Secured communications within a multi-processors architecture

Pascal Cotret, Guy Gogniat
Laboratory Lab-STICC
University of South Brittany
Lorient, France
pascal.cotret@univ-ubs.fr, guy.gogniat@univ-ubs.fr

***Abstract*—The development of secured embedded systems is in full expansion. The more we need complex treatments, the more we need to think about the security inside a system and its several elements. We participate in an ANR project aiming to develop a USB token able to cipher or decipher "on-the-fly" files stored on a computer. The USB token is based on a MPSoC and our work focus on the security of communications between the processors and the other IPs in order to avoid critical informations theft from attackers.**

***Index Terms*—MPSoC, communication architecture, cryptography, security protocols, security.**

I. Introduction

Thanks to the miniaturization of electronic systems, it is possible to include several processors within one circuit. Applications with security improvements are growing and they tend to integrate complex treatments. Security mechanisms should be flexible in order to follow the evolution of algorithms and protocols[1].
Embedded systems complexity implies a selective approach to define a smart distribution of security in terms of surface and latency. The global objective of our work is included into an ANR project, called SecReSoC, which aims to strengthen the security robustness of reconfigurable technologies such as FPGAs. We will focus on the communication between processors within a MPSoC architecture for a USB token (such as those you can find in Netheos[2] or even IronKey[3]). This architecture will also contain internal and external memory ressources, I/Os units and an internal communication architecture with several security levels.
First of all, we will present an architecture dedicated for a USB token: this system receives data frames from the USB port and basically do the ciphering (or deciphering). Then, we will set a threat model. Once the problem is clear, we will present some security solutions to counter the foreseen attacks. Finally, we will see if our proposals are reasonable according to the related work.

II. Overall presentation of the architecture

First of all, the system receives (or transmits) data frames from USB and operates the ciphering or deciphering according to its nature. The MPSoC architecture, based on PLB bus, contains several elements: processors, internal and external memories, processing IPs and every connection needed.

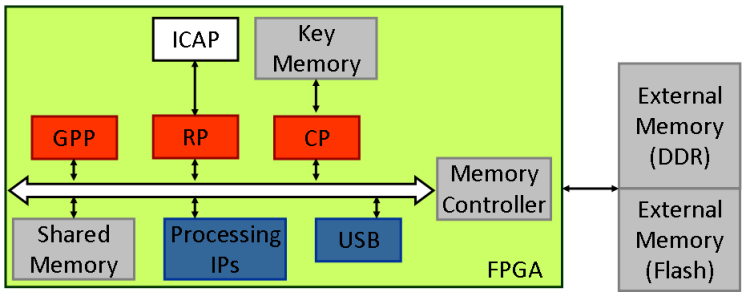

Fig. 1. Overall architecture of our system

A. Processors

The three microprocessors are the most important components of the system.

- **General Purpose Processor (GPP):** This processor will run the main application of the system in a multi-task operating system environment.
- **Cryptographic Processor (CP):** The CP is able to do symmetric cryptographic algorithms (such as AES, in several modes) using keys stored in an on-chip memory. His main task is the ciphering (or deciphering) of date frames received through the USB port.
- **Reconfiguration Processor (RP):** The RP should do the global (or partial) reconfiguration from a server where all the bitstreams are stocked. It is also responsible for the configuration of the security enhancements described in the section IV.

III. Threat model

This embedded system has several weak points that attackers can exploit to get critical informations such as cryptographic keys. The first kind of attacks we see is an illegal access to an external memory. We have to protect the system against spoofing, replay and relocation[4]. These attacks can cause the system to malfunction or even execute a malicious code instead of the normal one.
Then, we imagine that a sequence of operations is modified so that our system gives critical information in plaintext. In our work, the general processor (GPP) gives the CP the order

to cipher a frame of data. We imagine that an attacker is able to modify this sequence by probing or by integrating a malicious IP on the bus. In this case, the plaintext could be directly sent to the UART or the LCD screen. There is a need of authentication of each IP and a control of the sequences regulating our system (requests of ciphering/deciphering, global or partial reconfiguration).
Furthermore, we need to think that our system must have a multi-level security. Therefore, the Reconfiguration Processor RP has to access and modify the security enhancements described in the section IV; the attacker accessing it can overcome the system.

IV. Security improvements for the embedded system

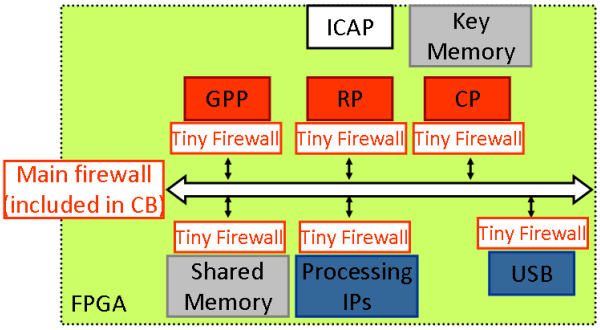

Fig. 2. Security enhancements implemented on the architecture

A. Cryptographic Bridge (CB)

Each frame of data stored in the external memory is ciphered so that if an attacker is able to probe the memory, the confidentiality of the data still exists. Therefore, we need a cryptographic mechanism to do the ciphering (respectively deciphering) of the data coming from the architecture (respectively the external memory). This CB (shown in figure 3) should be a "see-through" component, it must not affect the rest of the architecture as if it is transparent.

B. Firewalls

1) Main Firewall: The Main Firewall (MF) will be an addition of the "Crypto-Bridge" described above and another module that implements controls and monitoring on the bus to verify the sequence of operations as it is explained in the section III.

2) Tiny Firewall: The Tiny Firewall (TF) is a unique firewall implemented at the interface of each IP. Each Tiny Firewall is specific to its IP but it can also contain more generic control and monitoring operations.

C. Countermeasures

1) Isolation: If a Tiny Firewall detects that the IP is attacked, it will create a wrapper around the IP so that it will not be connected to the global bus anymore. Therefore, the malicious IP will not affect the rest of the system.

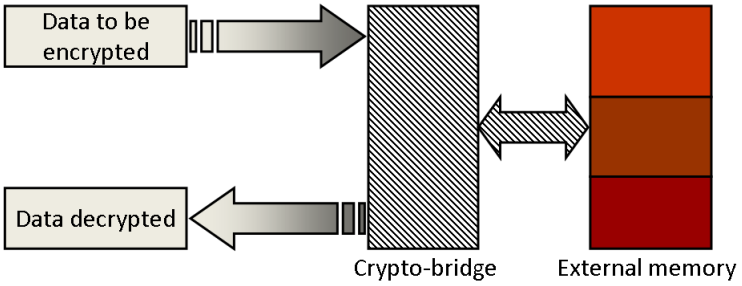

Fig. 3. Crypto-bridge

2) Dynamic reconfiguration: If a firewall (the main or a tiny one) detects an error, it may be useful to reconfigure the sub-network of firewalls, to integrate a new security policy to fit the new requirements of the system.

V. Conclusion and perspectives

We presented a solution to keep secure the communication within a MPSoC architecture where several processors compute complex treatments. We think that our work is quite innovative because it seems to be a "nearly complete" security solution : it integrates security mechanisms but also countermeasures. In related works, such as in [5] and [6], security enhancements are described but we do not see countermeasures. In other works, such as [7], the architecture is based on a NoC whereas our work is based on a bus technology.
With the association of firewalls and crypto-bridge, we think that the system make a MPSoC implemented in a reconfigurable target (FPGA) secured, strong enough to resist to a threat model and react to attacks. In future work, we foresee to study the interconnection of firewalls and the interaction with the IPs connected to the standard bus without forgetting to find a good compromise betwen surface and latency.

References

[1] P. Kocher, R. Lee, G. McGraw, and A. Raghunathan, "Security as a new dimension in embedded system design," in *Proceedings of the 41st annual conference on Design automation*. ACM New York, NY, USA, 2004, p. 753760. [Online]. Available: http://portal.acm.org/citation.cfm?id=996771

[2] Netheos, "Netheos," http://netheos.com.

[3] IronKey, "Ironkey - the world's most secure flash drive," https://www.ironkey.com/.

[4] R. Vaslin, G. Gogniat, J.-P. Diguet, E. Wanderley, R. Tessier, and W. Burleson, "A security approach for off-chip memory in embedded microprocessor systems," *Microprocessors and Microsystems*, vol. 33, no. 1, pp. 37–45, février 2009. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0141933108000823

[5] L. Fiorin, G. Palermo, S. Lukovic, and C. Silvano, "A data protection unit for NoC-based architectures," *Proceedings of the 5th IEEE/ACM international conference on Hardware/software codesign and system synthesis - CODES+ISSS '07*, p. 167, 2007. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1289816.1289858

[6] S. Chakradhar, J. Coburn, S. Ravi, and A. Raghunathan, *SECA: Security-Enhanced Communication Architecture*. New York, New York, USA: ACM Press, 2005. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1086297.1086308

[7] J. Diguet, S. Evain, R. Vaslin, G. Gogniat, and E. Juin, "SecureMemory Accesses on Networks-on-Chip," in *Proceedings of the First International Symposium on Networks-on-Chip*, vol. 57, no. 9. IEEE Computer Society, 2007, p. 223232. [Online]. Available: http://portal.acm.org/citation.cfm?id=1263074