Microsemi®

# Developing Safety Critical Applications that Meet IEC 61508 Standards

## Using System-on-Chip Devices with Embedded ARM Cortex-M3 and FPGA

March 2013

# Table of Contents

Abstract . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

Market Trends Driving the Use of Safety Standards . . . . . . . . . . . . . . . . . . . . . . . . 4

IEC 61508 Considerations When Using an FPGA . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

Failure in Time (FIT) Rates . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

Security Concerns for Networked Systems . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

The Security Life Cycle . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

Design Security Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

Data Security Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

Redundancy Creates Fault-Tolerant Systems . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

Example Design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

SmartFusion2 Architecture Description . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

Motor Control Design Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11

SmartFusion2 Implementation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11

Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13

References . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13

# Abstract

Industrial safety systems encompass complex manufacturing infrastructures and processes with many different levels of identified safety integrity requirements. This session will first provide an overview of numerous safety standards with particular focus on the IEC 61508 Functional Safety of Electrical/ Electronic/Programmable Electronic Safety-related Systems. Associated security concerns, in particular for networked embedded systems, will also be explored. A case study of a deployed Safety Level 3 system will be described. A design and implementation methodology will be used to create redundant, dissimilar processes to dramatically improve reliability. A configurable system-on-chip that includes an embedded ARM® Cortex™-M3 processor and FPGA logic will be used as the target implementation device.

# Introduction

All systems have the possibility of failing. Even the most robust systems can suffer a failure since it is impossible to design a system with an absolute zero failure rate. Thus each application should be designed with a target acceptable failure rate level and in many cases it is appropriate to have a design goal to provide a significantly long time between failures. For example, in a US nuclear power plant the goal is a single failure in 110,000 years. Many power plants exceed this to a single failure in one million years with the target of a single failure in ten million years. This example is an extreme case, but every application where significant loss of life or damage/destruction of capital equipment could result should be designed with high reliability in mind. Therefore, the tolerable failure rates/levels can vary dramatically per application and must be matched with the potential for significant loss. With this in mind, Safety Integrity Levels (SILs) have been categorized to quantify various levels of risk. Table 1 shows just a few of the many international standards that address various risk levels and identify appropriate reliability requirements.

*Table 1:* **Example of Functional Safety Specification per Market**

| **IEC 61508** | **Functional Safety in Industrial Equipment** |
|---|---|
| DO-178B/DO-254 | Functional Safety in Avionics |
| IEC 6060 | Functional Safety in Medical Equipment |
| EN 50128/9 | Railway Application – software for railway control and protection |
| ISO 26262 | Functional Safety in road vehicles |

IEC 61508 was developed by the International Electrotechnical Commission (IEC) and applies to functional safety in industrial equipment with certification by agencies such as the Technischer Überwachungs-Verein (TUV; translated as Technical Inspection Association). The specification originally applied solely at the system level but has also been applied to product and components by addressing electrical, electronic, and programmable electronics for both hardware and software.

*Table 2:* **IEC 61508 Safety Integrity Levels**

| SIL Level | Probability of Failure | Consequence | Application Example |
|---|---|---|---|
| 4 | 1 failure in 110,000 years | Potential for fatalities in the community | Nuclear power plant control |
| 3 | 1 failure in 11,100 years | Potential for multiple on-site fatalities | Hazardous area laser curtain sensors |
| 2 | 1 failure in 1,100 years | Potential for major on-site injuries or fatalities | Hazardous liquid flow meter |
| 1 | 1 failure in 110 years | Potential for minor on-site injuries | Thermal meter |

## Market Trends Driving the Use of Safety Standards

The global economic conditions and rise of emerging competition of the last few years have forced many companies to redesign products for differentiation as a way to better address emerging customer demands and avoid overlap with competitive offerings. Application areas such as functional safety are on the rise to protect life and expensive operating costs. Security concerns, in particular for network enabled embedded systems, are also increasing as the number of malicious attacks on these systems has significantly increased. This has driven manufacturers to include additional levels of safety and reliability in their network enabled embedded designs, using established international standards.

According to IMS Research, it is estimated that the world discrete machine safety component revenues grew by 17.5% in 2011 and have now exceeded $2B USD. This can be attributed to strong recovery from the 2009 recession and updates to the international and European machine safety standards.

## IEC 61508 Considerations When Using an FPGA

FPGAs are increasingly being used in safety applications. As such there are general requirements for usage as addressed in the IEC61508-2, Annex F.2. These are focused on avoidance of systematic failures of FPGAs and include documentation for different phases of design including:

1. Design Entry
    - Structured description in Hardware Description Language (such as Verilog or VHDL) with proven simulators
    - Functional testing of top level and submodules
    - Modular design
    - Avoidance of asynchronous data paths
    - Design for testability; in the case of SIL 3&4 >99%
    - Documentation of simulation results
    - Application and validation of soft IP
2. Synthesis
    - Internal consistency checks
    - Verification of the gate netlist against a behavioral model through simulation
    - Documentation of synthesis constraints, results, tools
    - Usage of proven in use synthesis tools
3. Placement, routing, layout generation
    - Application of validated hard cores. In the case of a cSoC, the embedded MCU is a standard ARM Cortex-M3 processor.
    - Simulation of gate netlist against behavioral model
    - Additional timing slack >20% for process technologies in use for less than 3 years. The 130 nm flash technology has been around for over 3 years, thus this is a non-issue.
4. Manufacturing
    - Application of a proven in use process technology, device series
    - Quality management systems, for example ISO 9000
    - Final verification and validation of the prototype in the system
    - Burn-in testing

Depending on the target SIL level, each line item has an associated recommendation level. All items that are highly recommended for implementation should be applied. The above list is just a sampling of requirements and these are dependent on design entry methods, tools, etc. As noted, the complete list can be found in IEC61508-2, Annex F.2. For instance, in the Design Entry method, the first item listed in Annex F.2 is Structured Description, which is listed as highly recommended. This is the case for most SIL3 & 4 applications.

## Failure in Time (FIT) Rates

The single event upset (SEU) phenomenon was first discovered in 1979 by Intel and Bell Labs as failures in DRAMs. The SEU is attributed to stray alpha particles or neutrons changing the configuration memory cell. In 1999, Sun Microsystems noticed errors in cached SRAMs for mission critical servers. In space and aviation applications the operational altitudes have a higher neutron flux but the SEU phenomenon is increasingly becoming a concern at sea level as well. The continuous drive to smaller node processes reduces the charge at each base cell, thus the likelihood of SEU errors increases.

Utilization of FPGAs in safety critical applications can be compelling due to their integrated system level features, configurability, and in-system upgradability to name a few reasons (but not all technologies are made the same so "your mileage may vary," depending on your selected FPGA device). In the case of a flash-based FPGA, the base architecture is immune to single event upsets due to the high voltage charge level required to reprogram the base cell. As mentioned, SRAM-based architectures are susceptible to SEU, thus additional design logic, up to three times the original design size, is required to identify and capture the SEU impact before putting the system into a known safe state or reconfiguring the system back into the original state.

## Security Concerns for Networked Systems

Standard operating systems have been targets for hackers using computer viruses and worms for many years. Recently these types of attacks have spread to networked embedded systems. For example, the recent attacks by the highly sophisticated Stuxnet computer worm not only spread through Microsoft Windows® systems but also directly affected Siemens industrial programmable logic controllers (PLCs), a relatively simple type of networked embedded system. As the number of these systems and their sophistication levels are growing dramatically, it is more likely they will be targeted for more and more attacks. In fact, the number of existing embedded systems is already significantly larger than the number of standard PCs, their dramatic growth will only make them even more obvious targets for malicious attacks.

Even more importantly, the very features that make embedded networked systems desirable to customers and manufacturers can create vulnerabilities. For example, remote diagnostics is a valuable feature that simplifies the gathering of quality of service metrics and can be used to predict system wear and degraded performance in common systems like power supplies or in mass storage subsystems. Unfortunately, the networked system that is used for remote diagnostics can also potentially be hacked to allow attackers to gather important information on system operating characteristics. This might also allow an intruder access to even more sensitive information if a software bug or operating system back door is available for exploitation.

Typically some amount of remote control is available along with remote diagnostics. This might be as simple as setting trigger levels for out-of-bounds operation or enabling or disabling automatic sensors, but if an attacker can adjust these settings in a malicious manner, the system might be allowed to operate outside of the safety zone, with potentially damaging results. Chemical processing or energy distribution systems could be particularly vulnerable to these types of simple malicious attacks.

Finally, an even more valuable but potentially very vulnerable feature is that of remote firmware updates. This feature allows the embedded processor’s code (or even FPGA logic) to be revised, perhaps as a service upgrade or to install new features, over the network without an on-site operator present. This is a significant savings if an in-person service call can be avoided. When there are thousands of embedded systems deployed, it may be the ONLY way to do updates in a reasonable manner. A malicious user could exploit this feature and in the worst case the entire program could be hijacked, creating a potentially catastrophic result.

As embedded networked systems become more pervasive, a wide variety of application can be vulnerable to malicious attacks. Some of the most obvious applications where significant loss of revenue could result are the communications grid (cell towers, call switching networks, etc.), the energy grid (smart metering systems, power switching systems, power plants and alternative energy systems such as solar or wind farms), chemical processing and transport (gas and oil wells, pipelines, refineries and shipping systems), and even emergency services networks.

In addition to these obviously critical systems there are other systems that are also vulnerable, perhaps to significant types of losses. For example, networked embedded systems used for the transfer or storage of personal information (financial and medical, for example) can be targeted for malicious attacks and the loss may not even be clear until the stolen information is used. Consumer medical devices for monitoring patient health could also be vulnerable (perhaps not as catastrophically, however, as when a pacemaker was hijacked on a recent episode of the TV show *Homeland*). Architects and designers must be increasingly aware of the types of malicious activity to which networked embedded equipment can be vulnerable and take appropriate action. Fortunately, there are existing guidelines that can be used to better target appropriate security levels.

For example, numerous international standards are addressing security by including requirements to address these types of threats. For instance, in the IEC 61508, Part 1, page 8, “the standard demands that any malicious and unauthorized actions are to be examined during the hazard and risk analysis. The application scope of this analysis includes all relevant phases of the security life cycle”. The security life cycle is an important concept and can help ensure a secure embedded system deployment.

# The Security Life Cycle

In order to improve the security of an embedded system, it is not sufficient to consider only the system as installed in the field. It is critical to secure the product even in the design, development, manufacturing and end-of-life phases as well as during deployment. Clearly a design that can be easily copied, reverse engineered or salvaged from a recycling effort is not secure, no mater what security precautions are taken during field deployment. Security must be a key consideration during the entire life cycle of the system. A useful way to consider the entire life cycle of the design is to group security concerns into two main classes: design security and data security.

## Design Security Requirements

The goal of design security is to ensure that the owner's hard work and valuable intellectual property (IP) is protected and intact at all times. Thus protection of the design must be considered during all phases of the product lifetime. This includes providing protection against reverse engineering, tampering, overbuilding, counterfeiting, cloning and the protection of any IP used in the design. In many cases, just understanding the IP used to secure the design may be enough to increase the likelihood of a malicious attack being successful.

There are a variety of design security techniques that are available to the system architect. Perhaps one of the easiest techniques is to store all configuration data and processor code on-chip using a nonvolatile memory, as done by SmartFusion®2 devices. This avoids the possibility of a malicious attacking agent that inspects the configuration data used to load the FPGA and/or MCU using off-chip storage. This can effectively prevent many types of attacks that attempt to reverse engineer, tamper, overbuild or clone a design. More advanced security features are available in SmartFusion2 devices to further improve design security. Key examples are secure programming during manufacturing, secure in-system programming and protection against reverse engineering.

## Secure Manufacturing

SmartFusion2 bitstreams are always encrypted using AES with 256-bit keys and fully authenticated with a 256-bit tag. This provides protection against device cloning and system-level overbuilding. Furthermore, the included bitstream validation service (usable without committing to programming the device), makes for safer upgrades using data which may have been corrupted (accidentally or maliciously) in transit to a system in the field. Automated secure key management and programming modes are available to secure the manufacturing of SmartFusion2-based designs. SmartFusion2 devices can be programmed in-house with an AES key, and then shipped to a contract manufacturer for final programming. The contract manufacturer programs the device with your AES-encrypted bitstream; hence only devices with the same AES decryption key will be programmed.

Since even the initial injection of user bitstream keys is done in encrypted form in SmartFusion2 devices, all programming configuration steps can be performed in a non-trusted manufacturing environment, thus eliminating requirements for an expensive trusted facility or the need to physically transfer devices between trusted and normal manufacturing locations. No longer do keys appear in plaintext or protected only by weak obfuscation techniques. The capability to load user keys using strong cryptography and best-in-class key management practices at every step is unique to SmartFusion2 devices in the entire FPGA industry.

## Secure Remote Programming

SmartFusion2 devices can be reprogrammed remotely using an AES encrypted programming file for easy and secure field upgrades. Intercepting the encrypted configuration bitstream is useless. An appropriate AES decryption key is required in order for an encrypted configuration bitstream to work. Bitstream files are not only encrypted for confidentiality, but also fully authenticated, providing proof of integrity and heritage. While SmartFusion2 devices can be reprogrammed remotely, if desired, they can also prevent denial of service (DOS) attacks by enabling remote reprogramming only to key holders or by disabling the remote programming capability completely (lock permanently).

## Protection Against Reverse Engineering

A number of additional factors complicate attempts to reverse engineer a SmartFusion2 device. In order to determine the state of any given flash element, the microscopic size and sheer number of the switches (20 million on the A3PE3000 for instance) make it essentially impossible to locate each cell and identify its programming state. Invasive probing to evaluate each flash switch would result in the destruction (flash cell charge) of the very programmed states needed to reverse engineer the design. Even if the bitstream could be extracted, reverse engineering the bitstream to a meaningful schematic is an extremely tedious process.

SmartFusion2 provides additional protections against reverse engineering. For example, bitstream keys are stored on-chip in encrypted form, and passcodes are cryptographically hashed. Differential power analysis (DPA) countermeasures are applied to prevent extraction of keys during the time when they are being used in computations. All protocols have been hardened against tampering. Countermeasures are even provided to prevent or at least detect many semi-invasive or invasive attacks, such as tampering with security settings with lasers or probes, whereupon the device can be commanded to destroy all sensitive data before it can be compromised.

# Data Security Requirements

Data security refers to applications that a system utilizes to protect and authenticate data. Common data assurance functions include confidentiality, integrity, authenticity, availability and non-repudiation. Key to implementing any of these functions is the establishment of a secure zone within which the vulnerable parts of the system can be protected from intrusion.

All fielded systems require a starting point to establish security for any software running on the system. A hardware root-of-trust (RoT) is used to create a secure zone within which security keys can be stored and operations performed while protected against malicious attacks. The hardware RoT can then be used to extend the zone of trust to cover other parts of the system, even allowing secure communications across an entire entrusted network. Examples of these types of zones include the execution of secure boot code, signature checking of software stored in external memory and validation of system boards for authenticity to combat cloning. Several advanced techniques are available to implement security zones. Three examples are the use of industry standard encryption IP, protection against security attacks and zeroization.

## Industry Standard Encryption IP

SmartFusion2 devices include the most robust suite of industry-standard data encryption IP available. Current functions include DES, 3DES, AES, pseudo random number generators, secure hash algorithm (SHA), RSA, elliptic curve cryptography, (ECC) and GCM for 802.1ae. Several of these functions are available as hardened IP blocks and can be used as hardware accelerators to simplify the creation of a target application with security features of its own. For example, AES-256, SHA-256 and HMAC cryptographic services are all available to the target application. Additionally, a 384-bit elliptic curve cryptography engine is also available to further secure application data. A pseudo-PUF challenge-response service is available to help establish application level key exchanges needed to establish trusted connections. SmartFusion2 devices thus not only establish a secure environment for the target design; they also assist in the creation of any security functions needed within the entire system.

## Protection Against Security Attacks

A more recent form of sophisticated attack on a cryptographic system called side channel analysis uses information that leaks, unintentionally, from the real-world implementations of cryptographic hardware. For example, an attack might examine the characteristics of a cryptographic device when a variety of security keys are presented. Even if the keys are incorrect measurements and analysis of the power use (called differential power analysis, or DPA), timing responses or electromagnetic radiation given off could provide clues as to the nature of the protected keys or algorithms used within the hardware root-of-trust described earlier. Given enough time and response examples, it may be possible to crack the stored keys and gain access to the other zones of trust for malicious purposes. SmartFusion2 devices include special patented DPA functions, licensed from Cryptography Research, Inc. (CRI, now a division of Rambus), that can protect against side-channel attacks to the FPGA’s initial configuration and reconfiguration process. Additionally, the users of selected SmartFusion2 devices can obtain a sub-license to use any of CRI’s DPA-mitigation techniques to protect their end-application from side-channel attacks.

## Zeroization

Zeroization is the practice of erasing sensitive parameters to prevent their disclosure if the system is attacked, or is at an increased risk of unauthorized access. SmartFusion2 devices support three different zeroization options, in addition to the ability to disable zeroization. The **Like New** option retains factory keys and serial numbers and the device can be reprogrammed if desired. The **Recoverable** option zeroizes factory keys, but new factory keys and serial numbers can be injected using authenticated key recovery service. The **Unrecoverable** option zeroizes everything and locks the device from all further operations. These capabilities make it exceedingly difficult for a detected attack on a SmartFusion2 device to capture any sensitive data prior to erasure.

## Redundancy Creates Fault-Tolerant Systems

In safety critical systems, redundancy is mandatory to operate properly in the event of a failure. There are two well-known techniques that are widely utilized: dual modular redundancy (DMR) and triple modular redundancy (TMR). In the case of dual modular redundancy, duplicate designs work in parallel. Each processing element receives the same input and a fail-safe certification engine checks for consistency. If a fault is identified, preventive action must be taken to avoid a failure. Triple modular redundancy creates three duplicate designs and the results of each output are presented to a voting circuit such that the output state that receives the most votes is set. Such a system can withstand the complete failure of one subsystem and allows a supervisor circuit to attempt to fix the fault, or alert an operator.

In conjunction with redundancy techniques, a design diversity methodology is sometimes employed to further improve reliability. Using this methodology, the parallel designs are not just duplicated (perhaps by using a copy of the same MCU or FPGA design), but will perform the same function using a different implementation. For example, an FPGA might be used for one of the designs and the parallel design might use an MCU. This diversity in the target implementations increases reliability even more since errors related to complex design or implementation bugs will not be duplicated in dramatically different targets.

# Example Design

It is helpful to explore an example design to better understand how the various aspects of design safety, reliability and security work together to deliver embedded systems for safety critical design that meet IEC 61508 standards. Prior to getting into the details of the design, a short description of the key features of the target SmartFusion2 system-on-chip (SoC) FPGA is presented.

## SmartFusion2 Architecture Description

The SmartFusion2 SoC FPGA, as shown in Figure 1, is a mix of hard IP blocks and FPGA on a single die. SmartFusion2 devices include critical features for high-reliability, advanced security and low power operation, but also deliver the sophisticated FPGA capabilities needed by a wide range of typical applications. SmartFusion2 devices thus combine the best of both worlds—dedicated features for specific applications and the advanced configurable logic features required by even the most aggressive of today's FPGA designs.

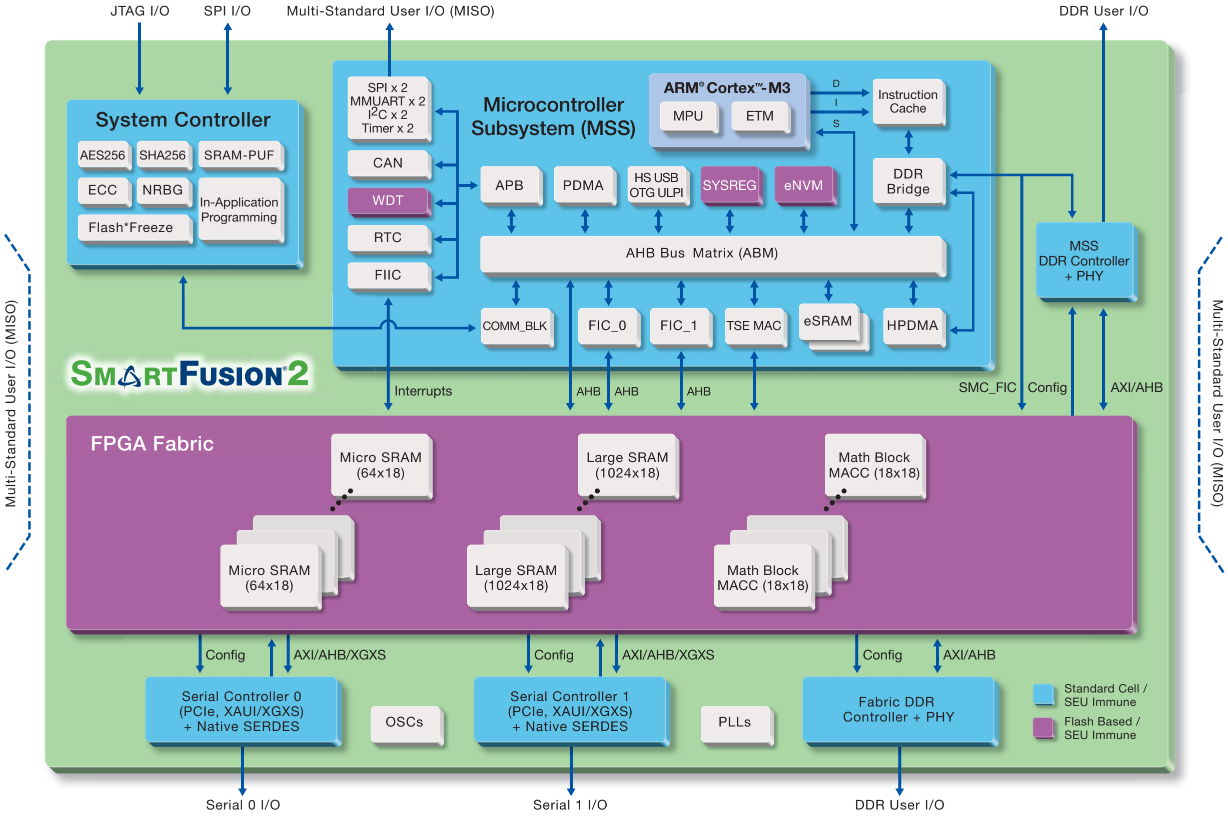

*Figure 1:* **SmartFusion2 Architectural Block Diagram**

The three key elements of the SmartFusion2 FPGA include the MSS (shown at the top of the diagram), the FPGA fabric (shown at the middle of the diagram), and the dedicated external interfaces: high-speed serial and high-speed parallel memory (shown at the bottom of the diagram). The MSS includes a high-speed ARM Cortex-M3 MPU with instruction cache, eNVM program memory and eSRAM data memory. A variety of peripherals are connected to the processor via an AHB bus matrix (ABM). The dedicated MSS peripherals include familiar MCU elements such as SPI, UART, I2C, CAN, WDT, RTC, USB and Triple Speed Ethernet (TSE) blocks. The MSS also has multiple interfaces to the FPGA, to allow for peripheral expansion and algorithm acceleration by creating custom peripherals and co-processing blocks within the

FPGA fabric. Advanced DMA controllers are available to efficiently move data between memory and peripherals. Specialized peripherals are also available within the MSS to efficiently implement security features such as AES-256, SHA-256, ECC and nondeterministic random bit generation (NRBG).

The SmartFusion2 FPGA fabric has up to 120K cells of 4-input look-up table (LUT) customizable logic blocks, small uSRAM blocks of 1Kbit each, large LSRAM of 18 Kb each, math blocks, PLLs and clock controllers. Several advanced features make it easy for SmartFusion2 devices to implement a wide range of applications. Each logic module includes dedicated carry chain logic to simplify the design of arithmetic functions. The dedicated math blocks efficiently implement up to 18×18 multiply operation natively with a dot-product option; have built-in addition, subtraction and accumulation units to simplify combining multiplication results such as those found in common DSP functions; support rich and flexible arithmetic rounding and saturation functions; contain several power optimization options; and support extendable operation sizes via the use of FPGA fabric.

The SmartFusion2 dedicated external high-speed interfaces include the high-speed MSS DDR controller, which is connected via a DDR bridge and interfaces to off-chip LPDDR, DDR2 or DDR3 memories. A similar DDR controller is connected via the FPGA fabric and supports specialized data access independent from the MCU to improve performance. High-speed serial interfaces, with up to sixteen 5 Gbps SERDES, also connect to the FPGA fabric and can operate in PCIe, XAUI/XGXS or native SERDES modes. Additionally, all SmartFabric2 devices feature a flexible I/O structure that supports a wide range of mixed voltages, along with programmable features to select drive strength, slew rate, input delay, pull-up and pull-down strength.

## Motor Control Design Requirements

A good example design that will help solidify many of the key topics covered so far is for motor control. Motors are a critical element in a wide range of applications with significant safety, security and reliability concerns. In transportation applications, for example, the failure of a motor to respond quickly and accurately to changes can result in loss of life, significant damage to expensive equipment or inefficient operation. In industrial processes, motors must maintain precise speeds or resulting chemical reactions may become unpredictable with catastrophic results. Even the failure of a simple motor used for something as innocuous as a cooling fan can start a chain reaction (perhaps literally) that leads to a significant loss.

Our example design will be for a motor controller using the Field Oriented Control (FOC) algorithm. FOC algorithms react much better to dynamic loads than traditional motor controllers and thus reduce vibration and save energy while reducing component count—all of which also improves system reliability. FOC algorithms use a variety of transforms (Park and Clark transforms, for example, for the numerically inclined readers) and computations (using the common proportional-integral and pulse-width modulation techniques, for example) for which either an MCU or FPGA are appropriate.

## SmartFusion2 Implementation

Assuming that a DMR approach is to be taken, an obvious advantage of the SmartFusion2 device is the ability to create one design in the MCU and one in the FPGA fabric, using different design methodologies and implementations. Further reliability is attained as this approach utilizes dissimilar technologies, as the MCU is a physically separate ASIC block from the FPGA, all located on the same die. The fail-safe detection circuitry is implemented in a separate design of the FPGA and compares the results from motor controllers 1 and 2. If a fault is detected, the error can be captured in a status flag or register and the system can be put into a known safe state until maintenance or replacement can be done. A block diagram of the target system is shown in Figure 2 on page 12.

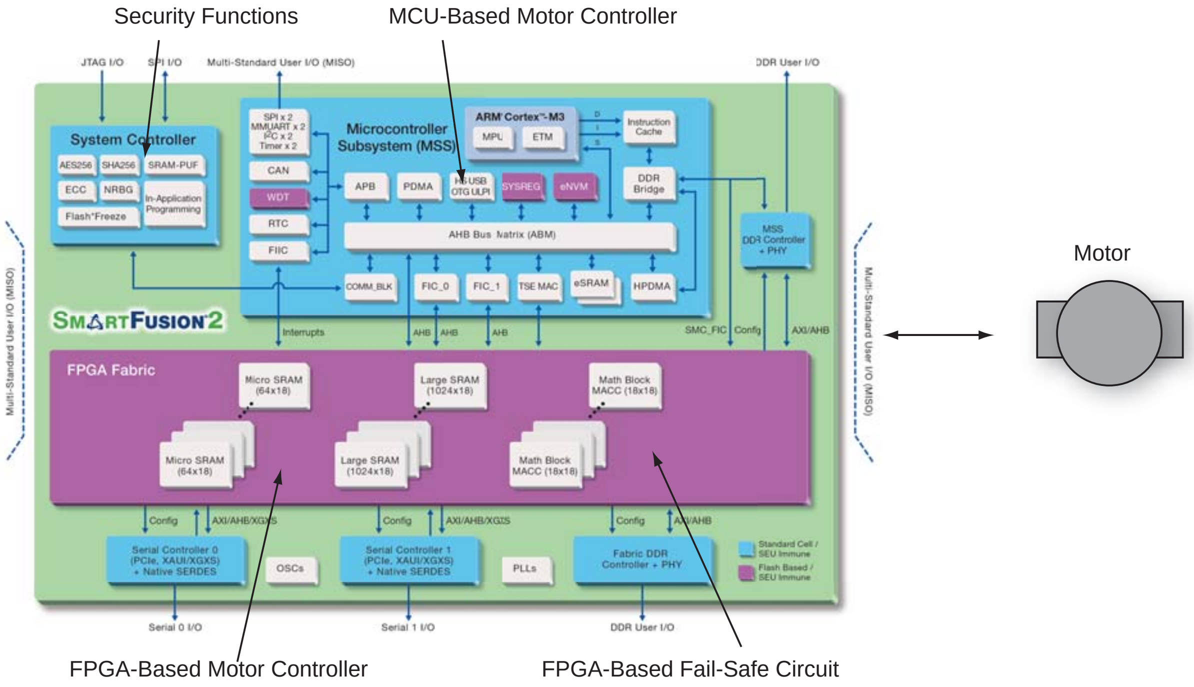

*Figure 2:* **DMR Implementation Using MCU and FPGA on a Single Die**

A SmartFusion2 design also benefits from other reliability advantages such as zero FIT-rate configuration memory, SEU protected memories, single error correct double error detect (SECDED) support for external memory controllers and built-in self-test. SmartFuson2 devices require no external configuration device, so this reduces component count and thus improves reliability. Another advantage of SmartFusion2 devices, sometimes overlooked when examining reliability, is their dramatically lower static power consumption. The unique ultra-low power Flash*Freeze mode, available in SmartFusion2 devices, can be used to preserve the state of the FPGA while stopping dynamic operation. Upon resumption, the FPGA continues operation from where it left off. These advanced low power capabilities can result in a smaller and less complex power supply and power distribution systems, which further improves system reliability. Additionally, when compared to SRAM-based FPGA implementations, where the start-up current spike associated with the long initial turn-on and configuration process of SRAM FPGAs is problematic, SmartFusion2 devices have a very clear start-up power advantage.

In the example motor control design there may be additional requirements for design and data security to protect the IP and supply chain and address other security life cycle concerns. In particular, there may be concern about protecting the manufacturing flow and implementing anti-tampering features. The single-chip nature of the SmartFusion2 implementation protects against attempts to reverse engineer the design. Additionally, by using the AES key to protect the programming bitstream and control the number of units that can be programmed, protection against cloning and overbuilding by manufacturing subcontractors is provided. The zeroization feature, perhaps using the U**nrecoverable** option, creates a very robust anti-tampering mechanism.

If the system uses remote reprogramming for updates and upgrades, there should be a secure communications channel for the FPGA configuration data and MCU code. The SmartFusion2 AES-256 encryption/decryption feature can be used to insure remotely sourced configuration bitstreams are protected from unauthorized observation.

# Conclusion

With the rapidly growing requirements for advanced safety and security in embedded network connected devices, not only will traditionally safety conscious designs need to implement advanced capabilities, but even traditionally mainstream industrial applications are seeing similar requirements emerging. SmartFusion2 devices are clearly the best-in-class solution for implementing safety critical, high-reliability designs. In addition, the advanced security features available in SmartFusion2 devices help create the secure life cycle on which high-reliability implementations ultimately depend.

# References

1. #World Nuclear Organization: www.world-nuclear.org/info/inf06.html.
2. *New Industrial Safety Standards Drive Component Revenue Growth in 2011*; July 11, 2012; Mark Watson.

**Microsemi Corporate Headquarters**
One Enterprise, Aliso Viejo CA 92656 USA
Within the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996