# Comments on "A PUF-FSM Binding Scheme for FPGA IP Protection and Pay-per-Device Licensing"

Brice Colombier, Lilian Bossuet

# A Comment on "A PUF-FSM Binding Scheme for FPGA IP Protection and Pay-per-Device Licensing"

L. Bossuet, *Senior Member, IEEE*, B. Colombier

***Abstract*—IP protection is a recent field of research. If passive protection schemes, mainly IP watermarking and fingerprinting, have been studied for more than fifteen years, active protection schemes using remote activation / unlocking / metering of IPs are highlighted by several recent works. Like any other new field of research, new concepts appear with sometimes not such good ideas. IP unlocking scheme without cryptography, as recently proposed in this journal, is one of these ideas. Expecting to obtain low overhead and high security this way is very hard. This comment proves this by presenting a short yet deep study.**

***Index Terms*—IP protection, security vs overhead, cryptography.**

## I. Introduction

More and more works are focused on IP protection and also IC counterfeiting. In the last decade, several academic works have proposed secure IPs active protection schemes for FPGA, allowing pay-per-device licensing [1]. All these works have in common to involve a trusted third party (which could be an independent trusted entity or the FPGA vendor itself) and to use cryptographic functions. To guarantee a high security level, these works mainly use public key cryptography and/or symmetric ciphers. Of course, using such cryptographic functions costs a lot of area and reduces the overall performance of the protected IP. But security is rarely free. However, a recent paper has proposed new IP pay-per-device licensing without using any cryptographic functions [2]. Despite the security guarantee provided by cryptographic functions, authors of this paper claim that the proposed IP protection scheme is more secure than other previously published schemes. Moreover, they present this IP protection as a secure and low overhead solution. Such a proposition is sufficiently original to require a deep study on the security / overhead tradeoff actually achieved.

In this paper we highlight the drawbacks of the IP protection scheme proposed in [2] and we propose some recommendations for IP designers concerned about the protection of their IP and interested in a pay-per-device licensing of IP deployment.

Manuscript submitted March 31th, 2015. The work presented in this paper was realized in the frame of the SALWARE project number ANR-13-JS03-0003 supported by the French "Agence Nationale de la Recherche"and by the French "Fondation de Recherche pour l'Aéronautique et l'Espace". This work was funded in parts also by a grant from "La Région Rhône-Alpes".

L. Bossuet and B. Colombier are with the Laboratoire Hubert Curien, University of Lyon, Saint-Etienne, 42000, France (e-mail: lilian.bossuet@univ-st-etienne.fr, b.colombier@univ-st-etienne.fr).

## II. On the Security and the Overhead of the PUF-FSM Binding Scheme

### A. The PUF-FSM binding scheme

The PUF-FSM binding scheme proposed in [2] is based on a previously published locking FSM scheme [3]. The main idea is to add a locked FSM called binding FSM to the IP FSM. The binding FSM is used to control access to the IP original behavior. In [2], in order to unlock the binding FSM and access the IP FSM, the IP end user must provide it with a license. This license is computed by the IP core vendor. It depends on the response of the PUF embedded in the same FPGA as the IP. Both the response of the PUF and the license are used to unlock the binding FSM state-layer by state-layer.

The binding FSM comprises $M$ (an even number) layers of states. Any even-number layer of the binding FSM consists of $m$ states, and any odd-number layer has only one state [2]. The $m$ transitions from an odd-number layer are based on $k$ bits of the PUF response. Each transition from one of the $m$ states of any even-number layer is based on $k$-bit xor output. This xor uses $k$ bits from the PUF response and $k$ bits from the license. According to this behavior, authors of [2] provide the two following equations (1) and (2) to compute the length of the PUF response $L_{PUF}$, and the length of the license $L_{license}$, as a function of $m$ and $M$. According to these two equations, we can simply deduce than $L_{license}$ has to be equal to half $L_{PUF}$.

$$L_{PUF} = M \times \log_2 m \qquad (1)$$

$$L_{license} = \frac{M}{2} \times \log_2 m \qquad (2)$$

In the following sub-sections we study the security-overhead tradeoff provided by such IP protection scheme.

### B. Security of the low overhead solution

Hardware implementation results provided in [2] show a very small overhead for large benchmarked FSMs (from 300 to 2000 states). These FSMs are generated randomly. In this case, the mean timing overhead is 0.64%, the mean power overhead is 0.01% and the mean logical resources overhead is -2.67% (on the number of FPGA LUTs used). With smaller benchmarked FSM (from 27 to 218 states) the mean timing, power and logical resources overhead are more significant. They are 17.77%, 0.03% and 52.02% respectively.

In both cases, these results are given for the following $M$ and $m$ parameters values: 4 and 4. What is the security level provided by these parameters? According to Eq. (2) the length

of the license, $L_{license}$ is only 4 bits. Such a length does not provide any security against brute force attack. By using such an attack, assuming that the clock frequency of the FPGA implementation is 100 MHz (totally realistic frequency with a 28 nm FPGA) an attacker can find the correct license value in 80 ns (finding the correct license requires the maximum delay to activate the transitions over the 4 state layers). The attacker has to try 16 different license values at most. So, by using an over-estimation, the attacker needs 1.28 µs to test all the license values and find the correct one. This over-estimated delay increases to 252 ms when using higher *M* and *m*, respectively 12 and 10, i.e. using a 20-bit license. Nevertheless, according to [2], with such *M* and *m*, the overhead of the proposed solution is already not negligible: more than 40% for timing and more than 125% for logical resources (LUTs).

According to this first study, the low overhead (or limited overhead) configuration of the PUF-FSM binding (*M* lower than 12 and *m* lower than 10) does not provide any security.

To obtain a sufficient level of security, the license length should be a least 128-bit (according to section VII of [2]). The following section tries to estimate the overhead with such requirement on the license length.

### C. Overhead of the high security solution

To be protected against brute force attacks, according to the requirements from national agencies of security, a cryptographic key has to be a least 128-bit long. With such a requirement for the license (used as an unlocking key in [2]), by using Eq. (2) it follows that:

$$m = 2^{\frac{256}{M}} \qquad (3)$$

In order to estimate the overhead of the configuration with a 128-bit long license, we compute the number of states (*SN*) of the binding FSM. Half of it consists in *M/2* even-layers (with *m* states) and the other half in *M/2* odd-layers (with 1 state). It follows than the *SN* expression is given by Eq. (4).

$$SN = \frac{M}{2}\left(2^{\frac{256}{M}} + 1\right) \qquad (4)$$

Assuming *M* ranges from 1 to 256, the value for *SN* is 368 with *M* equals 256. According Eq. (4), choosing a low *M* leads to a very high *SN*. For example, with *M* equal to 16, Eq. (4) gives 524 296 states. Such a *SN* value produces a high overhead and is not practical. A lower *SN* value is reachable by using a higher value for *M*. Searching for the zero of the derivative of *SN* with respect to *M* given by Eq. (5), we find that *SN* is minimal when *M* equals 177. *M* has to be an even number though, so you can chose *M* equal to 178 (and according to Eq. (3) *m* equals 3) that leads to *SN* equals 330. The overhead should be minimal with this configuration.

$$\frac{dSN}{dM} = \frac{1}{2}\left(2^{\frac{256}{M}}\left(1 - \ln(2) \times \frac{256^2}{M}\right) + 1\right) \qquad (5)$$

In the binding FSM, according to the chosen configuration *(M=178 and m=3)* and the Eq. (2) the license length is 178 bits (96 times two bits). Such a value could imply a high security level ($L_{license}$*>128),* but a security failure can be used to attack the binding FSM easily. Indeed, each transition from even-layer states to odd-layer state the binding FSM uses 2 bits of the license, independently of the others bits of the license. As a consequence the security is reduced to the security of a 2-bit license (4 possible values, extremely easy to attack by using brute force). For any PUF value, in order to perform a brute force attack, the attacker checks the first transitions (from the first even-layer states) by using the 4 values for the first 2 bits of the license.

To increase the security, it is necessary to select a high value for *m* (small value of *M*) that leads to a very high overhead (very high *SN* value). In order to temper this comment, we claim that the overhead of cryptographic functions such as public key ciphers is also very high and even prohibitive for IP protection. A lightweight symmetric cipher could be an efficient way to provide a good security vs. overhead tradeoff.

## III. Recommendations for a Secure Pay-per-Device IP Licensing Scheme

Designing a secure protection scheme for IPs using an unlocking mechanism is not trivial, especially when low overhead is targeted. Nevertheless, the security has to be kept in focus all the time. The study in [2] leads us to propose some recommendations about how to design a secure pay-per-device IP licensing scheme:

1) License bits have to be used jointly and not separately or by small sub-sets. The security level, against brute force attack, is always reduced to the security of the smallest sub-set. For example, in order to improve the security of the FSM-binding, each false transition should lead to the reset state called $S_r$ in [2].
2) Unlike [2], not all FPGA configurations may have access to the PUF response. This information should never be sent outside the FPGA without encryption.
3) Even if hardware implementation of cryptographic functions are sensitive to physical attacks (side channel analysis, fault injection …), they provide a high security level. Yet the overhead of such functions is really high and may be limited by using lightweight implementations.

## IV. Conclusion

This comment is a study of the security/overhead tradeoff of the IP active protection scheme presented in [2]. This study concludes that this protection comes with high overhead to provide enough security; nevertheless it is a useful work for the community and for designers in the IP protection field.

## References

[1] B. Colombier, L. Bossuet. (2014, Nov.). A survey of hardware protection of design data for integrated circuits and intellectual properties. *IET Computers & Digital Techniques*, Vol. 8, No. 6, pp. 274-287.

[2] J. Zhang, Y. Lin, Y. Lyu, G. Qu. (2015, Jun.). A PUF-FSM Binding Scheme for FPGA IP Protection and Pay-per-Device Licensing. *IEEE Trans. on Infor. Forensics and Security*, Vol. 10, No. 6, pp. 1137-1150.

[3] Y. Alkabani, F. Koushanfar, M. Potkonjak, "Remote activation of ICs for piracy prevention and digital right management," *in Proc. ICCAD*, 2007, pp. 674-677.