Secure Computing Environment for SoCs and FPGAs

by Ryan Kenny, Senior Strategic Marketing Manager, Ting Lu, Senior Security Architect, and Rod Frazer, Field Applications Engineer

WP-01235-1.0 White Paper

Programmable logic devices in the future will provide rich systems of programmable fabric, hardened processors and accelerators, and a wider range of high speed serial interface functions. The configuration process for these devices therefore becomes more complex, and the need for secure, authenticated, and protected boot increases. In this paper, Altera introduces the configuration and secure computing strategy for these devices beginning with the Arria® 10 SoC, and focuses on its strong authentication and boot order selection flexibility.

Introduction

The proliferation of embedded processors throughout industrial and commercial appliances has been discussed in thousands of publications over the last twenty years. These publications all agree on a couple of key implications: there is an explosion of opportunity in work and leisure efficiencies through embedded technology, and these new technologies create an unprecedented level of exposure in everything from phones to automobiles to industrial equipment. The inclusion of processors brings potential vulnerabilities and concerns associated with debug, device boot, and protocol synchronization that need to be addressed with strong security features and capabilities. The endpoints become the newest target of vulnerabilities in the Internet of Things (IoTs), but the routing backbone of the internet remains the area to invest in system-wide security awareness.

Altera's midrange Arria 10 SoC and FPGA products provide essential functionality at the backbone of wireless and wireline infrastructure and data centers. This class of equipment is both a target for persistent security threats and a potential critical point of vulnerability by unauthenticated users. This infrastructure can also provide strong security and authentication frameworks, with the added flexibility and scalability of ARM®-based embedded processing to manage the threats created by the explosion of new endpoints. As a result, secure boot has become a key security requirement and a design concern.

SoC FPGAs are modern FPGAs in that they provide a more intelligent level of security management and design authentication. With secure boot, the embedded processor will only boot and run authenticated software before the FPGA has even been configured (if HPS is booted first). If an intrusion is detected then the software can respond intelligently by not only detecting, capturing, and logging the intrusion details but also by executing a graceful system shutdown. FPGA fabric will continue to provide the capability of high-speed pipelined packet communication, but with an integrated control plane processor whose software can be updated and digitally signed to include new threat signatures and address new attack vectors.

101 Innovation Drive
San Jose, CA 95134
www.altera.com

These trends machine-to-machine connections have guided the security feature selection and security architectures of the Altera's SoC products. Both the security of the initial boot process and the trusted execution of a SoC design—the Root of Trust and the Transitive Trust of the design—now requires an integrated strategy and process to enable a secure computing environment. This is needed to design, test, and certify that a complex SoC design can survive both malicious and unintended changes in environments with sensitive data requirements, or very high safety and reliability requirements.

Altera's Approach to Secure Computing in Arria 10 SoCs and FPGAs

The Arria 10 SoC provides FPGA and SoC designers the only FPGA programmable logic product with hardened embedded processors available at the 20 nm process node. In addition, Arria 10 is the only programmable logic device with a comprehensive secure boot capability, which includes a true Root of Trust functionality with hierarchical public key infrastructure (PKI) support, the high-strength integrated elliptic curve cryptography for boot security with true Root of Trust support, and the ability to select boot order between FPGA fabric and hard processor system (HPS), which enables software to implement a tailored secure boot sequence involving intelligent intrusion detection and graceful system shutdown.

Altera is introducing the most comprehensive secure boot capability for SoC FPGA applications. This capability is divided at a high level into three major components.

- First, Arria 10 SoCs feature Elliptic Curve Digital Signature Algorithm (ECDSA)-based authentication for all boot load stages of software in the hard processor system. Elliptic curve is widely considered the strongest published algorithm for asymmetric key authentication; in addition, its asymmetric nature allows the user to designate whether the public or private key (and therefore Root of Trust) is resident in the SoC.
- Second, Arria 10 SoCs offer the ability for designers to choose the boot order of the FPGA logic versus the ARM-based HPS. When the HPS boots first, the processor is able to provide intelligent tailored intrusion detection and graceful system shutdown. When the FPGA is configured first, custom logic is able to provide a robust, highly reliable multistep authentication process for the subsequent processor boot sequence, reducing the attack surface on the system.
- Third, the Arria 10 SoCs and FPGAs feature new security features for addressing anti-tamper and key protection requirements, such as cryptographic resistance to differential power analysis attacks, integrated sensors for voltage and temperature, and user designated and accessible fuses.

Secure Boot Strength

The Arria 10 SoC implements an ECDSA with a key length of 256 bits, assisted by hard logic accelerators to perform the signature checking. ECDSA provides some of the highest cryptographic strength relative to key length available in asymmetric cryptographic technology. The algorithm is computationally intensive but has been optimized in 20 nm hard logic for rapid signature checking.

Secure Boot Flexibility

The Arria 10 SoC provides either One-Time-Programmable (OTP) fuse enforced, or pin selectable boot order between the FPGA bit stream and HPS ARM code. This provides designers maximum flexibility in either achieving an authenticated running state as early as possible in software, or establishing a deterministic state in FPGA logic to control the inputs and outputs of the HPS. No other integrated SoC product provides this boot order flexibility.

Additionally, the ECDSA signature checking functions in the HPS subsystem allow the user to select whether the private key Root of Trust is resident in the Arria 10 SoC, or in a protected device or computer somewhere else either in-system or off-system. This allows the designer to focus security and anti-tamper mechanisms wherever desired in system.

Anti-Tamper and DPA Resistant Support

Arria 10 FPGAs and SoCs provide all new advanced security features to enhance both the secure boot and transitive trust of your system. These features include decryption engine designs inherently resistance to known differential power analysis (DPA) attacks, protecting keys and design. Monitors and sensors are available on the control plane to both the FPGA fabric and the HPS to provide environmental status and response to other known attacks. The Arria 10 FPGA and SoC also has dozens of security register and fuse state settings that allow the designer to dynamically or permanently define the accessible ports and attack surfaces of the system. For the first time, this product also provides user fuses for use as immutable control logic or permanent records of tamper events.

Design Considerations for Secure Computing Environment for SoCs and FPGAs

The first step in architecting a robust secure computing environment for a SoC system is to understand the requirements for security and secure computing environments from the perspective of existing standards bodies. The second step is to analyze the vulnerabilities and attack vectors for the given security environment. The third step is to consider the approach to specifying Root of Trust, boot order, and transitive trust throughout the design process, including updates to design.

Standards for Security and Secure Computing Requirements

The basic sources for secure computing requirements today come from a series of standards published by the National Institute for Standards and Technology (NIST), and a recently released Framework for Cyber Security also published by NIST. Much of the evolution of these standards and frameworks in the last several years are guided by published vulnerabilities in embedded and networking systems—a large number of them enabled by a lack authentication from the user, user command, or set of instruction code.

The Cyber Security Framework in turn cites multiple security standards and best practices that generally divide security requirements into a simple trio of needs: Confidentiality, Integrity, and Availability. Confidentiality is the protection of data from examination or copying. Integrity is the protection of data from intentional or non-intentional modification or tampering. Availability is assured communication access between various elements within a system in order to meet performance requirements.

The secure boot schemes described here focus primarily on the Integrity component by providing the ability to authenticate both the source, and unaltered state, of coded instructions. Secure boot also provides options for protecting the Confidentiality and Availability of processer code and FPGA bit stream information as well. The methods chosen for these protections are based on a vulnerability analysis.

System Level Security Requirements

Protecting the Confidentiality, Integrity, and Availability of SoC software instructions is not a task undertaken in isolation from the overall system architecture, however. The fact that device instructions lie in off-chip flash storage, that power and clock sourcing are components of device security, and the fact that Arria 10 SoC provides you the option to designate the Root of Trust off-chip all make security a system consideration, and a process exercised during system architecture design.

Vulnerability Analysis

Security requirements for a system derive from the vulnerability analysis of the system architecture. This analysis should consider not only how the system operates and each component interacts with one another, but how the system itself and its firmware are updated and regression tested.

Vulnerabilities Derive from Attack Vectors

Arria 10 device security requirements derive not only from vulnerabilities, but from an explicit statement of known attack vectors. These attack vectors include attacks during operation, during product update and maintenance, and from boot operation.

Because Arria 10 device customers represent a variety of industries including military, broadcast, communications infrastructure, utilities, smart grid, and so on, there are many different attack vectors integrated into Altera SoC security architectures. Some examples are shown in Table 1.

Table 1. Map of Some Secure Boot Attack Vectors and Security Features

Attack Vector	Security Feature				
	ECDSA	Bit Stream Keyed-Hash Authentication Code (HMAC)	Bit Stream Advanced Encryption System (AES)	DPA Resist	Security Fuses
	HPS	FPGA	FPGA	Both	Both
Boot Code Substitution	Strong Authentication				Require Authentication
Boot Code Alteration	Strong Authentication	Strong Authentication			Safe Remote Update Mode
DPA Attack	ECDSA Immune			Inherently DPA Resistant	
Key Imaging	Public Key Only		Keys Uniquely Scrambled		
Copy Bit Stream			Option to Encrypt (AES)		Require Encryption
Read-back Bit Stream					Secure JTAG

The attack types here can be researched in open literature via university research and 'blackhat' hacking conferences. Attacks can include substituting or altering the contents of flash storage, to include ARM boot code or FPGA bit streams. Also documented are methods to use DPA to read keys or key variables from energy side channels in the device. Other common attacks on semiconductors include simply copying flash contents, attempting to access bit stream configuration, or on-chip keys through imaging technologies or JTAG probing.

Because secure boot is only one aspect of a secure computing environment, Altera's approach to security focuses on providing the user several security register and fuse setting options for securing the entire device. These options provide the tools to enable a secure computing environment tailored to your use case, rather than prescribed by a limited and constrained boot flow.

One of the first principles of configurable secure boot from Arria 10 SoC products is the choice to enable the primary boot order as either the FPGA fabric first or the HPS first. Beginning with the Arria 10 SoC, users will be able to force this boot order through fuse settings or pin toggle selection. FPGA or HPS boot order has several different advantages and elements of flexibility, including the ability to add custom secondary authentication schemes, contingency/trap/failure modes, and responses to include erasure of sensitive data.

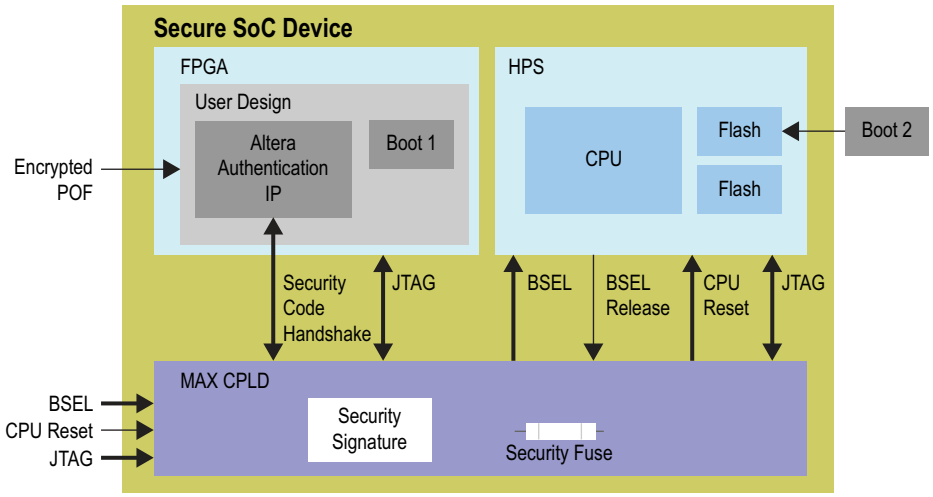
The second important principle of configurable secure boot from Altera is the ability to select the degree of security through the selection of which elements of the design are authenticated and encrypted. With several different options in both bit stream and processor code, use cases exist that prioritize confidentiality and integrity, or availability (including boot load time and reliability), or a tradeoff between them. Examples of this tradeoff include the division of bit stream information into LUT configuration, I/O configuration, and embedded memory initialized, which can each be protected or non-protected based on sensitivity.

Implementation

Identifying a Root of Trust

Altera products provide you a set of choices in selecting the Root of Trust in your secure system. Some choices include using the SoC as a Root of Trust (OTP fuses and ROM, with advantages and disadvantages), an external trusted product like a Trusted Platform Module (TPM), and even a system boot manager with integrated non-volatile memory like the MAX® 10 FPGA or MAX CPLD. Alternately, any trusted processor in your system can perform this function if booted and operating prior to the Arria 10 SoC. The Root of Trust may not even reside in the system itself, but in a configuration management center where all code changes are authenticated. This root will provide the services of a certificate authority for boot code authentication.

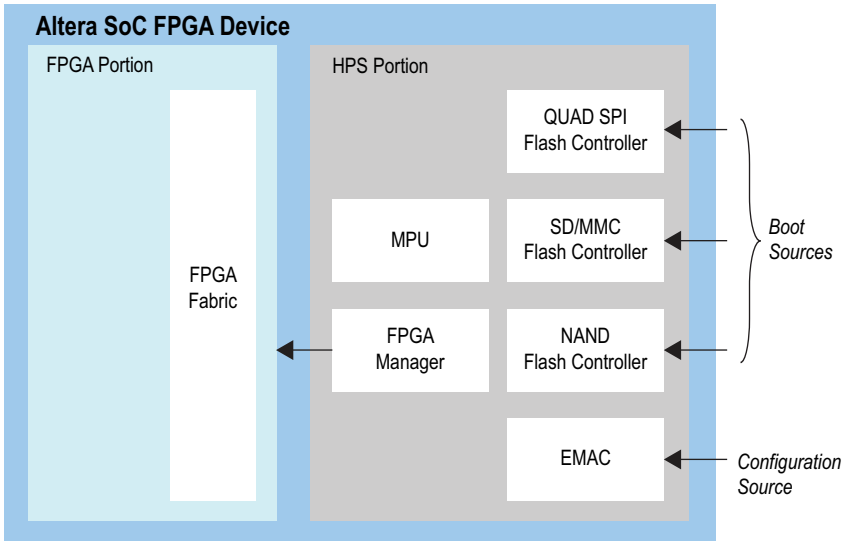
Figure 1. Example of a MAX 10 FPGA or MAX CPLD as the Root of Trust in a System



Initialization: Secure Boot and Boot Code Authentication

With a Root of Trust identified in the system, the secure boot mechanism can be defined and implemented in the Arria 10 SoC.

Figure 2. Arria 10 SoC Boot Separates FPGA and HPS Boot Processes, and Allows User to Select Order

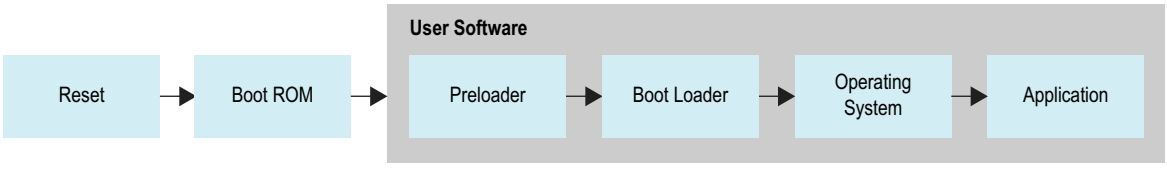


If the user selects the FPGA to boot first, then the FPGA bit stream loads into the device where it is authenticated and decrypted using a single on-chip AES and Secure Hash Algorithm (SHA) or HMAC with 256 bit key. The user selects which portions of the bit stream are both encrypted and authenticated; this can be all or only a portion of the configuration bit file. Only after the FPGA loads will the HPS subsystem be released from reset and enabled for boot.

The remainder of this secure boot description refers to the HPS subsystem, which boots either before or after the FPGA fabric is configured, enabled, and running. The purpose of the HPS operating system and user code boot process is the same as the FPGA fabric: to authenticate, and if applicable, to decrypt the user operating code.

Secure boot operation takes place in stages, with each stage of the boot load process finishing with a check of the authenticity and integrity of the boot code. This digital signature check is performed using either the public or private key stored in the SoC (corresponding to an internal or external Root of Trust). The next boot stage is not enabled unless the current stage passes the authenticity verification check. The signature algorithm options in the Arria 10 SoC include ECDSA256, and/or a SHA256. If confidentiality is also a concern, encryption or decryption of the boot code is then performed with an AES256 engine.

Figure 3. Secure Boot Occurs in Discretely Authenticated and Authorized Steps



Secure Computing: After the Boot Process

The secure computing environment is defined by the systems architect and engineer. Altera products provide a host of capabilities for trusted code execution, as well as physical integrity checks to ensure a secure computing environment. A host of advanced FPGA and SoC anti-tamper features, use of the ARM Trustzone infrastructure, and the additional multistage and multikey capabilities of Agile Partial Reconfiguration (partial reconfiguration regions protected by different keys) provide tools to ensure the run-time environment is as secure as the boot environment. By always forcing authentication on any new signed executable or data set, continued validation and authentication is maintained. Details on these capabilities will be provided in Arria 10 SoC secure boot documentation.

Implementation in the Arria 10 SoC

The techniques above describe the secure boot processes and options as implemented in the Arria 10 SoC. In addition to the listing and descriptions of the secure boot features and options for Arria 10, this section will provide a recommendation for a typical use case of the SoC with primary requirements in authentication and integrity, but not confidentiality. This addresses a use case discussed in the introduction of this paper, where internet routing infrastructure may not contain highly proprietary or protected embedded code, but security relies on the integrity or the code, or its protection from unauthorized modification.

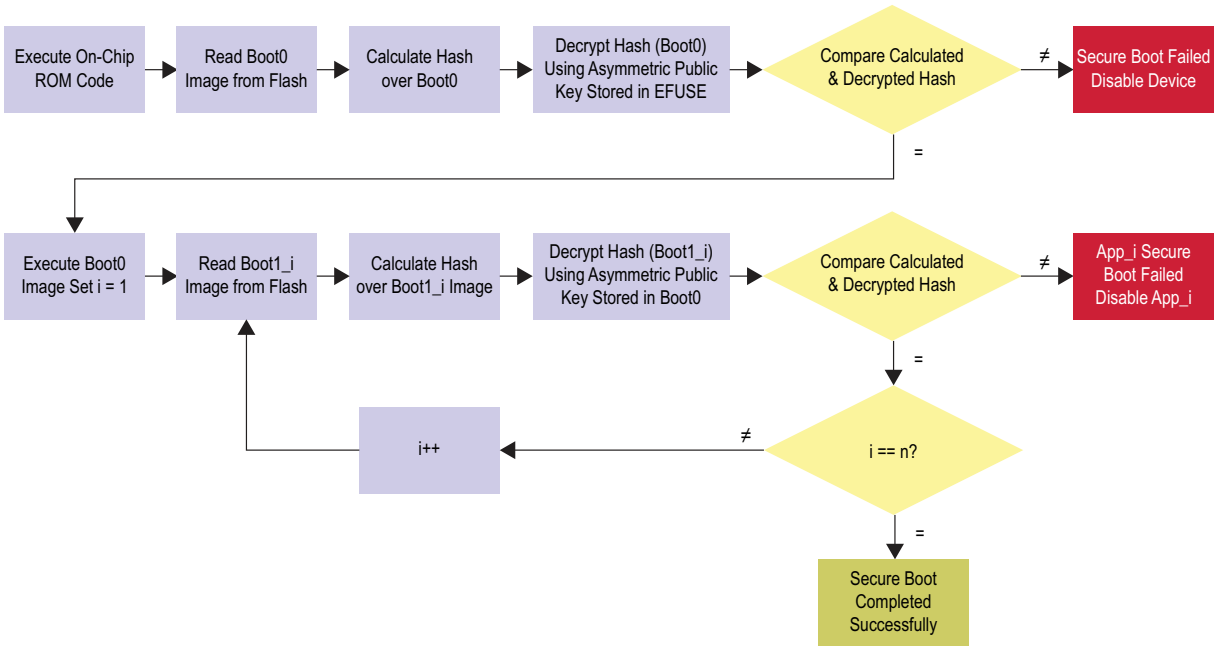
Recommended Boot Architecture for Arria 10 SoC

When using the Arria 10 SoC in a wireline communication system or industrial system with safety and high-reliability applications, ensuring the integrity of boot and application code is the paramount consideration in architecture design. In addition, the equipment for these use cases may not always be physically accessible, so the ability to update firmware or software with new authentication signatures is an important design factor.

For this boot architecture, the HPS application code will be broken down into n number of discrete images in boot flash. The decision on how to break up with application should be guided by how firmware upgrades will be performed in the future. If there will be separate updates to operating systems, drivers, and different pieces of application code, then these can be created into separate software images with their own encrypted signature or hash values that can be separately updated.

The resulting boot process is illustrated below, terminating in successful boot of the HPS followed by loading, authenticating, and decrypting the FPGA bit stream.

Figure 4. Step-by-Step Boot Process for the Arria 10 HPS Subsystem Before Loading Bit Stream



Another benefit of this approach is that an additional layer of authentication can be provided to the bit file for the FPGA load. Executable instructions loaded into the HPS user application can manage the load of the FPGA bit file, and perform another hash authentication of the bit file in addition to the HMAC fields provided within the bit file itself.

Using the SoC to Authenticate Other Devices

Once an entire SoC system is loaded and authenticated, it can then be used to authenticate other devices. It can do this using its own one-time programmable fuses as a root key, or can load user application code with encrypted hash code signatures used to authenticate algorithms on another device. One of the advantages of using the Arria 10 SoC for this function is the hardened math functions for the SHA and ECDSA signature algorithms, which can be accessed through a security control mailbox in the HPS subsystem. This enables fast signature checking as a system boot manager.

Conclusion

Security and reconfigurability are often at odds in embedded system components, and some of the first securely bootable processors on the market have lacked the flexibility necessary to tailor the security levels and goals to system security objectives.

With the new Generation 10 SoCs from Altera, secure boot is both an integrated capability and a flexible tool set for designing to and meeting your system security requirements. Altera provides and accelerates the use of powerful ECDSA signature algorithms, allows user selection of device boot order, offers a full set of options for the selection of the Root of Trust, and gives infinite flexibility to divide application code into updateable partitions or images. All of this allows your system to implement the right amount of security with minimal effort and qualification.

Further Information

- Federal Information Processing Standard (FIPS) 140-2, Security Requirements for Cryptographic Modules
  csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf
- Framework for Improving Critical Cyber Security Infrastructure, 12 February 2014
  www.nist.gov/cyberframework/upload/cybersecurity-framework-021214.pdf
- Federal Information Processing Standard (FIPS) 180-4, Secure Hashing Algorithms, 7 April 2012
  csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf
- Federal Information Processing Standard (FIPS) 186-4, Digital Signature Standard, July 2013
  nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf
- Handbook: *Arria 10 Hard Processor Subsystem Handbook*
  www.altera.com/literature/hb/arria-10/a10_5v4.pdf
- Arria 10 for Secure Communication Systems
  www.altera.com/literature/po/ss-soc-secure-comms.pdf
- Comparing Altera SoC Device Family Features
  www.altera.com/literature/hb/soc-fpga/UF-01005-2014.01.15.pdf
- White Paper: *Architecture Matters: Choosing the Right SoC for Your Application*
  www.altera.com/literature/wp/wp-01202-embedded-system-soc-design-considerations.pdf
- Arria 10 SoC HPS Release Notes
  www.altera.com/literature/rn/a10_hps_rn.pdf

Acknowledgements

- Ryan Kenny, Senior Strategic Marketing Manager, Military Business Unit, Altera Corporation
- Ting Lu, Senior Security Architect, Product Design, Altera Corporation
- Rod Frazer, Field Applications Engineer, Embedded Specialist, Altera Corporation

Document Revision History

Table 2 shows the revision history for this document.

Table 2. Document Revision History

Date	Version	Changes
October 2014	1.0	Initial release.