A System Architecture for Reconfigurable Trusted Platforms

Benjamin Glas Alexander Klimm Oliver Sander Klaus Müller-Glaser Jürgen Becker
Institut für Technik der Informationsverarbeitung, Universität Karlsruhe (TH)
E-mail: {glas,klimm,sander,kmg,becker}@itiv.uni-karlsruhe.de

Abstract

For improving the security of embedded systems, trusted computing is a promising technology. For the area of microprocessors in general and personal computers in particular the Trusted Computing Group (TCG) has published detailed specifications. The resulting hardware has been available for some years. This contribution discusses the feasibility of deploying ideas from trusted computing in the domain of reconfigurable hardware, esp. FPGAs, and possible benefits and drawbacks. We give a proposal to use actually available FPGA technology to build a trusted platform on reconfigurable hardware. We also show how trusted computing can deal with partial dynamic reconfiguration while still allowing the user to fully exploit its potentials.

Keywords: Trusted computing, TPM, FPGA, reconfigurable hardware, partial dynamic reconfiguration, embedded systems.

1 Introduction

In recent years security of electronic systems has gained more and more importance in many different application domains. In contrast to the market of personal computers (PCs) and consumer electronics, security issues of embedded systems in areas like automotive electronics, industrial automation or communication devices are often unattended.

A relatively new approach to secure systems is the idea of trusted computing, brought forward by the Trusted Computing Group TCG[8, 9] in 2003. The idea is to embed trust into a computing platform by providing a dedicated hardware device capable to attest trustworthily certain properties of the platform to the user or to remote communication partners.

In the case of embedded microcontrollers the ideas of trusted computing can be adopted from the PC domain quite easily. In contrast, using reconfigurable hardware devices, which are increasingly popular in embedded systems, the task is much more complex. Here the usual separation of hard- and software is no longer applicable, so that new approaches have to be developed to achieve a trusted platform according to the TCG specification [9].

We propose a system design that allows to build a trusted platform on reconfigurable hardware. It supports strong security requirements while maintaining the benefits of FPGA technology such as partial dynamic reconfiguration.

This paper is structured as follows. In chapter 2 we introduce basic technology and related work. Chapter 3 presents our proposed architecture, chapter 4 deals with the benefits of this architecture and in chapter 5 we conclude and suggest further research.

2 Technology basics

Trusted Computing The main idea of trusted computing is to equip computer systems with a device that can be trusted by both remote communication peers and the user of the platform herself. This device - called the Trusted Platform Module (TPM) - is able to perform some basic cryptographic tasks like signing, asymmetric encryption and hashing, stores secrets like keys securely and can attest the actual state and configuration of the platform it is part of [9]. Trust in terms of the Trusted Computing Group means that an entity can be trusted *if it always behaves in the expected manner for the intended purpose* [8]. Therefore a Trusted Platform enforces a distinct behaviour as response to some clearly defined requests.

Reconfigurable hardware and security features Reconfigurable hardware offers many advantages over fixed architectures, especially in flexibility and performance. Quite a few reconfigurable HW platforms are available (i.e. Cypress' PSoC Mixed-Signal Array, CPLD, FPGA etc.). Only a few devices[1] offer AES decryption blocks otherwise they offer no security features at all. This applies especially for low-cost FPGAs. We concentrate in this paper on Xilinx FPGAs, the only devices that offer partial reconfiguration so far.

[1] i.e. StratixII/III (Altera), Virtex II/4/5 (Xilinx)

978-3-9810801-3-1/DATE08 © 2008 EDAA

Related work. Trusted Computing on reconfigurable hardware is a mostly disregarded field of work so far. In [3] Eisenbarth et al present a proposal to implement a TPM that is reconfigurable. This improves the flexibility of the approach but may contain some problems with the certification. The problem of partial dynamic reconfiguration has by our knowledge so far not been addressed at all.

Looking at security issues in embedded systems the use of trusted computing has been proposed for some special applications, for example in the automotive domain [2, 6] or in eHealth [4]. Concrete implementations as in [4] are sparse.

In the sector of personal computers the use of TC requires the support of the respective operating system or an extra security kernel as in the Turaya system (see EMSCB[2]). At the moment such support is available only for Linux (OpenTC [1], TrustedGRUB[3]) while other major operating systems only use and support parts of the functionality, for example for drive encryption.

3 A Trusted Platform based on an FPGA

In the TCG main specification [9] the term *trusted platform* is defined for a PC system. The main difference with respect to systems based on reconfigurable hardware arises in the area of attestation.

While in PC systems the hardware configuration is assumed static the use of reconfigurable hardware demands for a different approach. Both hardware and software configuration have to be measured and controlled for any attestation. Also the boot process is different, because its main task is the configuration of the hardware. After that the system is often up and running without loading software sequentially as in personal computers. So the main issue is getting a correct measurement of the implemented HW configuration and storing it as a hash. In addition to the specified platform configuration registers (PCRs) of the TPM, hardware configuration registers (HCRs) as suggested in [3] (in a slightly different context) have to be implemented.

Our understanding of an *Trusted Reconfigurable Platform* is that it implements all features of a Trusted Computing Platform while being based on reconfigurable hardware. A main concern of ours is to use existing and available components and to demand as little changes to them as possible. At the same time the use of the platform should be constrained as little as possible.

3.1 Security Threats

We base our considerations on a powerful adversary model as follows. The in memory and computation power polynomially bounded adversary $\mathcal{A}$ has full control over the platform $\mathcal{TP}$. He is able to access all external pins of the FPGA and can read and write the flash memory where the functional cores of the FPGA are loaded from. All communication between $\mathcal{TP}$ and an external communication partner - *verifier* - $\mathcal{V}$ is handled by $\mathcal{A}$, so he can deliberately delete, delay, repeat, alter, create, and forward messages.

The assumptions that we have to make is the security of the boot logic against tampering as well as the security of all interconnects of the boot logic, TPM, BootROM, and the FPGA. Ideally those components are in a single package or even on the same die with appropriate security measures.

3.2 System setup

Our proposal is based on a standard FPGA architecture by Xilinx with an attached Trust Block and a slightly modified JTAG interfacing. It is essential that any reconfiguration process is detectable by the system. One solution is to completely disable the JTAG interface and only allow reconfiguration triggered by the system itself. This limits the user to use the device only for tasks that have been certified by the designer of the initial system. To avoid this limitation we introduce our new userJTAG (uJTAG) methodology. A static section on the FPGA's fabric supplements the aforementioned external hardware. The overall layout is shown in figure 1.

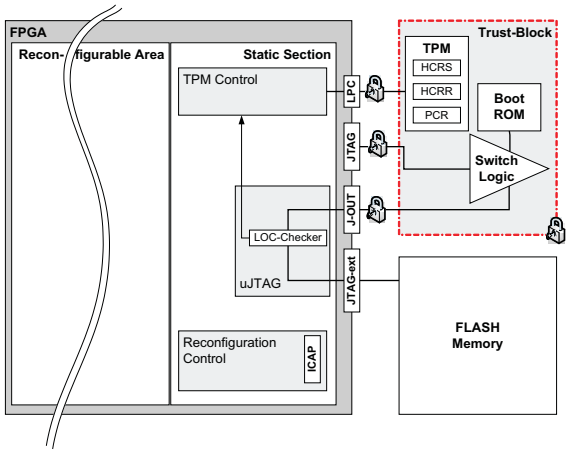

Figure 1. System overview

3.2.1 Trust-Block

The Trust-Block contains the following security relevant parts of the system, which have to be implemented in external hardware. This block and its interfaces to the FPGA have to be made secure and tamperproof according to the adversary model.

[2] for Turaya and the EMSCB project see `http://www.emscb.de`
[3] `https://prosec.trust.rub.de/trusted_grub.html`

- *A TPM* is connected via a Low Pin Count (LPC) bus interface [5] to the FPGA.
- *The secure BootROM* contains the initial bitstream for the device which has been certified and is therefore trusted. This bitstream is stored in the memory during production and must not be altered afterwards in any way by any party. It is mandatory loaded on start-up.
- *The Integrated SwitchLogic* allows configuration during power up solely from the BootROM of the platform.

3.2.2 Static Section

The Static Section of the FPGA contains the necessary components to build a trusted platform such as a TPM Control Block, FPGA-readout functionality, and the uJTAG implementation. Also the functionality for partial dynamical reconfiguration is included.

- *The TPM Control Block* implements the control of the TPM including basic cryptographic features like SHA-1 hashing and HMAC computation necessary for the communication with the external TPM.
- *Reconfiguration Control* manages the ICAP interface [10] for readout or partial reconfiguration. If no ICAP is available it can be substituted by a JCAP interface as in [7].
- *uJTAG* provides the user with an externally accessible programming port, the JTAG-ext. By the aid of the LOC-Checker block all incoming data over this port is scanned and the area of the FPGA that is targeted to be reconfigured by the incoming bitstream is determined. As soon as an illegal area[4] of the FPGA will be affected, this is signalled by the block.
- *External Memory* is used to store all bitstreams necessary for reconfiguration of the FPGA. It is not protected in any way and can be used freely.

3.3 Operation of the System

At power up of the device the initial bitstream from the BootROM is loaded onto the FPGA. The HW configuration of the static and reconfigurable area is read out, hashed and stored into the HCRS and HCRR respectively (see 3.4). The same is done with the software according to the specifications of the TCG. Subsequently the switchlogic connects the uJTAG to the FPGA's JTAG port, thus enabling the reconfiguration of the device through access to external memory via the JTAG-ext pins. All incoming bitstreams are monitored. If an illegal area of the FPGA is targeted for reconfiguration, the TPM is disabled immediately, before the reconfiguration is effectivet. This ensures that no reconfiguration is done without the TPM measuring it. Analogously the TPM is disabled if the configuration mode is altered by external signals thus enabling reconfiguration over ports other than the JTAG, e.g. the *SelectMAP* [11] mode.

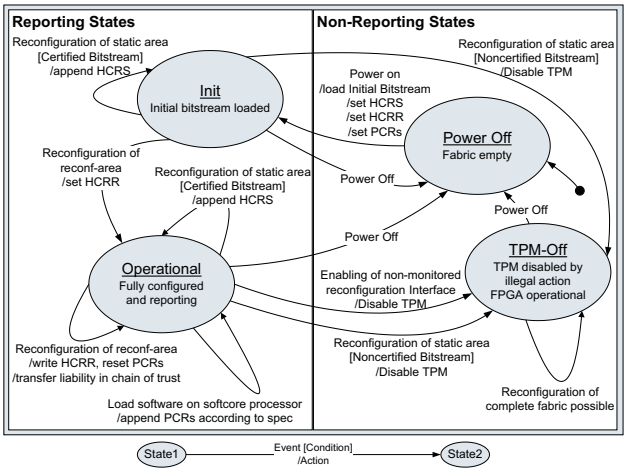

Figure 2. Security states of the system

In order to regain a trusted platform it is then mandatory to completely disable the whole system and reboot the device from the internal BootROM. This re-establishes the initial system and re-enables the TPM. The possible states of the system regarding the security and trust status are shown in figure 2. When asked for attestation the TPM sends the requested contents of its HCRR, HCRS, and appropriate PCRs signed with a secret key.

3.4 Placing Trust in Trusted Platforms

The initial, certified, and trusted bitstream acts as the Core Root of Trust for Measurement (CRTM) of the designated trusted platform as it trustworthily starts the measurement process by writing the initial configuration (as a hash) in one of the dedicated HCRs within the TPM, see paragraph 3.3. PCRs can be used for this purpose with existing TPMs.

Two HCRs are used in our system. One (HCRS) is used to store the hardware configuration of the static section as an incremental[5] hash. The other (HCRR) stores the hardware configuration of the reconfigurable area. This is done not incrementally, since we are expecting countless partial reconfiguration processes. The status of the reconfigurable area can be reported *"as is"* at any particular time. This can be handled this way because reconfiguration can erase all traces of the previous configuration. The actual state is therefore not biased, so it is also possible to leave an

[4] This means especially the static area where the TPM control and the LOC-checker is placed.

[5] In the HCRS each new configuration's hash is appended to the existing hash so that the sequence of reconfigurations remains traceable

untrusted configuration and regain a trusted state without having to reboot the whole platform, as long as the static section with the TPM control logic remains untouched. A software layer can be booted, e.g. if a softcore microprocessor on which an OS is running is implemented on the FPGA. Here the chain of trust is executed according to the PC TPM specification. Only this time it is based additionally on the hardware layer stored in the HCRs.

The bitstream is an integral part of the chain of trust because it functions as the CRTM for the PCRs. A communication partner receiving an attestation has to check first the contents of both HCRs. Only if the result is trustworthy, one is able to trust the contents of the PCRs. This exactly rebuilds the chain of trust in a PC system (see 2). If the hardware configuration is changed, the HCRS and/or HCRR are rewritten and the PCRs are reset if necessary to represent a new system booting on a changed hardware platform.

4 Benefits

With the given architecture we achieve a trusted platform on reconfigurable hardware. It is possible to report trustworthily the actual configuration of the platform taking into account the special circumstances given by the reconfigurable device. All features of the TPM remain unchanged by the reconfiguration and are usable on the reconfigurable platform as well. Additionally our approach preserves all benefits of the plain reconfigurable platform. So it is up to the user to use the security functions supported by the TPM and the control logic or to do without this functionality and use the complete fabric for her own applications. In this case the TPM is automatically disabled.

The proposed architecture can be implemented with commercially available technology. Only the TPM TrustBlock and its interfacing needs special handling by the platform manufacturer to adhere to the security assumptions given in 3.1. A reset of the PCRs is not yet implemented in currently purchasable TPMs. Implementations not needing this feature can be based on available hardware already.

Our approach combines the security assertions of a trusted platform with the flexibility of a reconfigurable device. Especially in embedded areas where devices are often under less direct control than a desktop PC it is desirable to have security mechanisms that enforce remotely transmitted policies automatically.

5 Conclusion and further research

In this contribution we have described how the core ideas and functionalities of trusted computing (TC) can be adapted to work on top of reconfigurable hardware. While offering a solution based on actual available components we have to point out that hardwired integration of security functions in reconfigurable devices could greatly advance the security properties and guarantee an even higher level of trust in the security of the platform.

Applying our trusted computing approach to embedded systems shows great potential to solve a whole class of actual security problems. These include securing intellectual property, protecting safety relevant functions, and getting trustworthy information from remote sensors, and actuators while still keeping all the flexibility of a configurable device. Further research in this area is suggested.

Timing behaviour of the security features for instance is a very important issue when thinking about real-time applications common in embedded systems like in the automotive area. Also the cooperation and interaction of ECUs, sensors and actors in distributed embedded systems raise a lot of questions concerning trust and security of the network as a whole. In the future it would be beneficial to implement a hardwired TPM into the FPGA itself to avoid eavesdropping on external bus connections or even disconnecting them. A wide range of questions is still to be answered and research is in progress to tackle related problems.

References

[1] Opentc: Executive summary of the first project year. available electronically at `www.opentc.net`, 2007.

[2] A. Adelsbach, U. Huber, and A.-R. Sadeghi. Secure software delivery and installation in embedded systems. In R. H. Deng, editor, *ISPEC 2005*, volume 3439 of *LNCS*, pages 255–267. Springer, 2005.

[3] T. Eisenbarth, T. Güneysu, C. Paar, A. Sadeghi, D. Schellekens, and M. Wolf. Reconfigurable trusted computing in hardware. In *STC '07*.

[4] U. Grossmann, E. Berkhan, L. Jatoba, J. Ottenbacher, W. Stork, and K. D. Mueller-Glaser. Security for mobile low power nodes in a personal area network by means of trusted platform modules. In *ESAS*, 2007. to appear.

[5] Intel Corporation. Intel® low pin count (lpc), interface specification, 2002.

[6] K. Lembke, A.-R. Sadeghi, and C. Stüble. An open approach for designing secure electronic immobilizers. In *ISPEC*, 2005.

[7] K. Paulsson, M. Hübner, and J. Becker. On-line optimization of fpga power-dissipation by exploiting run-time adaption of communication primitives. In *SBCCI, Brasil*, 2006.

[8] S. Pearson, editor. *Trusted Computing Platforms*. Prentice Hall, 2003.

[9] Trusted Computing Group. Tpm specifications. Available at `www.trustedcomputinggroup.org`, 2003.

[10] M. Ullmann, M. Hubner, G. Grimm, and J. Becker. An fpga run-time system for dynamical on-demand reconfiguration. *ipdps*, 04:135a, 2004.

[11] Xilinx. Spartan-3 fpga family complete data sheet ds099, April 2006.