Hardware Trojan Attacks in Analog/Mixed-Signal ICs via the Test Access Mechanism

Mohamed Elshamy, Giorgio Di Natale, Antonios Pavlidis, Marie-Minerve Louërat, Haralampos-G. Stratigopoulos

Hardware Trojan Attacks in Analog/Mixed-Signal ICs via the Test Access Mechanism

Mohamed Elshamy*, Giorgio Di Natale†, Antonios Pavlidis*,
Marie-Minerve Louërat*, Haralampos-G. Stratigopoulos*
*Sorbonne Université, CNRS, LIP6, Paris, France
†Univ. Grenoble Alpes, CNRS, Grenoble INP, TIMA, Grenoble, France

***Abstract*—We present a Hardware Trojan (HT) attack scenario for analog circuits. The characteristic of this HT is that it does not reside inside the victim analog circuit. Instead, it resides on an independent digital circuit on the same die where it is triggered, yet its payload is applied only to the analog circuit after being transferred via the common test infrastructure and the test interface of the analog circuit. This HT attack cannot be detected or prevented in the analog domain and it exploits the dense digital circuit to hide effectively its footprint.**

I. Introduction

Today's globalisation of the Integrated Circuit (IC) supply chain has brought many hardware security concerns. One of the major concerns is the inclusion of Hardware Trojans (HTs) into ICs that are deployed in safety-critical and mission-critical systems [1], [2]. A HT is an intentional malicious modification of the IC aiming at leaking valuable data, degrading performance, or resulting at complete malfunction, i.e. denial-of-service. A HT can be inserted into a System-on-Chip (SoC) during different phases, i.e. by an untrusted EDA tool provider, by an untrusted IP vendor, by an untrusted SoC integrator who inserts the test access mechanism, or by an untrusted foundry.

From the attacker's perspective, the goal is to design a minimum footprint HT that evades optical reverse engineering, as well as a stealthy HT that is activated in rare conditions and is hidden within the process variation margins such that it evades detection through conventional manufacturing testing. A HT design consists of two parts, namely the trigger and payload mechanisms. There is multitude of possible HTs that range from simple to very complex attack modes. The simplest HTs are combinational circuits that monitor a set of nodes to generate a trigger on the simultaneous occurrence of rare node conditions and, subsequently, once the trigger is activated, the payload is simply flipping the value of another node. More complex HTs include silicon wearout mechanisms [3], hidden side-channels [4], changing dopant polarity in active areas of transistors [5], siphoning charge from victim wires [6], etc.

From the defender's perspective, there are several paths to provide resilience against HTs depending on the phase wherein the HT is being inserted. Countermeasures can be grouped into pre-silicon and post-silicon HT detection and design-for-trust (DfTr) techniques. Pre-silicon HT detection techniques include functional validation and formal verification. Post-silicon HT detection techniques include optical reverse engineering, functional testing that aims at exposing the HT by applying test vectors, and statistical fingerprinting that aims at exposing the HT by its effect on parametric measurements, i.e. delay, power, temperature, etc. DfTr techniques include facilitating HT detection, i.e. based on run-time monitoring or on-chip sensors, and preventing HT insertion. Prevention can be achieved by obfuscation, locking, camouflaging or split manufacturing, which all aim to obscure the IC functionality so as to make it difficult for the attacker to insert the HT. Finally, there exist HT-specific defenses to address HTs that at a first glance seemed lethal and untraceable by known defenses. For example, the dopant-level HTs proposed in [5] were shown to be visible by Scanning Electron Microscopy (SEM) in [7].

While the hardware security problem for digital ICs has been studied extensively during the past decade, research for analog ICs is lagging seriously behind [8]–[10].

HT insertion in analog ICs has been demonstrated so far in the context of cryptographic wireless ICs aiming at leaking sensitive information, i.e. cipher keys. It has been demonstrated how the key can be encoded into minute differences in amplitudes or frequencies of the transmitted signal [11], [12] or into an unauthorized transmission signal that is hidden within the legitimate signal [13]. In both cases, the IC passes all conventional specification tests and the transmission signal still obeys to the transmission specifications and is within the margins allowed because of process variations. Therefore, the inconspicuous receiver cannot interpret the minute change in the transmitted signal as malicious. However, the attacker knowing the HT payload mechanism can *listen to* the channel and recover the key. It has been demonstrated that this type of HTs can be detected by statistical fingerprinting [11], [12], careful analysis of the transmitted signal spectrum [13], or channel estimation [14]. Another interesting direction for HT design is to exploit the fact that an analog IC may have undesired states or operation modes [15]. In this case, the HT attack consists of bringing the analog IC into one of these states to cause undesirable operation.

In general, designing HTs for analog ICs is very challenging since all criteria that make up an effective HT are difficult to meet. First, it is difficult to design stealthy HT since analog signal paths are typically very sensitive and a HT circuitry tapping into them is likely to result in some non-negligible performance degradation. Second, it is difficult to design small footprint HTs that will evade optical reverse engineering since analog designs comprise few components or can be clearly divided into sub-blocks or stages each comprising few components. Third, on any analog IC we can extract several information-rich measurements, such that it is unlikely not to be able to find a measurement subspace wherein HT-infected and HT-free instances are clearly distinguished [16], [17]. Similar to digital

ICs, HT prevention could be achieved via locking, obfuscation, and split manufacturing; however, few such approaches exist so far for analog ICs. For example, locking mixed-signal ICs via logic locking of their digital section can prevent effective HT insertion inside their digital section [18].

Exploring HT attack models is critical to understand to what extent this inherent resiliency of analog IC to HT is well grounded. In this paper, we propose a new HT attack for analog ICs in the context of a SoC with the key property that the HT does not reside inside the analog Intellectual Property (IP) block and is completely transparent to it. Instead, with the aim to increase the hardness of detection and prevention, the HT is hidden and triggered inside a digital IP block and the test infrastructure of the SoC, while its payload is transferred to the analog IP block by exploiting the common test access mechanism (TAM) that links all IP blocks in a SoC in a daisy-network, and is applied to the analog IP block via the built-in self-test (BIST) interface.

The rest of the paper is structured as follows. In Section II, we discuss BIST and TAM for analog and mixed-signal ICs. In Section III, we present the HT attack scenario. In Section IV, we demonstrate this HT attack in a case study. In Section V, we briefly discuss possible countermeasures. Section VI concludes the paper.

II. Analog and Mixed-Signal BIST and TAM

ICs deployed in safety-critical and mission-critical applications demand high dependability features. Equipping ICs with BIST capabilities, in order to perform in-field tests, is a first step towards dependability. The same BIST can also be used in post-manufacturing testing so as to reduce test costs and improve outgoing quality, i.e. towards low defective parts per million (dppm). In general, the BIST infrastructure consists of test instruments, including test stimuli generators, measurement extraction units, and output post-processing units to arrive at a pass or fail test response.

In the context of a SoC, each embedded IP comes with its own test instruments. The latest standard for test instrument controllability and observability is the IEEE Std. 1687 [19], which is driven by the needs for test portability and re-use. It deals with the great number of test instruments and connects them serially via programmable segment insertion bits (SIBs) to a reconfigurable scan network (RSN) between the scan in (SI) and scan out (SO) ports. When the SIB of a test instrument is opened, its test data register (TDR) becomes part of the RSN such that it is accessed from the SI port and its output is streamed to the SO port. IEEE Std. 1687 was developed with digital ICs in mind. The standard for analog test access is the IEEE Std. 1149.4 [20] and dates from the 1990s. It proposes a test bus paradigm that is still used today, but it requires a minimum of two additional test pins which is too costly and often prohibitive as many designs are pin-limited. To this end, nowadays there is an IEEE working group extending IEEE Std. 1687 [21] to include properties demanded by analog ICs, such as periodic sampling. The envisioned test access standard will be compatible for both analog and digital IPs connecting them onto a common test infrastructure.

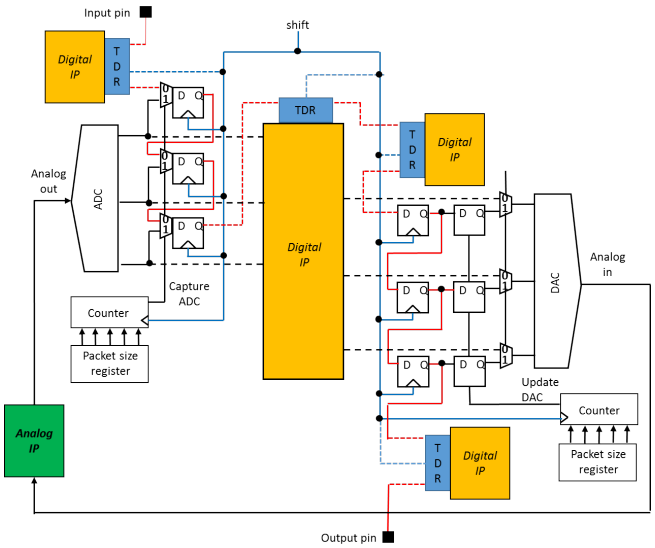

Fig. 1. Scan access including analog IPs (adapted from [22]).

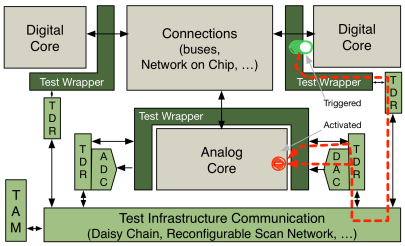

Fig. 2. Hardware Trojan scenario exploiting the SoC test infrastructure.

The principle for connecting analog IPs to the common test infrastructure is proposed in [22]. An example is shown in Fig. 1 including: (a) an Analog-to-Digital Converter (ADC) and its TDR connecting the analog output of an analog IP (or the analog output of a test instrument inside it) to the scan network; (b) a Digital-to-Analog Converter (DAC) and its TDR connecting the analog input of an analog IP (or the analog input of a test instrument inside it) to the scan network; (c) a counter and a packet size register for each ADC and DAC that set the periodicity of the TDR update and capture operations; (d) any number of intervening or appended TDRs that connect digital IPs to the scan network. The figure shows for simplicity 3-bit converters and also shows that the output of the analog IP may be digitized and driven into a digital IP or the output of a digital IP may be driving an analog input via the DAC.

III. Hardware Trojan Attack Scenario

It is well-known that the test infrastructure can be the vehicle for an unauthorized user for launching several types of scan attacks aiming at stealing secret keys, performing reverse-engineering and device cloning, performing memory dumping, and modifying memory values to attain privilege escalation [23], [24]. HT attacks are also possible by third-party malicious IPs connecting to the scan network, i.e. with the purpose of sniffing confidential data. For a more detailed description of security threats and countermeasures in IEEE test standards, the interested reader is referred to [25].

In this paper, we propose a novel HT attack scenario for analog IPs that exploits the fact that analog and digital IPs

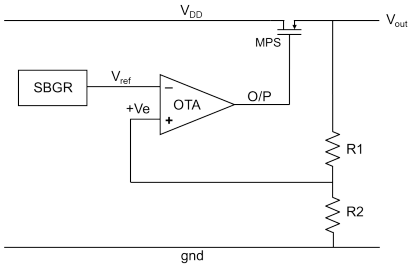

Fig. 3. Block-level schematic of the LDO.

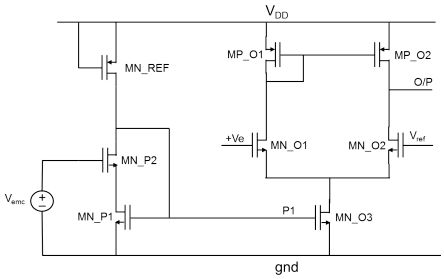

Fig. 4. Schematic of the error amplifier within the LDO implemented with an OTA.

coexist in the SoC and are linked together via the shared common test infrastructure. As illustrated in Fig. 2, the HT is not hidden inside the analog IP itself, thus neither detection nor prevention are possible in the analog domain. The HT is triggered instead in a digital IP and taps into the scan network so as to transfer its payload to the analog IP. More specifically, the HT payload consists in enabling the scan network, switching the analog IP into test mode during mission mode, and driving a malicious BIST signal at the input of the analog IP. All the IPs apart from the targeted analog IP can be bypassed thanks to the programmability features of the RSN. The malicious BIST signal can be designed to result in performance degradation or denial-of-service for the analog IP. In turn, if the analog IP controls other digital IPs, then the operation of the entire SoC can be jeopardized. In our threat model, the attacker can be the third-party SoC integrator who inserts the scan network, or a third-party specialized test infrastructure IP provider. In fact, nowadays, the design of test infrastructures has become such a complex task that even this task can be outsourced to third-parties, thus increasing the possibility for an untrusted provider to insert HTs.

This HT attack scenario is general and could be implemented using various types of HT triggering mechanisms, various ways to tap into the scan network, and various malicious BIST patterns infecting the victim analog IC in various ways.

In Section IV, we present an example of how this scenario might play out in a generic SoC. The HT is triggered inside the processor and its payload infects a low-dropout regulator (LDO). Although the LDO is the direct victim of the HT, since the LDO supplies one or more digital IPs inside the SoC, then the HT infects implicitly digital IPs too.

IV. Mixed-signal SoC case study

A. Low-dropout regulator design

The LDO is one of the most popular power management systems to supply the sub-blocks of a SoC. We designed an

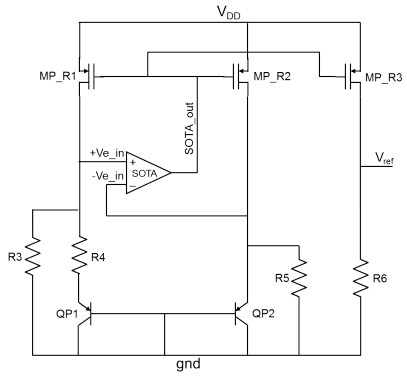

Fig. 5. Schematic of SBGR generator.

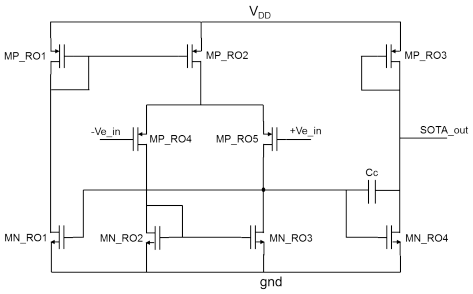

Fig. 6. Schematic of SOTA.

LDO in the 65nm technology by ST Microelectronics using the free open-source OCEANE tool [26]. Its block-level schematic is shown in Fig. 3. It consists of a sub-band gap reference voltage generator (SBGR), an error amplifier implemented with an operational transconductance amplifier (OTA), a power p-MOS transistor, and a feedback resistor network. The error amplifier monitors a fraction V_e of the LDO output voltage V_{out} through the resistor feedback network and compares it with the output voltage V_{ref} of the SBGR. If V_e is higher (lower) than V_{ref}, then the error amplifier drives the gate of the power transistor to decrease (increase) its output voltage so as to maintain a constant V_{out}. Figs. 4 and 5 show the schematics of the OTA and SBGR. Fig. 6 shows the schematic of the self-biased operational transconductance amplifier (SOTA) inside the SBGR.

The green curves in Figs. 7-9 show the nominal LDO performance in the HT-free scenario. Specifically, Fig. 7 shows the LDO output variation as a function of power supply voltage variations at 27^{o}C. As it can be seen, V_{out} shows a 33.4mV variation when V_{dd} varies from 1.4V to 3V. Fig. 8 shows the LDO output dependence on temperature variations for a V_{dd} equal to 1.5V. As it can be seen, V_{out} shows a 10mV variation when temperature varies from -55^{o}C to 125^{o}C. Fig. 9 shows the transient response of the LDO for a variation of load current from 50mA to 0mA and then from 0mA to 50mA, which corresponds to removing the load and then adding it back. The maximum overshoot is 44.9mV and settles after 875ns, while the maximum undershoot is 53.2mV and settles after 800ns.

B. BIST design

We use a generic defect-oriented BIST concept for low-frequency analog ICs proposed in [27]. The BIST principle is based on topology modification (or re-configuration) enabled

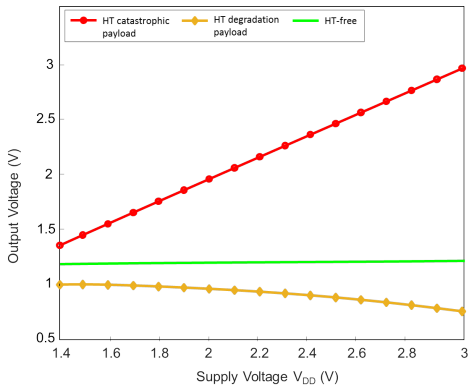

Fig. 7. LDO output variation as a function of power supply variation.

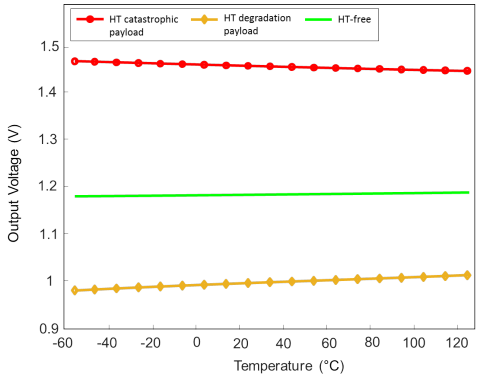

Fig. 8. LDO output variation as a function of temperature variation.

by the addition of pull-down (PD) and pull-up (PU) transistors. A PD transistor connects a circuit node to ground, while a PU transistor connects a circuit node to the power supply. PD and PU transistors are activated by applying a logic 1 and 0 at their gates, respectively. If N PD and PU transistors are added, then the circuit can be configured into 2^N topologies, including the original one where all PD and PU transistors are deactivated. The underlying principle is that by these re-configurations we are able to expose the presence of additional defects that are undetectable in the original topology.

A DC test is used for the LDO. In particular, the LDO is self-activated and its output is used as the test output. In the defect-free case, for each test configuration, a different nominal test output value $V_{\text{test,j}}$ may be observed, where j denotes the configuration number. To account for process variations and avoid yield loss, we consider a tolerance window $\pm k * V_{\text{test,j}}$, $k > 0$. For the purpose of our experiment, we set $k = 0.1$.

The defect simulation is performed at transistor-level and in an automated workflow using the Tessent®DefectSim tool by Mentor®, A Siemens Business [28]. We cycle through all configurations and for each configuration defects are injected one by one. If $V_{\text{test,j}}$ is outside the tolerance window then the defect is deemed detectable by the test configuration.

We rely on a standard defect model. In particular, for MOS transistors we use only gate open and drain-to-source short defects. Similarly, for Bipolar transistors, we consider base open and collector-emitter short defects. We consider the default short resistance of 10 ohms. Regarding opens, a weak pull-up or pull-down is assigned to each open defect to account for

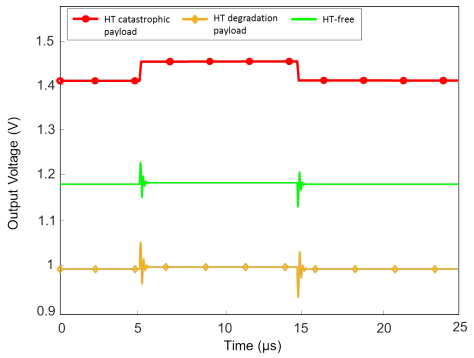

Fig. 9. Transient response of the LDO for a variation of load current.

the facts that an ideal open does not exist and, besides, it cannot be handled by a SPICE simulator [28]. For passive elements, i.e. resistors and capacitors, we consider $\pm 50\%$ variations. In total, the defect model contains 60 defects. Furthermore, any of the N added PU or PD transistors could also contain defects, which increases the number of defects by $2N$. We consider the absolute defect coverage defined as the percentage of detected defects.

A defect coverage of 80% is reached using only the original topology. We applied the BIST idea considering that in a given re-configuration only one PU or PD transistor can be enabled. The LDO has 14 nodes in total, thus the number of possible re-configurations is 28. We performed an exhaustive search and we identified 3 nodes where PD and PU transistors can be added to result in a defect coverage of 100%. The complete LDO schematic with the embedded BIST circuitry is shown in Fig. 10. One PD and one PU transistor, labelled by B1 and B2, respectively, are used inside the error amplifier, and one PD transistor, labeled by B3, is used inside the SBGR. The BIST is deactivated with the pattern [B1,B2,B3]=010, while the patterns for enabling the three test configurations are [B1,B2,B3]=110, [B1,B2,B3]=000, and [B1,B2,B3]=011.

C. Hardware Trojan payload design

An interesting aspect of this BIST is that the BIST infrastructure inside the analog IP has a digital word input and can be connected directly to the scan network without using a DAC. Another interesting aspect specific to the LDO is that the LDO is self-driven without needing to specify a BIST analog input.

The HT payload consists in applying a malicious BIST pattern during normal operation. We identified two such BIST patterns that result in degradation of the LDO performance and to complete malfunction, respectively. In turn, the HT can affect indirectly all digital IPs inside the SoC that are supplied by the LDO, thus resulting in degradation or complete malfunction of a large part or even the entire SoC.

In particular, applying the BIST pattern [B1,B2,B3]=110 results in shifting the LDO output by about 15% and also results in small variation of the LDO output for temperature and V_{dd} variations, as shown by the orange curves in Figs. 7-9. In more detail, enabling B1 results in zero gate voltage for transistors MP_O1 and MP_O2 which increases the current flowing through them. However, the sum of the currents stays fixed since it equals the current flowing through MN_O3 which

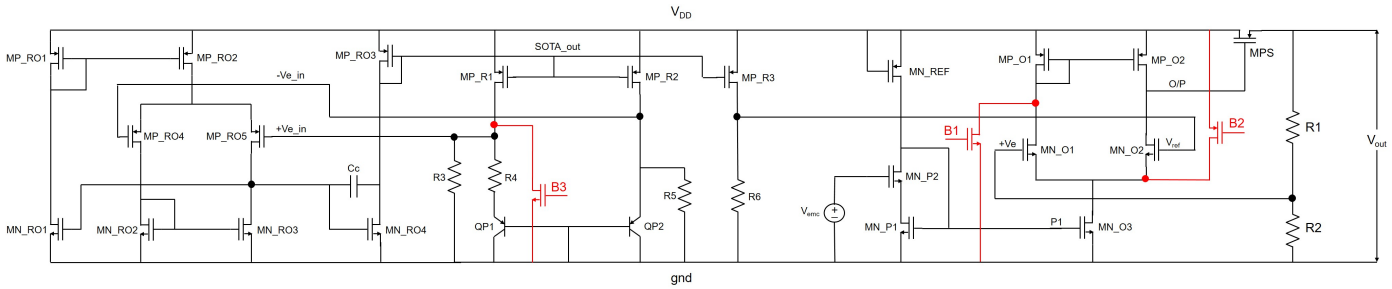

Fig. 10. LDO with BIST. The added PD and PU transistors are shown in red color.

is fixed. As the voltages of all terminals of MP_O1 are fixed, it turns out that the current through MP_O2 reduces, which is enabled by the increase of the drain voltage of MP_O2. This voltage drives the gate of the power p-MOS transistor MPS and, thereby, the current that flows through MPS reduces, which reduces the LDO output. In turn, this reduces the voltage on the $+V_e$ terminal which points to reduction of the source voltage of MN_O1 since the current flowing through MP_O1 is fixed. This feedback effect reduces the drain voltage of MN_O2 which is the gate voltage of MPS. In the end, as it can be seen from Figs. 7-9, the LDO output settles at a slightly lower value of around 1V.

Applying the BIST pattern [B1,B2,B3]=011 results in a catastrophic effect in the operation of the LDO, as shown by the red curves in Figs. 7-9. In more detail, setting B3=1 connects the $+V_{e\_in}$ terminal of the SOTA to ground. The result is that V_{ref} follows V_{dd} instead of being stabilized at 0.7V. Since the output of the LDO follows V_{ref}, it shows a linear relationship with V_{dd} acting like a non-stabilized power supply. In addition, once the load is removed the response overshoots and never settles back unless the load is added again.

D. Hardware Trojan triggering mechanism

For the HT triggering mechanism we can use any of the benchmarks in Trust-Hub [29]. Herein, we propose a HT that can be activated by the attacker in a deterministic way, by sending predefined data via the communication channels existing in the SoC. We assume that the attacker does not know which software is running on the processor. However, the target applications for which the SoC will be used are known, i.e. mobile phones. Therefore, the attacker can suppose that certain software will be present in the system when deployed, i.e. a JPEG decoder to show received images.

Fig. 11 illustrates the proposed HT triggering mechanism and the transferring of its payload via the scan network into the LDO. The triggering mechanism observes the data bus of the processor by tapping into the wrapper boundary register cells. The HT is triggered when three predefined 32-bit values, denoted by K1, K2, and K3, are observed consecutively on the bus. The idea is to allow the attacker to send this *activation sequence* to the SoC, i.e. via an e-mail with a particular JPEG image which contains three pixels of the colors encoded as K1, K2, and K3, respectively. The software running on the target processor will load the three values, i.e. for converting the JPEG image into its actual bitmap to be shown on the screen. Therefore, if the attacker sends such an image, the HT will observe the activation sequence and it will be triggered. The number of values in the activation sequence has been selected equal to three so to decrease at an extremely low value the probability of unwanted triggering of the HT, and also to render a minimum footprint HT.

The HT triggering mechanism is controlled by a Finite State Machine (FSM) whose implementation details are shown in Fig. 12. The first three states are meant to recognize the activation sequence on the data bus. Since the data bus is also used to fetch the instructions, the HT considers that K1, K2, K3 are in a sequence even if a certain number of other values are interleaved on the bus within a limited time-frame, i.e. 16 clock cycles. Once the sequence is recognized, the FSM forces the signals for shifting the malicious BIST pattern via the scan chain for a number of clock cycles required to reach the LDO. It then activates the update signal to switch the LDO into test mode and force the malicious BIST pattern.

V. Discussion on Countermeasures

The HT triggering mechanism resides outside the analog IP, and the payload is naturally applied to the analog IP via its BIST input, thus the HT mechanism is totally transparent to the analog IP. Countermeasures against the HT can be implemented only in the digital domain.

Countermeasures from the digital IP designer's perspective with the aim to detect the HT presence or prevent its insertion were discussed in the introduction and are generally applicable, although their actual effectiveness is questionable for a HT triggering mechanism that is both stealthy and with a tiny footprint.

On the other hand, prevention methods have been proposed to improve the trust in the test infrastructures. There are several techniques for restricting access to the test infrastructure, i.e. password-based authentication, obfuscation of the RSN structure, locking the SIBs, etc. [25]. In essence, all these methods utilize a key that must be first applied so as to enable the test infrastructure. However, the key is known to the SoC integrator or the test infrastructure IP provider who inserts the HT and, thereby, these methods can only be used against external threats. Another type of prevention method is to study dependencies among cores in a RSN so as to detect possible security and trust violations and, thereafter, build a new secure network [30]. Again, this cannot prevent HT insertion given our threat model. A possible solution would be to encrypt the scan path within the digital IP such that the intent HT trigger signal from the digital IP gets modified when it is shifted through the scan path, thus impairing the intent payload.

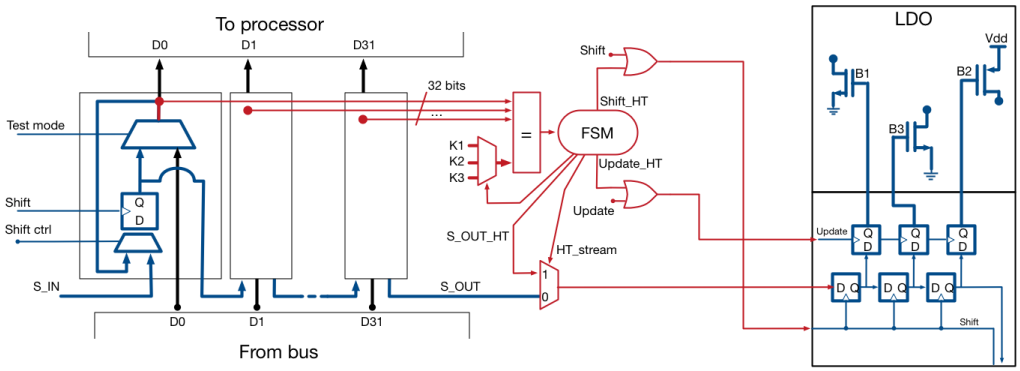

Fig. 11. Implementation details of HT mechanism. The original test infrastructure elements are shown in blue, while the added HT mechanism is shown in red.

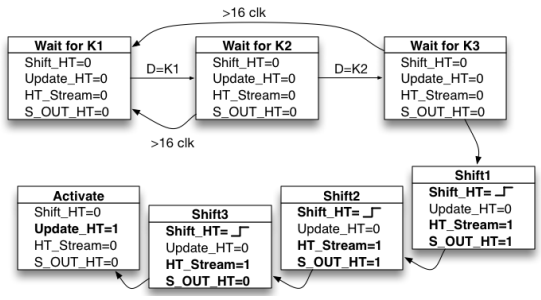

Fig. 12. Implementation details of FSM.

VI. Conclusions

We proposed a HT attack for analog ICs in the context of SoCs. The HT is triggered in a digital IP and generates a malicious bitstream that is shifted through the common scan chain and activates the BIST inside the analog IP, thus forcing an incorrect functionality that can range from a performance penalty to complete malfunction. We demonstrated this scenario for a generic SoC where the HT is activated inside the processor and drives its payload via the scan chain to an LDO, thus indirectly affecting the power supply of digital IPs inside the SoC. Future work will focus on studying countermeasures.

Acknowledgments

This work has been carried out in the framework of the ANR STEALTH project with N° ANR-17-CE24-0022-01.

References

[1] R. Karri et al., "Trustworthy hardware: Identifying and classifying hardware trojans," *Computer*, vol. 43, no. 10, pp. 39–46, 2010.

[2] S. Bhunia et al., "Hardware trojan attacks: Threat analysis and countermeasures," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1229–1247, 2014.

[3] Y. Shiyanovskii et al., "Process reliability based trojans through NBTI and HCI effects," in *Proc. NASA/ESA Conference on Adaptive Hardware and Systems*, 2010, pp. 215–222.

[4] L. Lin et al., "Trojan side-channels: Lightweight hardware trojans through side-channel engineering," in *Cryptographic Hardware and Embedded Systems*, 2009, pp. 382–395.

[5] G. T. Becker et al., "Stealthy dopant-level hardware trojans: Extended version," *Journal of Cryptographic Engineering*, pp. 19–31, 2014.

[6] K. Yang et al., "A2: analog malicious hardware," in *Proc. IEEE Symposium on Security and Privacy*, 2016, pp. 18–37.

[7] T. Sugawara et al., "Reversing stealthy dopant-level circuits," *Journal of Cryptographic Engineering*, vol. 5, no. 2, pp. 85–94, 2015.

[8] I. Polian, "Security Aspects of Analog and Mixed-Signal Circuits," in *Proc. IEEE International Mixed-Signal Testing Workshop*, 2016.

[9] A. Antonopoulos et al., "Trusted analog/mixed- signal/RF ICs: A survey and a perspective," *IEEE Design & Test*, vol. 34, no. 6, pp. 63–76, 2017.

[10] M. M. Alam et al., "Challenges and Opportunities in Analog and Mixed Signal (AMS) Integrated Circuit (IC) Security," *Journal of Hardware and Systems Security*, vol. 2, no. 1, pp. 15–32, 2018.

[11] Y. Jin and Y. Makris, "Hardware trojans in wireless cryptographic ICs," *IEEE Design & Test of Computers*, pp. 26–35, 2010.

[12] Y. Liu et al., "Silicon demonstration of hardware trojan design and detection in wireless cryptographic ICs," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 4, pp. 1506–1519, 2017.

[13] S. Chang et al., "Detection mechanisms for unauthorized wireless transmissions," *ACM Transactions on Design Automation of Electronic Systems*, pp. 70:1–70:21, 2018.

[14] K. S. Subramani et al., "ACE: Adaptive channel estimation for detecting analog/RF trojans in WLAN transceivers," in *Proc. IEEE/ACM International Conference on Computer-Aided Design*, 2017, pp. 722–727.

[15] Q. Wang et al., "Transparent side channel trigger mechanism on analog circuits with PAAST hardware trojans," in *Proc. IEEE International Symposium on Circuits and Systems*, 2018.

[16] Y. Liu et al., "Hardware trojan detection through golden chip-free statistical side-channel fingerprinting," in *Proc. Design Automation Conference*, 2014.

[17] F. Karabacak et al., "RF circuit authentication for detection of process trojans," in *Proc. IEEE VLSI Test Symposium*, 2018.

[18] J. Leonhard et al., "Mixlock: Securing mixed-signal circuits via logic locking," in *Proc. Design, Automation & Test in Europe Conference*, 2019.

[19] "IEEE standard for access and control of instrumentation embedded within a semiconductor device," *IEEE Std 1687-2014*, 2014.

[20] "IEEE standard for a mixed-signal test bus," *IEEE Std 1149.4-2010 (Revision of IEEE Std 1149.4-1999)*, 2011.

[21] "IEEE standard for describing analog test access and control, https://standards.ieee.org/project/1687_2.html," *IEEE Std P1687.2*.

[22] S. Sunter et al., "Streaming access to ADCs and DACs for mixed-signal ATPG," *IEEE Design & Test*, vol. 33, no. 6, pp. 38–45, 2016.

[23] K. Rosenfeld and R. Karri, "Attacks and defenses for JTAG," *IEEE Design & Test of Computers*, vol. 27, no. 1, pp. 36–47, 2010.

[24] A. Das et al., "Secure JTAG implementation using Schnorr protocol," *Journal of Electronic Testing*, vol. 29, no. 2, pp. 193–209, 2013.

[25] E. Valea et al., "A survey on security threats and countermeasures in IEEE test standards," *IEEE Design & Test*, pp. 95–116, 2019.

[26] J. Porte, "Outil pour la conception et l'enseignement d'électronique analogique (OCEANE)," https://www-soc.lip6.fr/equipe-cian/logiciels/oceane/, Online.

[27] A. Coyette et al., "Automatic generation of test infrastructures for analog integrated circuits by controllability and observability co-optimization," *Integration, the VLSI Journal*, vol. 55, pp. 393–400, 2016.

[28] S. Sunter et al., "Using mixed-signal defect simulation to close the loop between design and test," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 12, pp. 2313–2322, 2016.

[29] B. Shakya et al., "Benchmarking of hardware trojans and maliciously affected circuits," *Journal of Hardware and Systems Security*, vol. 1, no. 1, pp. 85–102, 2017.

[30] P. Raiola et al., "On secure data flow in reconfigurable scan networks," in *Proc. Design, Automation & Test in Europe Conference*, 2019.