MPA: Model-assisted PCB Attestation via Board-level RO and Temperature Compensation

Zimu Guo, Xiaolin Xu, Mark M. Tehranipoor and Domenic Forte
ECE Department, University of Florida
Email: zimuguo@ufl.edu; {xiaolinxu,tehranipoor,dforte}@ece.ufl.edu

***Abstract*—Printed circuit boards (PCBs) are the most common chip carriers of modern electrical systems. Compared with integrated circuits, it is more straightforward to apply attacks such as reverse engineering, unauthorized access and tampering on PCBs. Among these attacks, tampering is a serious threat to modern electronic devices such as smart meters, digital media players, and video game consoles. By tampering a PCB, an attacker can eavesdrop and/or manipulate critical inter-component communications. In the case of home electronics, this is often done to bypass the device's copyright protection mechanisms. In this paper, a model-assisted attestation framework, MPA, is presented to mitigate these concerns. This framework exploits JTAG ports to create board-level ring oscillators which monitor critical PCB traces. The impact of temperature on the framework is also analyzed, and modeling is used to compensate for this effect. Besides the methodology, we also present the detailed hardware implementation and enrollment-detection flow in this paper. Finally, the performance of our proposed framework is validated with commercial SoCs and custom PCBs. A high detection accuracy (> 99.7%) is achieved across various testing corners.**

I. Introduction

A printed circuit board (PCB) provides mechanical support and electrical connections for electronic components such as integrated circuits (ICs), capacitors, or resistors. As the major component of electronic systems, PCBs are vulnerable to various attacks [1]. Several security issues of PCBs are summarized in Figure 1: reverse engineering, recycling [2] and post-manufacturing alternation [3]. Reverse engineering a PCB enables an attacker to copy the design information and produce copies for their profit. Comparing with the ICs, reverse engineering a PCB is easier and more affordable [4]. Attackers can easily hire on-line agents to accomplish this task [5]. PCB recycling refers to reselling the out-of-spec PCBs, which are harvested from the discarded systems or electronic trash, to the consumers as new [6]. These PCBs may exhibit lower reliability and shorter mean-time-to-fail (MTTF). Alternation of a PCB provides the following capabilities: extracting secret keys by hacking test/debug interface, such as joint test access group (JTAG) [7], bypassing the copyright verification/carrier restriction of various consumer electronics (e.g., Microsoft Xbox and Apple iPhone), etc. [8].

Prior work has explored how to prevent attackers from cloning a PCB. An obfuscation based protection framework was proposed to eliminate both the destructive and non-destructive reverse engineering attacks in [9]. This framework permuted the board-level inter-chip connections with a key-controlled obfuscation block. A locking block is embedded on the PCB to realize such permutation. Besides adding an extra component to conceal these connections, blocking materials can be inserted into the middle layers of the PCB. This approach is developed against the non-destructive reverse engineering [10]. These blocking materials can create noise and block transmission X-ray by absorption. Therefore, a clear internal structure of the PCB cannot be obtained by X-ray tomography. Aside from the above prevention-oriented approaches, researchers also devote significant effort on detection-oriented methods against the recycling/counterfeit PCBs. For example, the JTAG based infrastructure has been developed to generate high-quality PCB signatures [11]. This approach combines the unique IDs generated from each IC's JTAG infrastructure. Researchers in [12] proposed a robust counterfeit PCB detection technique by exploiting the intrinsic trace impedance variations. The digitalized trace impedance measurements are utilized as the PCB signature.

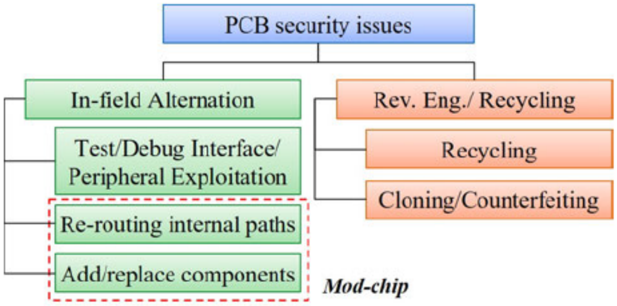


Fig. 1. Taxonomy of various security issues in a PCB.

Besides the PCB reverse engineering/recycling issues, the alternation of PCBs is growing as another significant threat. Specifically, re-routing internal paths and adding components on PCBs make it possible to bypass the copyright verification [13], break the carrier restriction [14], and run 3rd-party systems [15]. The primary method of achieving these goals is to install a modification chip (**mod-chip**) into the system. A mod-chip is a small electronic device used to alter or disable artificial restrictions of computers or entertainment devices. Mod-chips are mainly used in videogame consoles, DVD, and Blue-ray players. Not only the tampering appears during these in-field usage, the devices (e.g/, the Cisco routers) can be tampered during the distributions [16]. An example of a mod-chip implementation is shown in Figure 2. In this figure, the mod-chip manipulates the communications between the system's BIOS and DSP by attaching their pins. Mod-chips are low cost and can be easily obtained from various sources such as Ebay. Besides the availability, the installation of a mod-chip is straightforward even for general users with the instructions provided by the mod-chip vendor.

To eliminate tampering attacks, we propose a model-assisted PCB attestation framework (MPA). MPA monitors the changes in trace impedance introduced by tampering with board-level ring oscillators (ROs). In the implementation of MPA, a testing mode is executed for tampering detection by configuring the JTAG infrastructure as HIGHTZ mode. In this mode, the RO frequencies are collected to determine tampering detection based on a threshold value. Since the RO oscillation frequencies are also dependent on temperature, MPA uses a temperature-dependent model to compute this threshold. The normal operation will be suspended if any tampering event is detected. The major contributions of MPA are summarized as follows:

1) A framework (MPA) is proposed for detecting PCB tampering attacks such as mod-chips, jumper wires, and disabled traces. The hardware implementation of MPA incurs reasonably low overhead to the original design.
2) A comprehensive flow for constructing a temperature-dependent model is provided. This model is exploited to compensate for the side effects induced by the temperature fluctuations and improve the detection accuracy.
3) Test PCBs are fabricated and a platform is constructed to validate MPA. The results indicate a high detection accuracy ($> 99.7\%$) under various temperature conditions.

The remainder of this paper is organized as follows. The enrollment and detection flows of our framework are described in Section III. In Section II, existing solutions and their limitations are discussed. Section IV provides the schematics of the hardware implementation. Finally, the detection accuracies of MPA are evaluated and concluded in Sections V and VI.

II. Related Work

Physical countermeasures have been seen for either detecting or preventing board-level tampering. PCBs can be protected in a hard steel case with opening switches installed. These switches will be activated to suspend the system if the case is opened. Temperature/vibration sensors are used to monitor abnormal tampering activities and abrupt movements. Active tamper mesh is also employed which continuously sends and receives a random sequence of data along the PCB traces and monitors any interruptions or short circuit along the path. Texas instruments (TI) released an application report which provides a system-level tamper protection framework [17]. This framework involves identifying security assets in the system and defining a trust line boundary around it, and any attempt to cross the trust line.

Besides the above physical countermeasures, a path-resistance based protection method has been proposed in [1]. In this method, the physical tampering detection and protection circuit are implemented. This circuit consists of a high-resolution analog-to-digital converter (ADC) to measure the resistance of the critical paths. To achieve this measurement, a constant current source is required to connect with these critical paths as the reference.

Though significant works have been devoted to addressing the tampering issues, the solutions above are limited by their drawbacks in certain applications. The temperature/vibration sensor based protection scheme will be compromised if these sensors are removed. The vibration sensors are not suitable for the applications with expected movements since these sensors are likely to raise false alerts. Additionally, the systems, which are protected by opening switches, should be handled carefully to avoid the switch activation. The PCB tamper mesh for active tamper monitoring is typically an expensive measure at system-level. Such limitations make these solutions inappropriate for consumer electronics such as cell phones and video game consoles. Moreover, involving the current source and ADC requires custom analog modifications or is not even applicable. In the next section, MPA is introduced with detailed design modification.

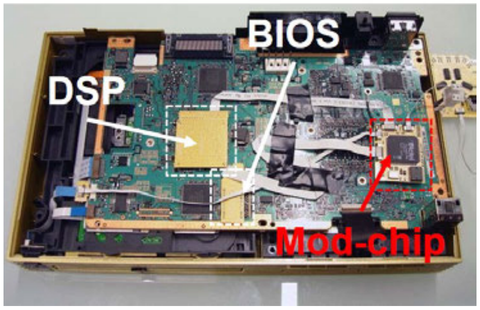


Fig. 2. Mod-chip installation on a Xbox 360 videogame console [13].

III. Methodology

MPA detects tampering by measuring the frequencies of board-level ROs. Additional impedance which is introduced by the tampering, may decrease the oscillation frequencies of ROs. In order to catch these differences, a **detector** is engaged in MPA. This detector can be implemented in either the FPGA, CPLD, or system-on-chip (SoC). In this paper, we will refer to the SoC as the carrier of the detector. The hardware implementation of the detector can be found in Section IV-B.

In MPA, the detector uses two types of ROs: **reference RO** and **detection RO**. Reference RO provides the reference for the current in-chip temperature condition. The reference is used to compute an RO frequency threshold referred to as the **genuine frequency** (i.e., expected frequency from the non-tampered PCB). The detection RO covers the board-level critical paths and monitors the impendence changes on these paths. By comparing this frequency of the PCB under test with an enrolled threshold, a tampering determination can be made. In this section, the enrollment and detection flows are introduced. The enrollment process generates the model based on the RO frequencies. This model will be exploited for monitoring tampering during the detection process.

A. Model Enrollment

Both the detection and reference ROs consist of multiple CMOS inverters, which are sensitive to temperature variations. The detection ROs also consist of PCB traces which are less impacted by temperature. Since the RO frequency is utilized as an indicator of the tampering attack, it is expected that the tampering is the dominant factor, which changes the frequency of ROs. Unfortunately, the effect of temperature on the detection ROs is not negligible. The experimental proof and discussions are provided in Section V.

During the detection, the non-tampering frequency (i.e., genuine frequency) of each detection RO is the criteria to be compared with. Due to the non-negligible temperature effects, the genuine frequencies for different temperatures need to be adjusted. To achieve this adjustment, a model can be computed from the measurements collected at multiple temperature points. To avoid enrolling all the possible temperatures, it is crucial to build a temperature-dependent module which estimates the genuine frequency under current temperature. A general expression of this model is described as below:

$$f_{det} = G(f_{ref}) \tag{1}$$

where, f_{det} refers to the frequency of each detection RO and f_{ref} refers to the frequency of reference RO. f_{det} represents the genuine frequency. The frequency measurement f_{ref} can be considered as the temperature reference. Thus, each frequency pair, (f_{det}, f_{ref}), indicates the relationship between the genuine frequency with certain temperature condition. A third-order polynomial $G(x)$ is utilized to model this relationship. This polynomial has four parameters, $p_{3\ldots0}$, as shown in the following equation.

$$G(x) = p_3x^3 + p_2x^2 + p_1x^1 + p_0 \tag{2}$$

To calculate these parameters, the reference and detection RO frequencies are measured under three temperature conditions. For each temperature condition, both the detection and reference ROs are evaluated ten times. These measurements are collected by a dedicated firmware which is downloaded into the detector by the designer. The enrollment data are utilized to compute the model, $G(x)$, by applying the polynomial fitting. An experimental example of this model construction process can be found in Section V-C.

Besides the model parameters, three standard deviations are computed from the genuine frequencies under enrollment temperatures. The model ($G(x)$) and the largest standard deviation (σ_{max}) are stored. Finally, the enrollment firmware is replaced by an detection firmware, which holds these parameters. This firmware is designed to detect and respond to tampering. For security considerations, the programming capability of the SoC may need to be disabled from updates by anyone but the designer.

B. Detection

During in-field startup, the system enters the testing mode (see Section IV-A). Based on the enrolled model, standard deviation, and reference RO frequency, the lower and upper boundaries (b_{lo} and b_{up}) can be formulated as

$$b_{lo} = G(f_{ref}) - 3 \times \sigma_{max};\; b_{up} = G(f_{ref}) + 3 \times \sigma_{max} \quad (3)$$

In Equation (3), $\pm 3 \times \sigma$ indicates the offset for tolerating the frequency fluctuations of the same RO. These fluctuations are due to the physical process variations of the SoC. Due to process variations, the same detection RO may produce different frequencies even though the temperature condition is the same. This offset aims to capture the process variations and reduce the errors.

When tampering induces additional impedance to the traces (e.g., mod-chips or jumpers added), the frequency of detection RO should decrease. Thus, when any tampering attack occurs, the frequency of the detection RO should be located outside the range between b_{lo} and b_{up}. By comparing the detection RO frequency with these two boundaries, the detector can determine whether the system is being tampered or not. When traces are opened (disabled) on the PCB, the RO frequency can increase, which our framework could also detect. An experimental example of this comparison can be found in Section V-C. When the tampering attack is identified, the detector suspends the system from the normal operation by either blocking the signals (E-mode detector) or sending out the wrong signals (S-mode detector). The latter is achieved using a permutation/obfuscation approach.

IV. MPA Hardware Implementation

In this section, the hardware implementation of MPA is discussed. The design modification scenarios are provided in Section IV-A. As mentioned in the previous section, a detector is embedded into the original design to apply MPA. The design of the detector and its modules are introduced in Section IV-B.

A. Design Modification

A chip set and the critical connections are shown in Figure 3(a), *Chip 1* and *Chip 2* indicate two active components on the PCB. The original design refers to the system with no MPA incorporated. In this figure, the labels t_1 and t_2 refer to the groups of ports from chip 1 and chip 2. The critical connections between ports are the ones which should be protected from tampering. A practical example can be found in Figure 2, where DSP and BIOS can be these two chips.

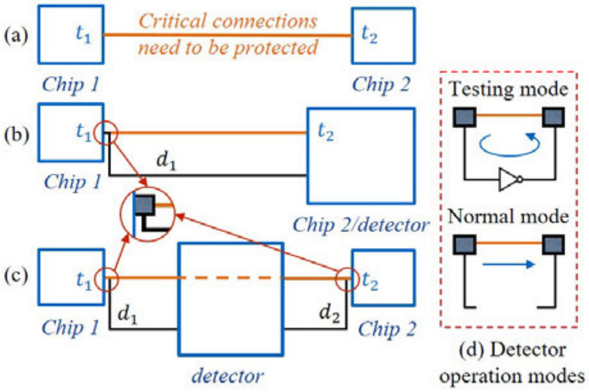


Fig. 3. High-level implementation diagram and operation modes. (a) Critical connections in the original design. (b) The embedded mode detector scenario. (c) the standalone mode detector scenario. (d) Two detector operating modes.

In order to implement MPA, the detector is involved in the original design to protect these critical connections. Considering different application scenarios, the protected system can be built as one of the following modes: *embedded mode* (**E mode**) in Figure 3(b) and *standalone mode* (**S mode**) in Figure 3(c). For the embedded mode, the functionalities of chip 2 are merged into the detector. Multiple dummy traces (labeled d_1 in Figure 3(b)), are placed to connect the pads of chip 1 with the detector. These connections of the pads are shown in a zoomed view. The standalone mode addresses the situation when the functionalities of chip 2 cannot be combined with the detector. In this mode, a standalone detector is established in between of these two original chips (i.e., Figure 3(c)). Since the critical connections are split into two parts, two groups of dummy traces (labeled d_1 and d_2) are implemented respectively. *Note that the two chip examples are used for simplicity only and the proposed detector can be applied to monitor traces of multiple chips at once.*

In order to apply MPA, chip 1 and chip 2 should be compatible with the Joint Test Action Group (JTAG) IEEE Std 1149.1 (boundary scan) instructions. The JTAG standard supports the following instruction: EXTEST, PRELOAD, SAMPLE, HIGHZ, etc. The JTAG standard is widely utilized to test electrical connections of the ICs and has been implemented by many manufacturers, such as Actel, Microsemi, and Altera in their products. By configuring the JTAG infrastructures of chip 1 and chip 2, the detector can operate in two modes (Figure 3(d)): **testing mode** and **normal mode**. In testing mode, the JTAG infrastructure is configured into HIGHZ mode, which sets all pin outputs into the high-impendence state. Thus, internal logics of these chips are isolated from port groups t_1 and t_2. The dummy traces link these ports and critical traces into a ring through the inverter bank (more details in Section IV-B3). Thus, a group of ROs which cover these critical traces are constructed. The frequencies of these ROs are exploited as tampering indicators. During the normal mode, the detector disconnects the dummy traces and the inverter bank, and the JTAG infrastructure is deactivated. In this mode, the normal operation of the system can proceed. The detailed design of the detector will be elaborated upon in the following section.

B. Detector Design

The schematics of the E-mode and the S-mode detectors are provided in Figure 4. The **E-mode detector** in Figure 4(a) consists of two groups of ports: t_1 and d_1. The **S-mode detector** in Figure 4(b) contains four groups of ports: t_1, t_2, d_1, and d_2. t_1 and t_2 are connected to the ports of chip 1 and chip 2 with the same labels. d_1 and d_2 are the ports connected with the dummy traces.

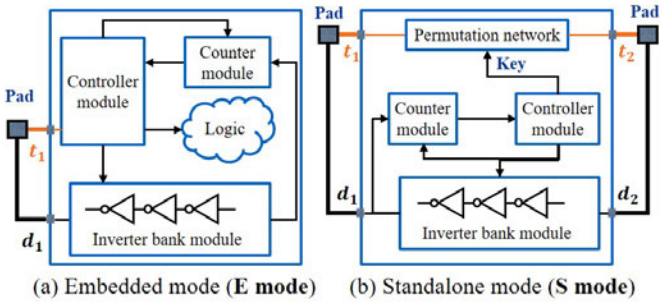
The S-mode and E-mode detectors have three modules in common: the *counter*, *controller*, and *inverter bank* module. Besides these common modules, the *logic* module in the E-mode detector refers to the functions which are inherited from the merged chip (Chip 2 in Figure 3). The *permutation network* module in the S-mode detector indicates a key-control network which permutes the connections between t_1 and t_2. The functions of permutation network and three common modules are introduced below.

1) Permutation network module: The permutation network module is only implemented in S-mode detector. This module connects the input group t_1 to the output group t_2 as shown in Figure 3(b). This permutation network takes multiple input signals and routes them to the outputs in a specified order. This input-to-output order is controlled by a key. Since in the S mode, the detector behaves as a transceiver which receives the signals from chip 1 and transmits them to chip 2. Thus, only the correct input-to-output order (i.e., the correct key) can drive the board to operate properly.

For the S-mode detector, the permutation network module performs the function of tampering response. When the tampering is detected, this module receives a wrong key and mismatches the inputs in group t_1 and outputs in group t_2. In this case, chip 2 accepts the misguided signals and the correct system function is disabled. Additionally, since the permutation network is an essential component of the obfuscation-based PCB anti-reverse engineering framework [9], introducing it provides anti-reverse engineering capability to MPA. As proved in [9] it will be extremely complicated for the attacker to guess the correct key.

2) Counter module: This module is connected to the inverter bank module and exploited to measure their associated RO frequencies during the testing mode. Both the E-mode and S-mode detectors share the same counter design. The controller module specifies the time instants when the counter starts and completes counting. The counter module sends the counting outcomes to the controller module for further analysis. In MPA, the reference RO operates as an indicator for providing the temperature effect adjustment of the detection RO. Since the reference and detection ROs share the same counter module, the temperature effect on the counter module is self-adjusted.

3) Inverter bank module: The schematic of the inverter bank module is presented in Figure 5. This module achieves two ring oscillator (RO) configurations: reference RO and detection RO. To select from different RO configurations, the MUX controller sets different selection signals for the two multiplexers (i.e., n-to-1 and 2-to-1 MUX) and two demultiplexers (i.e., 1-to-n and 1-to-2 DEMUX). This MUX controller receives the control signals from the controller module and delivers the configurations to different MUXes/DEMUXes. These MUXes/DEMUXes compose the inverter chain into either reference or detection ROs.



(a) Embedded mode (**E mode**) (b) Standalone mode (**S mode**)

Fig. 4. The detector designs: (a) embedded mode and (b) standalone mode

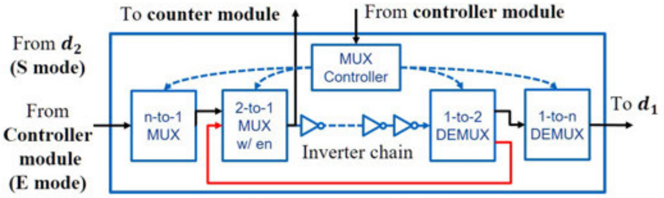


Fig. 5. The schematic of the inverter bank model. The connections to the counter model, controller model, and from d_1 are the same for E-mode and S-mode detectors. The inputs of the n-to-1 MUX are different for different detector modes.

The reference RO is formed by joining the two ends of the inverter chain together. This structure is shown as the red path in Figure 5. For the construction of a single detection RO, the inverter bank module receives a group of inputs from either the controller module (E-mode detector) or d_2 (S-mode detector). The n-to-1 MUX selects one of these inputs and assigns it to one of the inputs of the 2-to-1 MUX. Next, the 2-to-1 MUX routes the output of the n-to-1 MUX to the input of the inverter chain. The output of the inverter chain is linked to the input of the 1-to-n DEMUX by the 1-to-2 DEMUX. Finally, the input of the 1-to-n DEMUX is transmitted to one of the traces among the output group d_1. The path of the detection RO is presented as the black lines in Figure 5. The controller module selects one detection RO at one time for the frequency measurement among all the detection ROs. This selection can be achieved by setting both the n-to-1 MUX and 1-to-n DEMUX.

Additionally, the 2-to-1 MUX has an *enable* function. When the enable signal is asserted, this multiplexer behaves as a normal 2-to-1 MUX. Otherwise, this multiplexer is disabled, and the output is not connected with any logic (i.e., floating or blocked). In this case, the inverter chain is not isolated, and the inverter bank module is open. Thus, for both the E-mode and S-mode detectors, this enable signal is asserted during the testing mode and de-asserted during the normal mode as shown in Figure 3(d).

4) Controller module: This module acts as a central controller and processor of the detector. It provides the control signals and receives/sends data from/to other modules. The connections of the controller module and other modules can be found in Figure 4. This controller module performs several different functions. Some of them are applied to the controller module in both E-mode and S-mode detectors. These functions are labeled as $\{S\ \&\ E\ mode\}$. Other functions, which can be applied only in either E-mode or S-mode detectors, are labeled as $\{E\ mode\ only\}$ and $\{S\ mode\ only\}$. The functions of the controller module are summarized in the following list:

- $\{S\ \&\ E\ mode\}$ Send the MUX configuration signal to the inverter bank module. This signal activates the testing mode and selects the RO under measurement from the reference RO and n detection ROs.
- $\{S\ \&\ E\ mode\}$ Send the counter control signal to the counter module. This signal sets the starting time instant and the counting duration.
- $\{S\ \&\ E\ mode\}$ Collect the output of the counter module and compute the RO frequency. This frequency is exploited as tampering detection indicator.
- $\{S\ mode\ only\}$ Send the correct key to the permutation network when no tampering is detected. Otherwise, a wrong key will be sent and the normal system functions will not

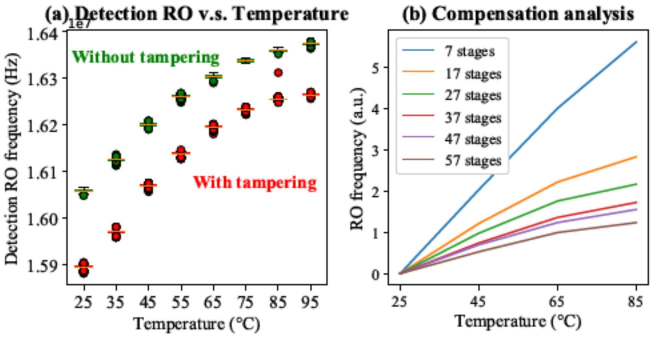


Fig. 6. (a) Detection RO frequency v.s. temperature with/without the tampering. (b) RO frequency cancellation analysis with different length of the inverter chain.

be obstructed.

- $\{E$ *mode only*$\}$ During the testing mode, the controller module connects the input group t_1 to the inverter bank module. During the normal mode, t_1 is connected to the *Logic* by the controller module if no tampering is detected. Otherwise, the paths between t_1 and *Logic* will be blocked.

V. Experimental Results

In order to validate the tampering detection performance of MPA, we implemented the detector in a commercial SoC and custom designed test PCBs. The experimental setups and hardware overheads based this setup are provided in Section V-A. In Section V-B, temperature effects and the necessity of modeling are discussed. Finally, A modeling example and the evaluation results are provided in Section V-C.

A. Experimental Setup

The experimental platform is shown in Figure 8. The detector is implemented in a Zynq SoC on a ZYBO development board. This SoC consists of an ARM Cortex-A9 processor and an Artix-7 FPGA. The processor is mainly utilized to communicate with the desktop and the FPGA is utilized to implemented the detector. Each test PCB consists of 8 ROs which cover the traces with different length. The test PCB is connected to the detector through the on-board Pmod connections. The ZYBO development board sends the RO frequency measurements to the laptop through UART for further analyses. Table II shows the FPGA resources utilizations which are reported by the Xilinx Vivado design suite. The power consumed by the FPGA during the operation is reported as 0.016W.

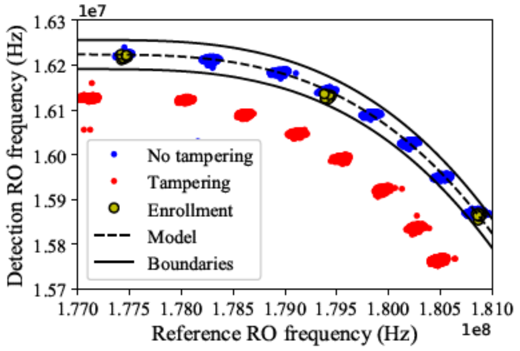


Fig. 7. Modeling example. The detection RO are evaluated 1,000 times for both tampering and non-tampering conditions under eight temperatures.

TABLE II
FPGA Resources Utilizations

Primitive type	Flip-Flops	Look-up tables	Carry logic	Other
Count	363	757	78	4
Percentage	1.03%	4.30%	1.12%	0.20%

In our experiment, we perform 1,000 measurements for each detection RO across eight temperature conditions ranging from 25°C to 95°C. These measurements are obtained under both tampering and non-tampering conditions. In order to examine the detection sensitivity of MPA, the Tektronix TDP1000 probe is utilized to mimic the tampering. We select this probe model for its negligible input capacitance ($< 1pF$). In practice, the active tampering attacks such as mod-chips are expected to induce largermore impedance.

B. Temperature Effect

To perform the analysis theanalyze the effects of temperature effect, we sweep the environmental temperatures from 25°C to 95°C with a step size of 10°C. 1,000 RO frequencies are collected from each temperature condition. The results are provided in Figure 6(a). In this figure, where we compare the RO frequency changes induced by the temperature and tampering. When no tampering does not occurpresents, the detected RO frequency at 25°C is 1.605e7 Hz. While under the tampering condition at 45°C, this frequency becomes 1.607e7 Hz. These two frequency measurements collected from the tampering and non-tampering conditions are close. If the genuine frequency is enrolled at 25°C, the tampering at 45°C will be falsely determined. Thus, the enrolled genuine frequencies should be adjusted to different temperature conditions, and the modeling is required.

Theoretically, high temperature slows down the transistor's switching speed. Thus, the RO frequency should decrease when the temperature becomes high. However, in Figure 6(a), the frequency of the detection RO follows the same direction as the temperature. This “irregular” observation is possibly due to the impact which derives from the pad of the SoC port. To compensate for the slow-down caused by higher temperatures, the pad of the port is designed to cancel out a certain amount of delay. To verify this assumption, we measure the detection RO frequencies when the inverter chain consists of a different number of stages (i.e., inverters). The experimental results are summarized in Figure 6(b). The frequencies collected from the same inverter chain are normalized by subtracting the frequency value at 25°C. According to this figure, the longer the inverter chain is, the slower the frequency grows with the temperature. This finding indicates that the delay compensation from pad becomes less significant when more delay is generated by a longer inverter chain.

C. Performance Evaluation

In Figure 7, an example of modeling and detection processes are presented. For enrollment, the reference and detection RO frequencies are measured under three temperature conditions: 25°C, 65°C and 95°C. These enrollment data are marked in yellow. The enrolled model, which has a polynomial form as shown in Equation (2), is presented as the dashed curve. The solid curves refer to the upper and lower boundaries, b_{lo} and b_{up}, which can be computed by Equation (3). In this figure, the eight clusters of blue dots indicate the genuine detection RO frequencies, which are collected under eight different temperatures. The red dots show the tampered frequency measurements. The split of the frequencies under the same temperature are caused by process variation. It can also be

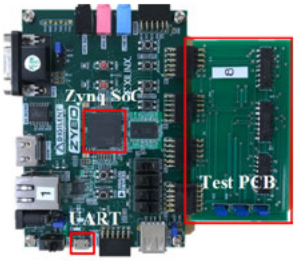


Fig. 8. Experimental setups based on ZYBO.

	RO0	RO1	RO2	RO3	RO4	RO5	RO6	RO7
$25°C$	99.77%	100%	99.99%	100%	100%	99.99%	99.93%	99.99%
$35°C$	99.81%	100%	100%	100%	100%	100%	100%	100%
$45°C$	99.97%	99.99%	99.98%	99.99%	100%	100%	99.98%	99.99%
$55°C$	100%	100%	100%	100%	100%	100%	100%	100%
$65°C$	100%	100%	100%	100%	100%	100%	100%	100%
$75°C$	100%	100%	100%	100%	100%	100%	100%	100%
$85°C$	100%	100%	100%	100%	100%	99.99%	100%	100%
$95°C$	100%	100%	100%	100%	100%	100%	100%	100%

TABLE I
TAMPERING DETECTION SUCCESS RATES.

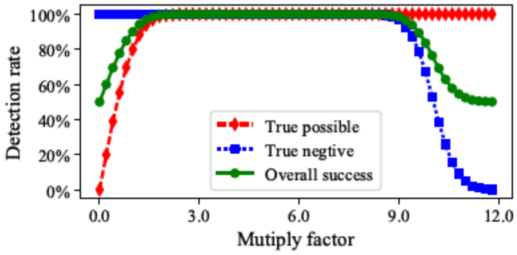


Fig. 9. Offset analysis by varying the multiplying factor. This factor is the multiplying coefficient of σ in Equation (3)

observed that the boundaries fully capture these frequency fluctuations.

We perform the MPA detection accuracy evaluation on eight testing platforms. The results are summarized in Table I. In this table, the detection **success rate** is utilized as the major metric to evaluate MPA. This metric can be computed by:

$$\text{Success rate} = \frac{(TPR + TNR)}{2} \tag{4}$$

where the true positive rate (TPR) measures the proportion of non-tampered situations that are correctly identified as such. The true negative rate (TNR) measures the proportion of tampered situations that are correctly identified as such.

Each detection success rate in Table I is calculated by averaging the corresponding success rates from all the testing platforms under the same temperature and RO. According to Table I, the detection accuracy is higher than 99.7% among all temperature conditions and ROs. The detection errors are possibly due to the offset estimation. The offset is computed based the 68-95-99.7 rule (i.e., three-sigma rule of thumb). This rule expresses a conventional heuristic that 99.7% values are taken to lie within three standard deviations of the mean. However, when this offset fails to capture all the non-tampered frequencies, the errors are observed.

We also investigated the effect of different offsets on the success rates. The multiplying factor is swept from 0 to 12. This factor is the coefficient of σ in Equation (3). The TPR, TNR, and success rates, which are computed from one RO, are provided in Figure 9. When the multiplying factor (i.e., the offset) is small, the true positive is low due to the overlapping between the boundaries and the non-tampered frequencies. When the multiplying factor is large, the true negative rate decreases since lower boundary overlaps with the tampered frequencies. Due to the overlapping of the tampered and non-tampered RO frequencies, for some ROs, the highest success rate is constantly 99.7% with all multiplying factors.

VI. CONCLUSION

In this paper, we proposed a model-assisted PCB attestation (MPA) framework against the tampering such as mop-chips. The proposed framework includes a testing mode to detect tampering attacks before the system starts its normal operation. This test mode can be initialized by a detector by configuring the JTAG infrastructures into HIGHZ mode. The proposed MPA technique is validated by testing platforms, and the results show a high detection accuracy (> 99.7%).

VII. ACKNOWLEDGEMENT

This project was supported in part by US Army Research Office grant under award number grant W911NF-16-1-0321 and by the AFOSR MURI grant under award number FA9550-14-1-0351.

REFERENCES

[1] S. Paley, T. Hoque, and S. Bhunia, "Active protection against pcb physical tampering," in *Quality Electronic Design (ISQED), 2016 17th International Symposium on*. IEEE, 2016, pp. 356–361.

[2] S. E. Quadir, J. Chen, D. Forte, N. Asadizanjani, S. Shahbazmohamadi, L. Wang, J. Chandy, and M. Tehranipoor, "A survey on chip to system reverse engineering," *ACM journal on emerging technologies in computing systems (JETC)*, vol. 13, no. 1, p. 6, 2016.

[3] J. Grand, K. D. Mitnick, and R. Russell, *Hardware hacking: have fun while voiding your warranty*. Syngress, 2004.

[4] "How to reverse engineer a schematic from a circuit board: 18 steps. [online]. available: http://www.instructables.com/id/how-to-reverse-engineer-a-schematic-from-a-circuit/."

[5] "Pcb copy and pcb clone-shenzhen sichi technology co., ltd. [online]. available: http://www.pcbic-reverse.com/service/show.php?lang=en&id=285."

[6] U. Guin, D. DiMase, and M. Tehranipoor, "Counterfeit integrated circuits: detection, avoidance, and the challenges ahead," *Journal of Electronic Testing*, vol. 30, no. 1, pp. 9–23, 2014.

[7] "F. domke, blackbox jtag reverse engineering, 2009, [online]. available: http://events.ccc.de/congress/2009/fahrplan/events/3670.en.html."

[8] "Modchip: Wikipedia, the free encyclopedia. [online]. available: https://en.wikipedia.org/wiki/modchip."

[9] Z. Guo, M. Tehranipoor, D. Forte, and J. Di, "Investigation of obfuscation-based anti-reverse engineering for printed circuit boards," in *Proceedings of the 52nd Annual Design Automation Conference*. ACM, 2015, p. 114.

[10] Z. Guo, B. Shakya, H. Shen, S. Bhunia, N. Asadizanjani, M. Tehranipoor, and D. Forte, "A new methodology to protect pcbs from non-destructive reverse engineering."

[11] A. Hennessy, Y. Zheng, and S. Bhunia, "Jtag-based robust pcb authentication for protection against counterfeiting attacks," in *Design Automation Conference (ASP-DAC), 2016 21st Asia and South Pacific*. IEEE, 2016, pp. 56–61.

[12] F. Zhang, A. Hennessy, and S. Bhunia, "Robust counterfeit pcb detection exploiting intrinsic trace impedance variations," in *VLSI Test Symposium (VTS), 2015 IEEE 33rd*. IEEE, 2015, pp. 1–6.

[13] "mod-chip.net. [online]. available: http://www.mod-chip.net/index.htm."

[14] "How the original iphone was hacked [online]. available: http://www.imore.com/original-iphone-hacked."

[15] "How to: Convert your xbox to a nas [online]. available: http://www.tomsguide.com/us/how-to-xbox-nas-pt1,review-608.html."

[16] "Photos of an nsa upgrade factory show cisco router getting implant [online]. available: https://arstechnica.com/tech-policy/2014/05/photos-of-an-nsa-upgrade-factory-show-cisco-router-getting-implant/."

[17] "System-level tamper protection using msp mcus. [online]. available: http://www.ti.com/lit/an/slaa715/slaa715.pdf."