Co-optimization of security and accessibility to on-chip instruments

Larsson, Erik

Published in:
2023 IEEE 24th Latin American Test Symposium (LATS)

DOI:
10.1109/LATS58125.2023.10154500

2023

Co-optimization of security and accessibility to on-chip instruments

Erik Larsson
Lund University
Lund, Sweden
Email: erik.larsson@eit.lth.se

***Abstract*—The semiconductor technology development constantly enables integrated circuits (ICs) with more, faster and smaller transistors. While there are many advantages, there are also many and new challenges, for example tighter margins, wear-outs and process variations. To address these challenges, the traditional approach with external test instruments used at manufacturing test must be complemented with on-chip instruments to provide possibilities to test for defects that manifest themselves during the operational lifetime. These on-chip instruments provide, on one hand, better controllability and observability, which is helpful for testing purposes. On the other hand, the increased possibility to control and observable the IC's internals can be a security risk. We discuss how to provide access and how to co-optimize security and accessibility for these on-chip instruments.**

I. Introduction

The semiconductor technology development makes it for every generation possible to produce integrated circuits (ICs) with more, faster and smaller transistors. There are many advantages but also several challenges. The increasing transistor count puts pressure on development time and integration. Faster and smaller transistors give tighter margins and new effects, like process variations and wear-out problems.

The traditional approach with external test instruments used at manufacturing test must be complemented with on-chip instruments to provide possibilities to test for defects that manifest themselves during the operational lifetime. In automotive industry, ISO26262 demands in-field test on a regular basis [1]. An example of work in this direction is that by Tille *et al.* [2] where a Digital Twin is used to re-compute Logic Built-In Self-Test (LBIST) signatures in-field. In addition, there are reports that hardware in server farms are subject to unexpected errors during operation [3], [4].

To provide testing in operational life-time, increased controllability and observability are needed, typically with the help of on-chip instruments. However, this increased controllability and observability can be misused, for example by observing internal states or controlling the operation.

In this paper we discuss how to provide access to on-chip instruments and how to co-optimize security and accessibility. The rest of the paper is organized as follows. In Section II the need of on-chip instruments is illustrated and existing ways to access these instruments are introduced [5], [6], [7], [8], [9], [10], [11]. A way to secure the test access port without embedding a private key is discussed in Section III. As key handling can be cumbersome, Section IV discusses how to partition the instruments such that a subset can be accessed via the functional bus.

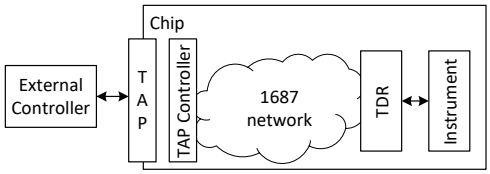


Fig. 1. An typical way to access instruments

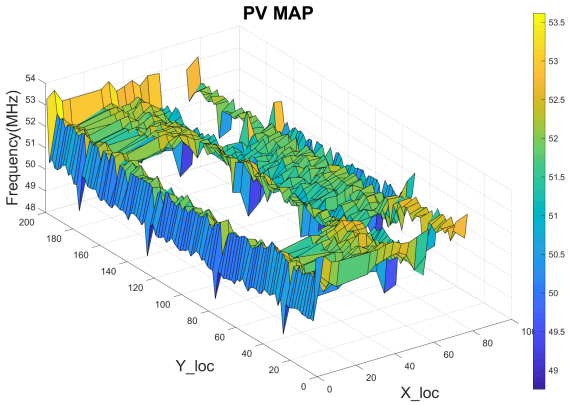
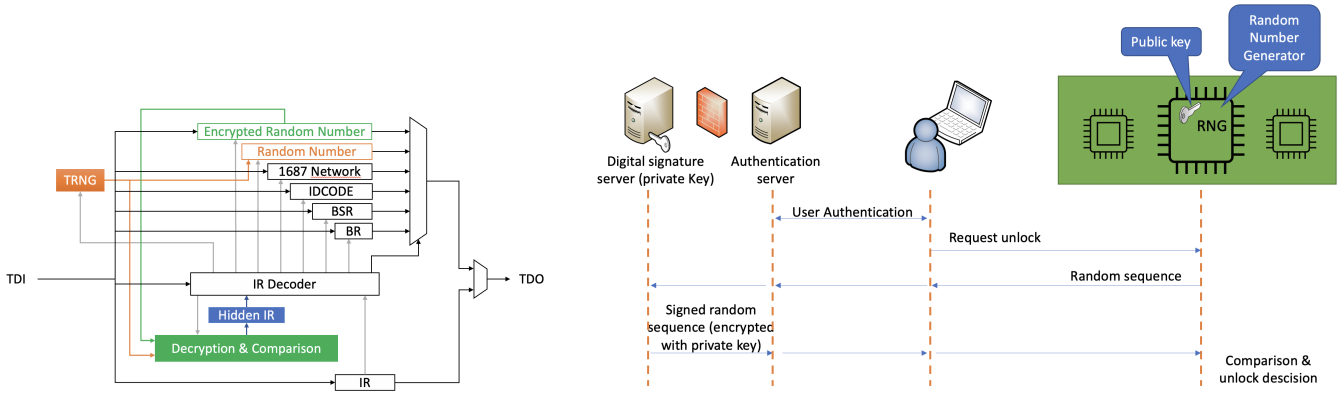


Fig. 2. Performance variation (PV) at different Nexys 4 FPGA boards

II. Instruments

Figure 1 shows a typical way to access on-chip instruments from an external controller. The external controller is connected to the chip via the test access port (TAP) of IEEE Std. 1149.1, also known as JTAG [12], [13]. JTAG is connected to an IEEE Std. 1687 network of test data registers (TDR), which in turn are connected to instruments [14]. Connecting instruments with IEEE 1687 enable flexible and dynamically configurable access [15]. To get a feeling for the need of instruments, Pengxian [16] developed a method for monitoring performance variation (PV). The method was applied to Digilent Nexys4 boards equipped with 28nm XILINX ARTIX 7 XC7A100T FPGAs. Figure 2 shows the variation of clock frequency in a single FPGA device based on 1400 instruments. Experiments were performed with 20 devices where each showed a unique profile.



(a) Details of the JTAG TAP

(b) Overview

Fig. 3. Securing the JTAG port

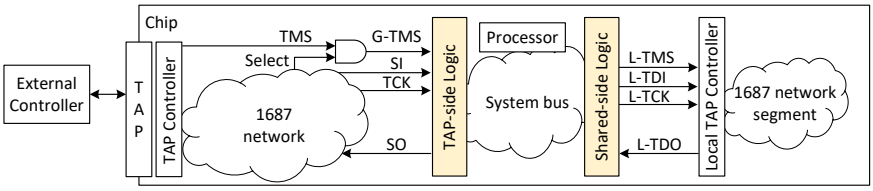


Fig. 4. Using Serial Transfer to share a network segment over a system bus

III. Securing the Test Port

JTAG can be used for attacks such as extracting information from chips connected to the same JTAG network [17], [18]. Cryptography is a way to provide authentication. While it is efficient to store private keys in hardware, a danger is that keys might get lost. We developed a scheme using public keys. Figure 3 overviews the scheme. A user requests an unlock and a public key in from of a random sequence is created by the device. A signed random sequence is then created by the digital signature server, which keeps the private key. At the device, the signed random sequence is compared against the locally generated random sequence. Figure 3(a) shows the detailed JTAG TAP. The TDRs for Encrypted Random Number and Random Number are accessible via the Instruction Register (IR) Decoder. For the implementation, we learned that True Random Number Generator (TRNG) does not incur much hardware overhead, but high hardware overhead is needed for the decryption algorithm (used for verification of the signed random number).

IV. Sharing Instruments

In Section III, access control of the JTAG TAP was discussed. One short-coming is that access to instruments during operational lifetime is cumbersome. Zadegan *et al.* propose a scheme where some instruments are shared such that they are accessible from both the test port, like IEEE Std. 1149.1, and an processor [15]. Figure 4 shows a system where an external controller can access all instruments in the IEEE Std. 1687 network via the IEEE Std. 1149.1 TAP while the processor only can access a subset of the instruments. Zadegan *et al.* analyzed the requirements and derived some constraints on timing in respect to test clock (TCK) and system clock (CLK) which must be fulfilled so that an EDA-tool can be used without explicit knowledge of the sharing.

References

[1] ISO 26262, "ISO26262: Road Vehicles Functional Safety-part 5," 2018.

[2] D. Tille et al., "A novel LBIST signature computation method for automotive microcontrollers using a digital twin," in *VTS*, 2023.

[3] H. D. Dixit et al., "Silent data corruptions at scale," *CoRR*, vol. abs/2102.11245, 2021.

[4] P. H. Hochschild et al., "Cores that don't count," in *Workshop on Hot Topics in Operating Systems, USA, June, 2021*. ACM, 2021, pp. 9–16.

[5] F. G. Zadegan et al., "Test Time Analysis for IEEE P1687," in *Proc. ATS*, 2010, pp. 455–460.

[6] F. Ghani Zadegan *et al.*, "Design automation for IEEE P1687," in *Design, Automation & Test in Europe Conference (DATE)*, 2011.

[7] F. Zadegan *et al.*, "Access time analysis for ieee p1687," *IEEE Transactions on Computers*, vol. 61, no. 10, pp. 1459–1472, Oct 2012.

[8] R. Krenz-Baath et al., "Access time minimization in IEEE 1687 networks," in *International Test Conference (ITC)*, 2015.

[9] E. Larsson, P. Murali, and G. Kumisbek, "IEEE Std. P1687.1: Translator and Protocol," in *International Test Conference*, 2019, pp. 1–10.

[10] E. Larsson et al, "System-level access to on-chip instruments," in *2021 IEEE European Test Symposium (ETS)*, 2021, pp. 1–6.

[11] ——, "Accessing general IEEE std. 1687 networks via functional ports," in *2021 IEEE International Test Conference (ITC)*, 2021, pp. 354–363.

[12] "IEEE standard test access port and boundary-scan architecture," *IEEE Std 1149.1-2001*, 2001.

[13] "IEEE standard for test access port and boundary-scan architecture," *IEEE Std 1149.1-2013 (Revision of IEEE Std 1149.1-2001)*, 2013.

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

[15] F. G. Zadegan et al., "Reusing IEEE 1687-compatible instruments and sub-networks over a system bus," in *Int. Test Conf.*, 2022, pp. 219–228.

[16] P. Cheng, "Study of monitoring circuitry for ageing in FPGAs," 2021, Master thesis at Lund University, Sweden.

[17] G. Vishwakarma and W. Lee, "Exploiting JTAG and Its Mitigation in IOT: A Survey," *Future Internet*, vol. 10, no. 12, 2018.

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