# Design of a Secure and Reliable Data Recorder

## Application Example

## Introduction

Data recording of critical operations has long been a requirement in many safety oriented applications. Perhaps the most notable example of data recording is the black box used in commercial and military aircraft. The large amounts of data generated and stored from aircraft flight operations is not only useful in determining failure mechanisms but is also useful for more mundane purposes like maintenance and reliability studies. Aircraft data recorders need high levels of reliability, even in the face of challenging environments with increased radiation levels. Additionally, it is important to make sure the data is authentic and known to be associated to the right data sources. The use of encryption can also be critical to protect the data from being downloaded and extracted by unauthorized parties.

Secure and reliable data recording applications are also finding their way into non-aerospace applications. Automobiles and trucks, both in commercial and industrial areas, are finding that data recording can be a valuable tool as insurance, liability, and safety considerations are becoming more important. As the move to autonomous vehicle operation accelerates, expect to see the equivalent of the black box become a ubiquitous element in the transportation infrastructure. Indeed, insurance companies are offering financial incentives to promote good driving behavior through their own black boxes connected to the OBD vehicle port. Clearly these insurance companies need to verify and authenticate that the data gathered came from the vehicle that was insured and that no tampering of the data exists.

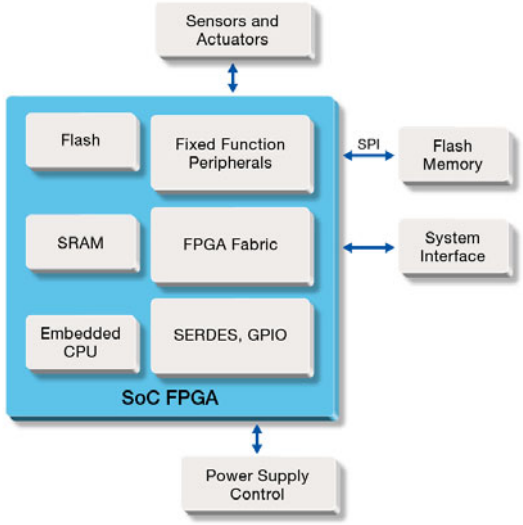


*Figure 1:* **Data Recorder Block Diagram**

This critical combination of reliability and security is a growing trend and looking in some detail at the design of a secure and reliable data recorder can help illustrate the key design techniques and features needed. A block diagram of a typical data recorder, without specific security and reliability requirements is shown in Figure 1 on page 1. A SoC FPGA is used as the main controller for the system. It interfaces to sensors and actuators to control and measure the various data sources. A system interface to a central controller integrates the data recorder with the rest of the system, as it isn't unusual for multiple data recorders to be located throughout the system. A SPI flash memory is used to store the recorded data managed by the FPGA. The FPGA may also supervise and monitor the power supply, since often the operation of the power supply is important to log. Let's look at the requirements for security and reliability that need to be added to the generic system from Figure 1 on page 1.

## Security and Reliability Requirements for the Example Design

The primary security requirement for the example design is the protection of the data being recorded. The data should be encrypted so that only an authorized party can download or inspect the data. Encryption should also be augmented with cryptographic authentication techniques to prove that the data is associated with the actual system. In some cases, hardware binding (proving the recorder communicated with specific hardware elements of the design) may be a requirement to further protect the system from tampering or aggressive intrusion techniques such as divide and conquer. Security keys must also be protected from attack. In particular, security key attacks like side-channel analysis (where timing and power measurements made during cryptographic operations can detect real world information 'leaks') need to be protected against.

The requirement to secure and authenticate the data being stored by the recorder is perhaps an obvious requirement, but less obvious is that securing the actual design of the data recorder is also of critical importance. If the design can be reverse engineered the security of the data can be easily compromised. A competitor could also use the reverse engineered information as the basis to clone the design, saving the expense of the significant effort involved in development and testing. For example, an unscrupulous contract manufacturer could copy the design and sell it on the open market at a much reduced price. Any of these attacks on the design could lead to a possible financial disaster, one that even litigation wouldn't fix.

Reliability is the other main requirement of the target design. Several design techniques can be used to improve reliability, one of the most common, is to duplicate functions adding redundancy to the design so that a single error doesn't result in a functional failure. For example, functions can be duplicated within the FPGA and checking logic employed to detect if the function outputs are the same. If they are different and an error was encountered it can then be adjusted for. Further opportunities for redundancy will be explored in the design and will include triple modular redundancy, design diversity, and error detection and correction techniques.

For designs that experience challenging environments, like increased radiation at higher altitudes, it is also important to protect against single even upset (SEU) events that come from energetic particle strikes. These particles carry enough energy that they can alter the content of sensitive SRAM cells in fine lithography electronics components. These events are less common at ground level, but when large numbers of devices are deployed, the chance of a 'flipped' bit grows and needs to be carefully considered. The implications of an altered bit in a safety critical application can be catastrophic, as illustrated by recent reports of unintended acceleration in an automotive control system. We will need to protect the example design from data or computation errors that could be created from SEU events.

# An Overview of Microsemi SmartFusion2 SoC FPGAs

A block diagram of the SmartFusion®2 SoC FPGA is shown in Figure 2, below. The hardened CPU and associated peripherals is shown in the large blue block at the top of the diagram. The System Controller, shown in the smaller blue block at the top left, includes a range of key security related functions that help with our implementation. The FPGA logic is shown in the large purple block in the middle of the diagram. SmartFusion2 SoC FPGA devices have a variety of features that make them not only attractive in general purpose applications, but also support advanced security and reliability requirements.

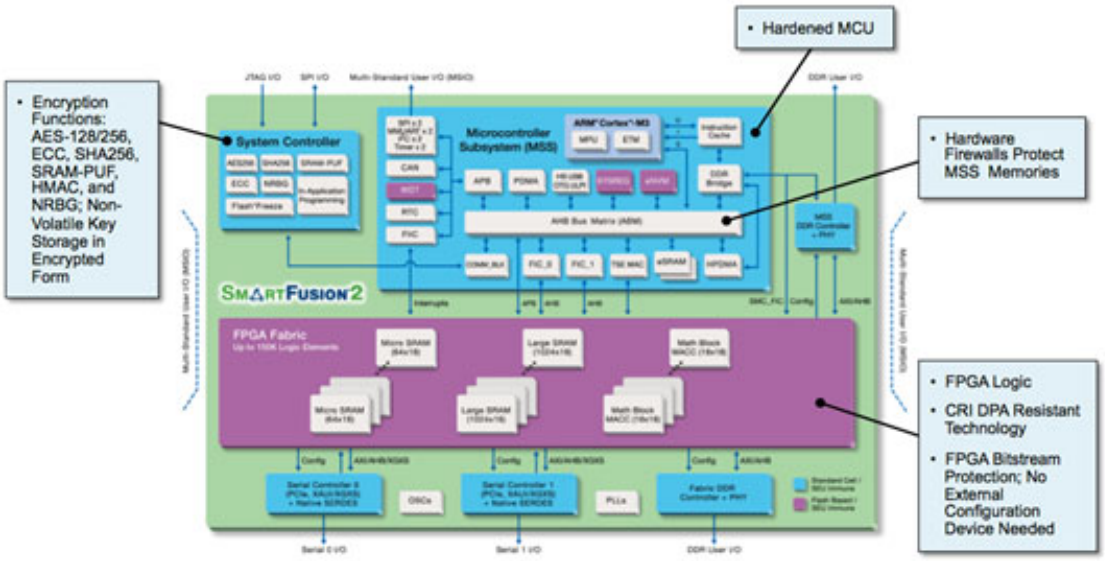


*Figure 2:* **SmartFusion2 SoC FPGA Architectural Block Diagram and Key Security Features**

SmartFusion2 SoC FPGA devices combine the full features ARM®Cortex™-M3 CPU with a variety of support functions. These functions are implemented in hard logic making them higher performance, lower-power, and lower-cost than an equivalent that is implemented in a programmable fabric. The FPGA fabric includes configurable logic modules (based on 4-input Look-Up-Table, or LUT, logic blocks), block memories, and dedicated arithmetic logic. Advanced SERDES support, including hardened implementations for PCIe®and other higher-level functions, makes implementation of standard serial interfaces very efficient. Other on-chip security features, like the security services available in the system controller for a variety of standard encryption and authorization functions, will be described in more detail in the implementation section later in this document.

The use of on-chip fabric-embedded configuration bitstreams on SmartFuson2 SoC FPGA devices supports a range of IP protection features, perhaps the most important is that configuration data isn't 'snoopable' on every start-up cycle. SRAM-based FPGAs require configuration data to be loaded into the FPGA on every start-up cycle, making them very easy to copy. For even more design security, the programming of SmartFusion2 SoC FPGA devices is done through an encrypted and authorized bitstream that is protecting the design when manufactured in an unsecure location. The security keys associated with bitstream programming need to be protected from even the most aggressive forms of attack, or the security of the design can be compromised.

The SmartFusion2 FPGA fabric is also protected from SEU-induced errors, since the flash configuration bits require much more energy to flip than their SRAM-based competitors.

Augmenting this inherent advantage is the liberal use of SEU-hardened functions (like implementing smaller buffer memories with latches, which are SEU resistant, and adding dedicated parity features to larger block memories, to easily detect memory-associated SEU events) to provide even more reliability.

# Implementing a Secure and Reliable Data Recorder Using Microsemi FPGAs and SoC FPGAs

Now that the key security and reliability design requirements have been described and we understand some of the key capabilities of the SmartFusion2 SoC FPGA, let's look at the example implementation in more detail to see how to satisfy these requirements in a SmartFusion2 SoC FPGA-based implementation.

## Satisfying Security Requirements Using Microsemi SmartFusion2 SoC FPGAs

We have determined that we need to secure the design intellectual property (IP) from reverse engineering and copying. We will also need to be able to protect application data using standard encryption and authentication techniques. Security keys should also be protected from attack and common side-channel attacks. Let's see how we satisfy these requirements using SmartFusion2 SoC FPGA devices.

### Protecting the Example Designs Intellectual Property

Both on-chip MCU code and FPGA configuration bitstreams in the example design are intrinsically protected from common forms of copying and reverse engineering, since they are stored on-chip in flash memory. It is much easier to copy or reverse engineer SRAM-based FPGAs. Even attempts to reverse engineer SmartFusion2 SoC FPGA devices using invasive techniques (by 'decapping' the device and attempting to read out the values of the configuration bits on chip) are protected against, since the vast number of cells and their embedded locations within the fabric makes it very difficult to reverse engineer the design.

Design IP is also protected during programming, since the bitstream is encrypted and authenticated. This eliminated the possibility of device copying during programming. For even more protection during manufacturing programming, a Certificate of Conformance (CoC) can be generated and logged. The CoC cryptographically verifies that the device was programmed with the intended bitstream. This makes it virtually impossible for a contract manufacture to clone or overbuild systems when you use SmartFusion2 SoC FPGAs as the basis for your design.

Additional protection against reverse engineering is available to eliminate the possibility of an intruder gaining access to the design through JTAG or the debugging interface. These interfaces can be protected by security 'locks' that keep out unauthorized accesses. If desired, locks can be put in place to make the device inaccessible for testing, debugging, or programming. This puts the device into a One Time Programmable (OTP) mode and makes it secure from even the most aggressive intruders. You can find out more about Design Security by referring to the articles and videos listed in the "To Learn More" section at the end of this paper.

### Supporting Data Security Standards in the Example Design

The SmartFusion2 SoC FPGA "S" devices (for example M2S090**S**-FG484) provide a significant amount of Data Security capabilities, typically implemented through service calls to the Security Subsystem within the System Controller. Several industry standard cryptographic functions are supported by simple security service calls including encryption and decryption algorithms, such as AES-128 and AES-256, a Message Authentication Code function (HMAC based on SHA-256), and a Non-Deterministic Random Number Generator (used in some advanced Data Security algorithms to improve secure transmissions by eliminating 'repeated' datasets).

More advanced functions include KeyTree Key Derivation (an alternative to HMAC), advanced challenge-response protocols to secure the transmission channel between the sender and receiver, and a Physically Unclonable Function (PUF) used to create a physically unique device ID (much like a fingerprint) to support more advanced security capabilities. These features can all be used to protect sensitive applications data stored within or transferred to/from the system. You can find out more about Data Security by referring to the articles and videos listed in the "To Learn More" section at the end of this paper.

### Protecting Security Keys from Attacks

We need to protect the security keys used in the example design from aggressive attacks that could compromise the cryptographic operations of the system. One popular method of attack is to use side-channel analysis (such as observations of power or timing signatures during security key-related operations) to try and determine on-chip secure information. This side-channel approach is similar to one a safecracker might use to determine the safes combination by listening to the noise made by the tumblers while manipulating the lock. In this case, the side-channel is the sound made by the physical implementation of the security “function”. The SmartFusion2 SoC FPGAs implement side-channel attack resistant decryption algorithms and in particular, are designed to be resistant to the most advanced Differential Power Analysis (DPA) form of side-channel attack.

If DPA-resistant techniques are not used, an intruder can measure the power consumed by the design when security keys and algorithms are being processed. Knowing the typical algorithm operations, a statistical analysis of the power use can be used to determine the security key value. For example, many security algorithms are implemented on an 8-bit basis, which means that only 256 combinations need to be checked through the DPA to identify an 8-bit section of the security key. DPA resistance can be dramatically improved by changing the architecture of the algorithm to limit such divide and conquer strategies. Additionally, changing the security key frequently will limit the number of measurements an attacker can use for statistical analysis making such approaches dramatically more difficult. Furthermore, circuit design tricks, like pre-charging registers and busses, will limit the “noise” available to an intruder. Many of the techniques used on Microsemi DPA-resistant FPGAs are licensed from Cryptographic™Research Inc. (CRI, a division of Rambus, Inc.) and contribute to making them the most secure devices available.

You can read more about these techniques in the articles and videos listed in the "To Learn More" section at the end of this paper under the heading "Protecting Your Design from Side-Channel Attacks".

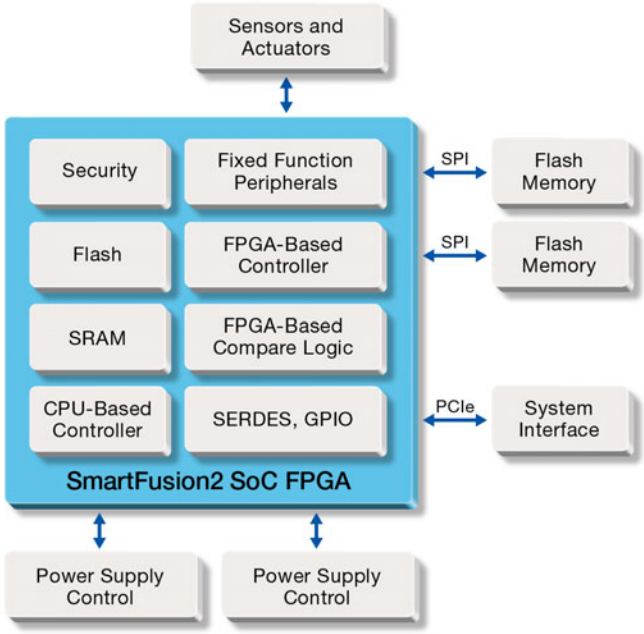


*Figure 3:* **Redundancy Techniques Applied to the Example Design**

## Satisfying Reliability Requirements in the Example Design

Let's take our example design and look at how we can significantly improve reliability by using redundancy techniques. Figure 3 on page 6 shows the changes to the example system. In order to improve storage reliability we added redundant SPI memories. This provides a back-up in case one memory fails or if a non-recoverable error damages data in one device. The new redundant power subsystem helps recover from a failure in the main supply. Power will switch over to a redundant supply if the main supply fails.

### Mitigation of Errors Through Redundancy and Design Diversity

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 then prevention 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. This can withstand the complete failure of one subsystem and allows a supervisor circuit to attempt to fix the fault, or alert an operator.

A design diversity methodology is sometimes employed to further improve reliability. Using this methodology parallel designs are not just duplicated 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.

The System Controller is now implemented using a Dual Modular Redundancy (DMR) technique. The controller functions are duplicated and FPGA-based compare logic is added to identify any outputs that do not 'agree'. When such an error is detected the subsystem responsible for the error can be reset and diagnostics performed. This mitigates the chance of the error resulting in a system failure.

Note that the dual implementations of the System Controller use a design diversity technique. One controller is implemented with the CPU and the other is implemented with the FPGA fabric. This provides additional reliability since each implementations error characteristics will be significantly different and thus the chance of a common systematic error (for example, their response to noise, temperature, voltage, timing differences, or even implementation ‘bugs’) will be significantly reduced.

### Implementing Additional Reliability Features

SmartFusion2 SoC FPGA devices can easily implement custom reliability features not found on standard ASSP devices. For example, SmartFusion2 SoC FPGAs have multiple SERDES channels (up to 16) on-chip making it possible to duplicate communications channels, perhaps PCIe in our example design, for additional levels of redundancy and reliability. Once these functions are integrated on-chip, additional redundancy, error checking, and advanced error recovery mechanisms could be included in the design. These additional features would create a more reliable design than those available in an ASSP. Note that the inclusion of the SEU-protected SRAM blocks and a Single Error Correction Double Error Detection external memory controller as features on SmartFusion2 SoC FPGA devices simplifies the design of higher level functions, since these memory-related reliability features don't need to be designed ‘from scratch’ and won’t require the use of additional FPGA fabric to implement them.

### Single Event Upset (SEU) as a Source of Errors

The Single Event Upset phenomenon was first discovered in 1979 by Intel and Bell Labs as failures in DRAMs and is attributed to stray alpha particles or neutrons ‘flipping’ the memory cell. In 1999 Sun Microsystems noticed errors in cached SRAMs for mission critical servers. In space and aviation applications the effects of radiation on electronics is well understood as operational altitudes have a higher neutron flux. However, the SEU phenomenon is increasingly becoming a concern at sea level as well. The continuous drive to smaller semiconductor geometries reduces the charge at each SRAM cell and the ever increasing content of electronics in fielded systems increases the likelihood of SEU-related SRAM errors. Note that flash memories, which require a significantly higher energy level to ‘flip’ state, are immune to these types of SEU events.

SmartFusion2 SoC FPGA devices make it easy to protect against SEU events due to benefits from several other ‘built-in’ reliability advantages that come from the underlying flash technology used in their implementation. For example, flash-based configuration memory is immune from SEU events and results in a zero FIT-rate contribution from configuration memory, unlike SRAM-based FPGAs which are orders of magnitude more susceptible to SEU events.

Another reliability feature related to the use of on-chip configuration storage is that SmartFuson2 SoC FPGA devices don't require an external configuration device, unlike SRAM-based FPGAs, which do. The reduced component count for a SmartFusion2 SoC FPGA implementation thus improves system reliability. SmartFusion2 SoC FPGA devices also have very low static power consumption, due to the inherent low-power advantages that come from using flash for configuration memory.

# Conclusion

The advanced safety and security capabilities of Microsemi SmartFusion2 SoC FPGAs satisfy all the key requirements needed for the most robust implementation of a secure and reliable data recorder. As illustrated in the example design, security requirements for the design, the application data and the security keys are all met and exceeded by the advanced features available on the Microsemi SmartFusion2 SoC FPGA device. The inherent security features of on-chip fabric-embedded flash configuration memory are augmented by encrypted and authenticated configuration bitstream loading with the associated keys protected from advanced side-channel attacks by CRI licensed mitigation techniques. On-chip hardware acceleration of a host of standard cryptographic functions further simplifies the implementation of highly secure systems. SmartFusion2 SoC FPGA resistance to errors from SEU events, also inherent from the use of flash-based fabric-embedded configuration memory, is also augmented with several additional features to further reduce system errors. SmartFusion2 SoC FPGA devices are your most secure and reliable platform on which to implement your embedded system.

# To Learn More

## Design Security

1. Securing Your Supply Chain Life Cycle White Paper
2. Securing Your Supply Chain Life Cycle Video
3. Securing Your Embedded System Life Cycle Video
4. What is Design Security in a Mainstream SoC? Webinar

## Data Security

1. Overview of Data Security Using Microsemi FPGAs and SoC FPGAs
2. SmartFusion2 and IGLOO2 Design Security and Cryptography Services

## Protecting Your Design from Side-Channel Attacks

Protecting FPGAs from Power Analysis

## Protecting Against SEU Events

1. Single Event Effects
2. Extreme Environments

**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

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for communications, defense & security, aerospace and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; security technologies and scalable anti-tamper products; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 3,400 employees globally. Learn more at www.microsemi.com.



55900180-0/3.14