module at the final stage helps significantly enhance the SNR of the measured magnetic field signature by 1) minimizing out-of-band noise through a low-pass filter; and 2) detecting the signature waveform through matched filtering.

The FPGA fabric generates the electrical signal that carries the digital signature information, and is thus the source of the  $\vec{H}$ -field emissions. In a typical implementation, the FPGA fabric contains modules for random number generation, error correction, and parallel-in to serial-out conversion. The PRNG circuit can be an LFSR that receives a seed value (the initial state of the shift registers) and generates a 1-bit output message at the rising edge of each clock cycle. Two synchronous LFSR circuits with different seed values using an XOR operation to output 1-bit message can be used to increase the maximal sequence length, thus improving the security of the signature. An  $n$ -bit digital signature with high entropy is generated by running the PRNG for  $n$  clock cycles. The latter is then encoded by an error correction algorithm (a Hamming code) for single bit error detection and correction. The final encoded bitstream (also known as a Hamming vector) is finally serialized, encrypted by a symmetric key for obfuscation purposes, and repeatedly written into different registers.

Each bit of the encoded bitstream is written sequentially into a large number (2000-5000) of registers mapped across the device to increase the strength of the  $\vec{H}$ -field emissions.

The layout and routing of the circuits within the FPGA fabric are also optimized to maximize interconnect inductance, thus further increasing the field strength. Both sequential logic blocks (the PRNG and serializer) are clocked by either an off-chip 50 MHz clock or an RO built using logic gates. While the off-chip clock is more stable, the on-chip RO adds inter-die frequency variations that generate chip-specific EM signal patterns. To simplify our experimental procedure, a GPIO pin is configured to output a periodic pulsed trigger to signal the start and end of each signature measurement. During normal use, this wired trigger signal will be replaced with a synchronization sequence (e.g., a periodic ‘0101...’ pattern) embedded within the emitted  $\vec{H}$ -field.

#### B. ASIC-based system

The hardware integrity verification system described in the previous section can be readily extended to ASICs by replacing the FPGA fabric with custom logic. Standard cell placement and routing on ASICs is highly customizable, thus allowing 1) the SNR of the EM emission signals to be more easily optimized during the design phase; and 2) dummy metalization patterns to be customized for EM shielding purposes.

#### C. HPS-based system

The proposed  $\vec{H}$ -field emission-based integrity verification approach was extended to software running on an embedded HPS. For convenience, an SoC containing both an FPGA and HPS was used for testing, but with only the HPS activated. Fig. 9 shows the architecture of the software authentication system. Since all functions are executed in the HPS through software instructions, this system is also applicable to more general cases where only an HPS exists (e.g., in a computer system or an embedded processor). The same mapping procedure is implemented in the HPS to find the seed value corresponding to an LFSR signature input. The list of signature and seed pairs is saved in the file system, which is booted from an external memory dedicated to the SoC. The receiver and signal detection algorithm are identical to that used in the hardware authentication system.

The PRNG and error correction functions now consist of processor instructions in a high-level programming language. The PRNG uses an LFSR software function running recursively to generate periodic multiple-bit digital signatures with values specified by the user input. The error correction mechanism uses the same Hamming code algorithm to detect and correct bit errors. A processor-based GPIO pin is configured to deliver a trigger signal for periodic measurements. During actual use, this trigger pin can be replaced by an embedded synchronization sequence, as described earlier.

Our approach utilizes processor instructions that produce strong  $\vec{H}$ -field emissions, thus maximizing the received SNR. For example, we find that writing one word into a HPS register in the SoC produces strong emissions. This observation can be used to maximize amplitude modulation of the  $\vec{H}$ -field due to the embedded signature, as summarized in Fig. 10. The modulation process for one signal period starts by reading the Hamming vector for error detection and then controlling the

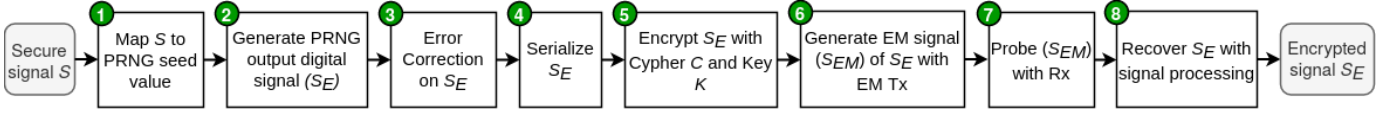


Fig. 6. Flow chart of secure information generation and detection through the proposed hardware/software integrity verification system.

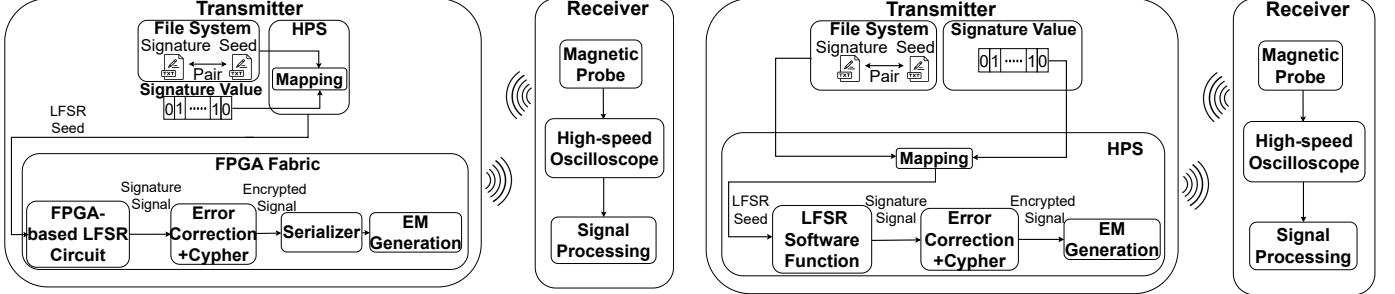


Fig. 7. System architecture of FPGA-based hardware integrity verification using  $\vec{H}$ -field emissions for secure information generation and sensing.

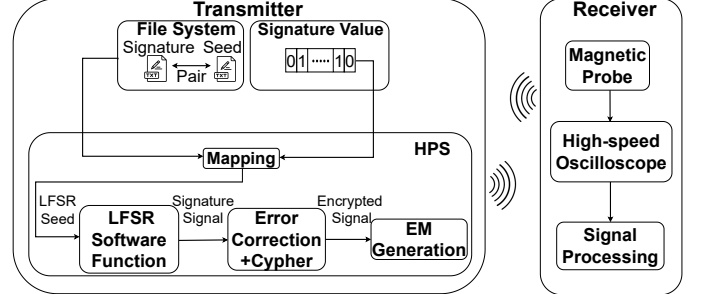


Fig. 9. System architecture of HPS-based software integrity verification using  $\vec{H}$ -field emissions for secure information generation and sensing.

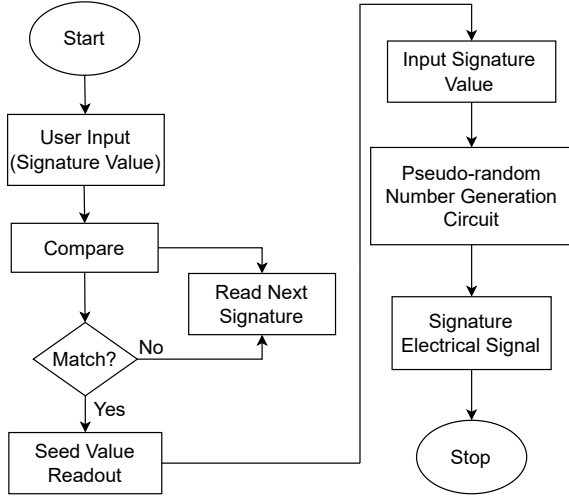


Fig. 8. Flow chart of mapping an input signature value,  $S$ , to the electrical output of a PRNG,  $S_E$  (steps 1-2 of Fig. 6).

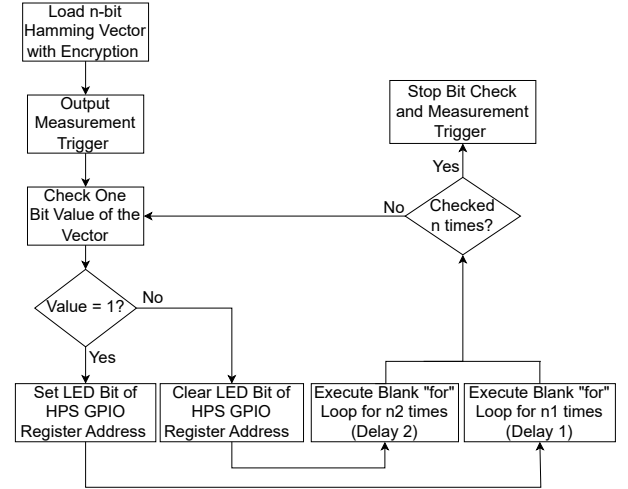


Fig. 10. Flow chart of generating an EM signal in the HPS instance for EM-based integrity verification (step 6 of Fig. 6).

processor-based GPIO pin to output a trigger signal. Next, every bit value of the signature is checked sequentially. If the bit value is equal to a logic “1”, the LED bit in the GPIO register address is set and a time delay is set by looping through  $n_1$  successive blank “for” loops; otherwise, the LED bit is cleared with another time delay set by executing  $n_2$  loops. In this way,  $\vec{H}$ -field signals for logic “1” and “0” emitted from the IC can be differentiated via their delay values. After the values of all signature bits are examined, the modulation process stops and the trigger signal is disabled. Modulation for the next signal period continues after a third time delay, which is created by executing  $n_3$  blank “for” loops.

## IV. EXPERIMENTAL RESULTS

### A. Experimental Setup

The performance of the proposed hardware and software integrity verification system was verified using experiments on both the FPGA fabric (hardware integrity, Fig. 7) and the HPS (software integrity, Fig. 9 of the Cyclone V SoC present on the Terasic DE10 Standard development board shown in Fig. 11(a)). The same experimental setup was utilized for both cases, as shown in Fig. 11(b)).

Signatures were generated by programming the SoC with both the software and hardware code designs for the transmitters shown in Fig. 7 and Fig. 9. For the hardware-based system, software functions in the HPS were interfaced to logic circuits in the FPGA via SoC-to-FPGA memory mapping, with the HPS only used to map digital signature values from user input to the corresponding LFSR seeds as shown in Fig. 8. A

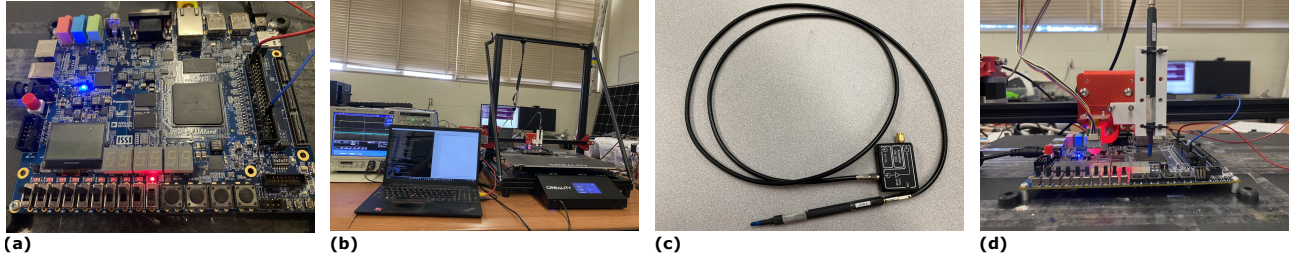


Fig. 11. (a) Terasic DE10 Standard development board containing a Cyclone V SoC. (b) Experimental setup for non-contact  $\vec{H}$ -field emission-based integrity verification of both the FPGA fabric (hardware) and the HPS (software) within the Cyclone V SoC. (c) Near-field probe and pre-amplifier used for EM measurements. (d) Zoomed-in view of the probing setup for the SoC.

terminal application was used to control the signature mapping process.

The probing system includes 1) a low-frequency (50 MHz bandwidth) near-field magnetic probe with a 30 dB low-noise pre-amplifier, shown in Fig. 11(c); 2) a high-speed (12.5 GHz) oscilloscope; and 3) a 3D printer for scanning the probe over the chip surface. Fig. 11(d) depicts a zoomed-in view of the probing setup in which the probe is suspended vertically several millimeters above the center of the chip. Both the vertical and horizontal positions of the probe can be controlled by the 3D printer, thus allowing the chip surface to be scanned to study the field pattern and also maximize SNR.

## B. Experimental Results

1) *EM signal generation results:* We first evaluated the logic utilization and memory usage of the authentication system. The FPGA-based system has  $\sim 13\%$  logic utilization and  $< 1\%$  memory usage when 5,000 output registers are used, as summarized in Table I. Next, the speed of EM signature generation is assessed. The measurement result shows that  $\sim 10$  s is required to map the signature input to an LFSR seed value, which is reasonable for most applications.

TABLE I  
LOGIC AND MEMORY UTILIZATION THE AUTHENTICATION SYSTEM.

Description	Value
Top-level Entity Name	DE10 Standard GHRD
Family	Cyclone V
Device	5CSXFC6D6F31C6
Logic Utilization (in ALMs)	5,580/41,910 (13%)
Total Registers	9661
Total Pins	138/449 (28%)

2) *EM signal sensing results:* The center of the SoC was found to emit the strongest  $\vec{H}$ -field signals. Thus, signatures were sensed after centering the probe above this point while maintaining a vertical separation of  $\sim 1.5$  mm to obtain adequate SNR. The FPGA-based system was clocked at 2.5 MHz through an on-board crystal oscillator (XO) and at  $\sim 2.4$  MHz through a RO. A 32-bit maximum-length LFSR was used as the PRNG. The signatures were serially written to 2,000-5,000 output registers to increase SNR. All measurements were averaged 100 times to further increase SNR by  $\sim 20$  dB.

The  $\vec{H}$ -field measurements of a digital signature with a value of 0x1d2f968b acquired from both the FPGA fabric

and the HPS are shown in Fig. 12. Fig. 12(a) shows the trigger signal, while Fig. 12(b) shows the time-aligned  $\vec{H}$ -field emission signal (measured as voltage) from the FPGA fabric clocked at 2.5 MHz by the on-board XO with a clock division circuit built into the FPGA fabric and 2,000 output registers. Fig. 12(c) shows the result of replacing the XO with the RO and using 5,000 output registers. Note that the amplitude of the  $\vec{H}$ -field decreases by  $> 2\times$  when the XO is replaced by the RO. This is because the clock signal generated by the RO has longer rise/fall times, resulting in less high-frequency content. In either case, the measured signal cannot be easily decoded by an attacker, thus providing security against illegitimate interception and reuse of the signature.

Fig. 12(d) shows  $\vec{H}$ -field measurement results from the HPS-based system. The measured signal can be easily demodulated to reconstruct the digital signature because the signature insertion procedure uses pulse-width modulation (PWM), i.e., the symbol duration is modulated by bits in the digital signature, with “0” being  $100\times$  longer than “1”. Note that multi-bit PWM can be used for enhanced security. For example, four symbol durations can be used to encode “00”, “11”, “01”, and “10” bit patterns in the signature. In addition, both hardware and software integrity verification systems can use the well-known 8b/10b encoding algorithm [24] to minimize distortion of the emitted waveform by ensuring DC balance.

The vertical position of the probe was adjusted (with its horizontal position remaining fixed) to measure the signal strength as a function of probe-chip distance. Figs. 13(a)-(b) show the SNR of the measured magnetic emission signals (in dB) versus probe-chip distance for the FPGA fabric-based and HPS-based systems, respectively. SNR decreases with sensing distance in both cases, as expected, but is always significantly higher for the HPS. The relationship between the SNR of the  $\vec{H}$ -field measurement and the clock frequency of the FPGA fabric-based system clocked by the XO was also evaluated for the same digital signature value (0xFFFFFFFF78). The result, shown in Fig. 13(c), suggests that SNR decreases with clock frequency due to limited probe bandwidth.

The performance of the encryption mechanism for the FPGA-based system was also assessed by using the RO in Fig. 1(b) as a clock source with an expected frequency of  $\sim 2.4$  MHz. Experiments were conducted on two identical Cyclone V SoC chips on DE10 Standard boards with 6 consecutive trials conducted on each chip and a time interval

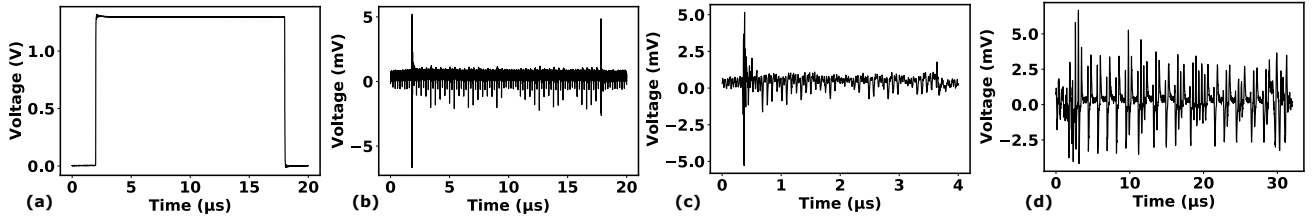


Fig. 12. (a) Terasic DE10 Standard development board containing a Cyclone V SoC. (b) Experimental setup for non-contact  $\vec{H}$ -field emission-based integrity verification of both the FPGA fabric (hardware) and the HPS (software) within the Cyclone V SoC. (c) Near-field probe and pre-amplifier used for EM measurements. (d) Zoomed-in view of the probing setup for the SoC.

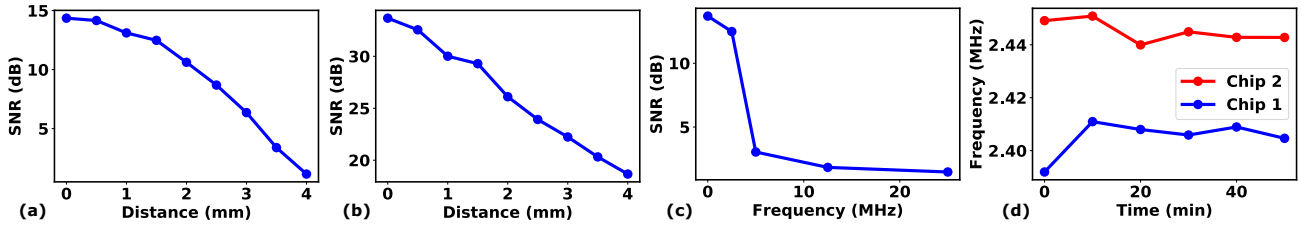


Fig. 13. SNR of the measured  $\vec{H}$ -field emissions from (a) the FPGA fabric and (b) the HPS versus sensing distance. (c) SNR (in dB) of the  $\vec{H}$ -field signals from the FPGA fabric versus clock frequency. (d) Clock frequency versus time measured by the hardware integrity verification system when a RO is used as a clock source on two identical Cyclone V SoC chips.

of 10 minutes between trials. An overall frequency difference of  $\sim 40$  kHz between the two chips is visible in Fig. 13(d). This difference is primarily due to inter-die variations in transistor properties, with temperature fluctuations expected to play only a minor role. Such clock frequency variations can contribute to improved encryption of the signature information by increasing the difficulty of recovering digital signature values (as binary sequences) from the measured  $\vec{H}$ -field emission signals.

3) *Signature detection results:* We implemented matched filtering in our receiver to enable automatic detection of the digital signature information from a measured  $\vec{H}$ -field emission signal. Signals containing 50 different signature patterns were generated for testing from the FPGA-based verification system and used as template waveforms (i.e., matched filters) for new measurements. Each template was prepared by 1) using the trigger signal to select the desired portion of the waveform; 2) removing out-of-band measurement noise via filtering; and 3) averaging over many acquisitions to minimize residual in-band noise. An example of such a rebuilt template signal is shown in Fig. 14(c). New measurements are then cross-correlated with each template, and the maximum correlator output selected to identify the embedded digital signature. Note that this procedure implements a standard correlation or maximum likelihood (ML) receiver.

Fig. 14(d) shows the maximum output of the correlation receiver when fed with a measurement of the same signature. Note the significant amount of pulse compression (about  $20\times$ ) and resulting increase in pulse amplitude and SNR. Measurements on a set of 50 different signatures show 100% detection accuracy after matched filtering. A similar matched filter receiver can also be used to automatically detect signatures from  $\vec{H}$ -field measurements acquired by the HPS-based system. However, it is less critical in that case due to the use

of PWM encoding, which simplifies the decoding procedure.

## V. CONCLUSION AND FUTURE WORK

Two non-contact software and hardware authentication systems were designed and tested in a FPGA fabric and an HPS, respectively. Both systems facilitate the process of integrity verification of either a hardware or a software target. To the best of our knowledge, they are the first systems designed to easily insert digital signatures into millions of target ICs and then securely detect them without physical contact. The applications of these systems can be extended from the chip-level to the PCB-level and even to system-level software and hardware authentication. The systems may also prove useful in transferring other secure information such as sensor measurements and biometric information privately associated with each user to an authorized party for record-keeping or scientific analysis. Apart from these novelties, our proposed methodology exploits the encryption mechanisms of stream ciphers, programmable LFSR PRNGs, and RO clock sources to make the systems robust to attacks. Also, an error correction mechanism is added to minimize bit errors during reconstruction of digital signatures from  $\vec{H}$ -field measurements. Experimental results show that the systems can generate  $> 80$  million unique pseudo-random digital signatures from either FPGA logic circuits (hardware) or HPS instructions (software) in  $\sim 10$  sec and sense the resulting  $\vec{H}$ -field emissions of emissions with high sensitivity ( $\text{SNR} > 10$  dB) using a near-field probing system. Matched filtering was used to detect a subset of 50 selected signatures with 100% accuracy. In our future work, we will implement the methods of protection against EMI attacks presented in the previous section to further improve the SNR of the  $\vec{H}$ -field emissions of the embedded pseudo-random digital signatures from either FPGA or HPS. Also, the

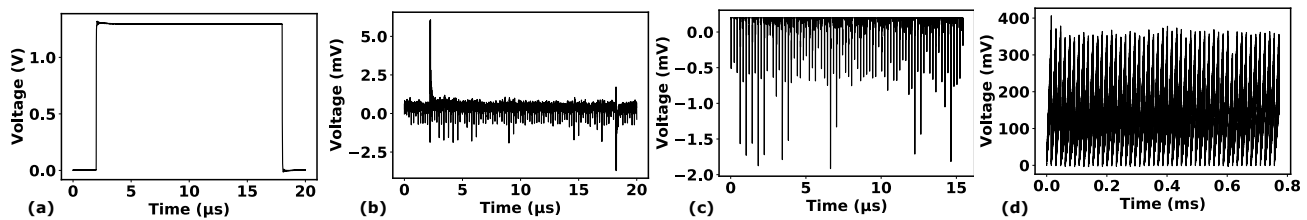


Fig. 14. (a) Trigger signal. (b) Original  $\vec{H}$ -field measurement result for a 0xFFFFFFFF8 digital signature value inserted in the FPGA-based system. (c) Rebuilt template estimated from the original measured signal. (d) Matched filter output for a new measurement of this signature using 50 template waveforms.

presented hardware integrity verification system design will be conducted and tested on ASCIs for integrity verification on custom-designed IC chips.

## REFERENCES

- [1] M. M. Ahmed, D. Hely, N. Barbot, R. Siragusa, E. Perret, M. Bernier, and F. Garet, "Radiated electromagnetic emission for integrated circuit authentication," *IEEE Microwave and Wireless Components Letters*, vol. 27, no. 11, pp. 1028–1030, 2017.
- [2] M. M. Ahmed, E. Perret, D. Hely, R. Siragusa, and N. Barbot, "Guided electromagnetic wave technique for IC authentication," *Sensors (Basel)*, vol. 20, no. 7, Apr. 2020.
- [3] P. KarthigaiKumar and K. Baskaran, "An asic implementation of a low power robust invisible watermarking processor," *Journal of Systems Architecture*, vol. 57, no. 4, pp. 404–411, 2011.
- [4] B. Min and V. Varadharajan, "Rethinking Software Component Security: Software Component Level Integrity and Cross Verification," *The Computer Journal*, vol. 59, no. 11, pp. 1735–1748, 11 2016.
- [5] M. Aydos, T. Yantk, and C. Koc, "A high-speed ecc-based wireless authentication on an arm microprocessor," in *Proceedings 16th Annual Computer Security Applications Conference (ACSAC'00)*, 2000, pp. 401–409.
- [6] J. K. Brotz, R. W. Hymel, R. J. Punnoose, T. Mannos, N. Grant, and N. Evans, "Fpga authentication methods," May 2017.
- [7] F. Bache, C. Plump, J. Wloka, T. Güneysu, and R. Drechsler, "Evaluation of (power) side-channels in cryptographic implementations," *it - Information Technology*, vol. 61, 01 2019.
- [8] M. Nagata, D. Fujimoto, N. Miura, N. Homma, Y.-i. Hayashi, and K. Sakiyama, "Protecting cryptographic integrated circuits with side-channel information," *IEICE Electronics Express*, vol. 14, no. 2, JAN 25 2017.
- [9] G. Keramidas, A. Antonopoulos, D. N. Serpanos, and S. Kaxiras, "Non deterministic caches: a simple and effective defense against side channel attacks," *Design Automation For Embedded Systems*, vol. 12, no. 3, pp. 221–230, SEP 2008.
- [10] G. T. Becker, M. Kasper, A. Moradi, and C. Paar, "Side-channel based watermarks for integrated circuits," in *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2010, pp. 30–35.
- [11] L. Lin, M. Kasper, T. Güneysu, C. Paar, and W. Bursleson, "Trojan side-channels: Lightweight hardware trojans through side-channel engineering," in *Cryptographic Hardware and Embedded Systems - CHES 2009*, C. Clavier and K. Gaj, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 382–395.
- [12] W. Liang, B. Liao, J. Long, Y. Jiang, and L. Peng, "Study on puf based secure protection for ic design," *Microprocessors and Microsystems*, vol. 45, pp. 56–66, 2016.
- [13] J. Zhang and G. Qu, "Recent attacks and defenses on fpga-based systems," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 12, no. 3, aug 2019.
- [14] G. E. Suh, D. Clarke, B. Gassend, M. v. Dijk, and S. Devadas, "Efficient memory integrity verification and encryption for secure processors," in *Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture*, ser. MICRO 36. USA: IEEE Computer Society, 2003, p. 339.
- [15] X. Li, Q. Wen, W. Li, H. Zhang, and Z. Jin, "Secure privacy-preserving biometric authentication scheme for telecare medicine information systems," *Journal of Medical Systems*, vol. 38, no. 11, NOV 2014.
- [16] A. Clark, E. Dawson, J. Fuller, J. Golic, H. Lee, W. Millan, S. Moon, and L. Simpson, "The lili-ii keystream generator," in *Information Security And Privacy*, ser. Lecture Notes In Computer Science, L. Batten and J. Seberry, Eds., vol. 2384. Deakin Univ; iCORE; Australian Comp Soc, 2002, pp. 25–39, 7th Australasian Conference on Information Security and Privacy (ACISP 2002), Melbourne, Australia, JUL 03-05, 2002.
- [17] S. Bhunia and M. Tehranipoor, *Hardware Security: A Hands-on Learning Approach*. Elsevier Science, 2018.
- [18] U. Jetzek, *Galois Fields, Linear Feedback Shift Registers and their Applications*. Carl Hanser Verlag GmbH & Company KG, 2018.
- [19] R. W. Hamming, "Error detecting and error correcting codes," *The Bell System Technical Journal*, vol. 29, no. 2, pp. 147–160, 1950.
- [20] B. Deutschmann, H. Pitsch, and G. Langer, "Near field measurements to predict the electromagnetic emission of integrated circuits," in *International Workshop on Electromagnetic Compatibility of Integrated Circuits*, 2005, pp. 27–32.
- [21] V. Pano, I. Tekin, Y. Liu, K. R. Dandekar, and B. Taskin, "Tsv-based antenna for on-chip wireless communication," *IET Microwaves, Antennas & Propagation*, vol. 14, no. 4, pp. 302–307, 2020.
- [22] P. P. Ray, "Intelligent ingestibles: Future of internet of body," *IEEE Internet Computing*, vol. 24, no. 5, pp. 19–27, SEPT 1 2020.
- [23] Z. Martinasek, V. Zeman, P. Sysel, and K. Trasy, "Near electromagnetic field measurement of microprocessor," *Przeglad Elektrotechniczny*, vol. 89, pp. 203–207, 01 2013.
- [24] I. A. Aref, N. A. Ahmed, F. Rodriguez-Salazar, and K. Elgaid, "Rtl-level modeling of an 8b/10b encoder-decoder using systemc," in *2008 5th IFIP International Conference on Wireless and Optical Communications Networks (WOCN '08)*, 2008, pp. 1–4.