Microsemi
Power Matters.™

Differential Power Analysis Countermeasures for the Configuration of SRAM FPGAs

White Paper

September 2015

Differential Power Analysis Countermeasures for the Configuration of SRAM FPGAs

Abstract

This paper presents a practical way to mitigate Side Channel Analysis vulnerabilities inherent in the bitstream decryption engines in SRAM FPGAs. The method uses SmartFusion®2 flash based FPGAs as the secure root-of-trust. A multi-staged approach is used, where the SmartFusion2 host establishes a secure boot loader in the target FPGA. The host and boot loader establish a secret key pair to securely transmit the long term application keys used to decrypt the final bitstream to be loaded in to the device. All cryptographic algorithms have Differential Power Analysis (DPA) countermeasures in place. This work further presents data on the effectiveness of the underlying solution using a statistical characterization of side channel leakage using the Test Vector Leakage Assessment (TVLA) methodology proposed by Cryptography Research, Inc. (CRI).

The following sections are covered in this white paper:

- Introduction—validation of the effectiveness of this work is performed using TVLA.
- TVLA Background—provides background information on TVLA and TVLA results of a specific leakage function in the configuration bitstream loader on Xilinx Virtex5 FPGAs.
- Validation Of DPA Mitigation Schemes—provides detail about the overall Microsemi® secure boot methodology, and details characterization using TVLA of the underlying cryptographic algorithms used in the solution.
- Tamper Resilience—examines other aspects of the implementation that provide tamper resilience.
- Conclusions—concludes this white paper.

Introduction

Most mainstream commercial FPGAs employ bitstream encryption as a means of protection of the underlying design contents. However, many published works have described successful key extraction from the bitstream decryption engine using Side Channels, most frequently with DPA and Differential Electromagnetic Analysis (DEMA). Vulnerabilities have been discovered in a several generations of SRAM FPGAs, including (as of the time of the current writing) the latest generation of commercially available 28 nm FPGAs [1, 2, 4, and 13].

DPA, first introduced by Cryptography Research Inc, remains a powerful and non-invasive way to extract secret key material from a device [5, 6]. In vulnerable devices, physical implementation or execution of the cryptographic circuit influences power consumption resulting in information leakage that is dependent on the secret key. The analysis relies on a specific leakage or set of leakage attributes. That is, there are discernible data or key-dependent differences in the distribution of power consumption measurements of specific intermediate processing steps in the underlying algorithm. For example, all traces sorted according to whether the least significant bit output of the first S-box in the AES algorithm is a '1' or a '0' may have different average values between the power distribution curves.

Given assumed information, or knowledge a priori about a specific leakage, an attacker will make a guess at a sub-key then sort all traces based on the assumed key value, specific leakage point and the leakage model, often as a binary selection criteria. The pair-wise distributions are then compared for differences. If the sub-key guess is correct, there will be a discernible difference of means between the two distributions at the processing time coinciding with where the leakage occurred. Hence, knowledge of any specific leakage function is enough to deduce secret variables related to an intermediate processing step. With the known secret intermediates, the secret key can be derived in a straightforward manner in a divide-and-conquer manner by an attacker with some knowledge of the underlying algorithm, but without requiring hardly any knowledge of the implementation details. In many cases, much is learned about the implementation from the non-invasive leakage measurements.

There are a variety of countermeasures that will mitigate the effectiveness of a side channel attack [16]. The Microsemi developed secure boot FPGA solution uses a SmartFusion2 as the root-of-trust and extends this trust to an SRAM based Xilinx® or Altera® FPGA through instantiation of a trusted configuration loader. The configuration loader is an IP that resides in the fabric of the target SRAM FPGA that replaces the functionality of the vulnerable built-in decryption engine in the SRAM target device. The final application image is decrypted and authenticated through the Microsemi configuration loader and loaded using partial reconfiguration. All cryptographic elements are protected from the loss of secret key materials, with DPA countermeasures where appropriate.

This study uses Xilinx Kintex®7 and Virtex®5 FPGAs, however, the general solution is extensible to all SRAM FPGAs capable of partial reconfiguration.

TVLA Background

A challenge that is common to side channel evaluation testing is repeatability between different evaluators. Pass/Fail testing based upon successful key extraction depends heavily on the expertise of the individual to determine specific information leakage attributes of the underlying system. Cryptography Research, Inc. has proposed a methodology called Test Vector Leakage Assessment (TVLA) as a means to derive an objective score for side channel vulnerabilities [3]. The core premise is to focus on characterizing the amount of information leakage and not a successful key extraction. This approach is based on a Welch's t-test to determine the correlation between two groups of power traces sorted according to input data or keys, (for example, fixed vs. random data), or by intermediate values computed during cryptographic processing steps. It relies on the assumption that a null hypothesis between the estimated distributions of the two groups of power traces will have identical means if there is no data-dependent first-order information leakage. Welsh's t-statistic is proportional to the difference in the means normalized by the estimated variances and the sizes of the two populations, and in actual laboratory practice, proves to be very sensitive to side-channel leakage.

As example of a TVLA analysis, the Xilinx Virtex 5 XC5VLX30 on the SASEBO GII DPA evaluation board [17] has a known leakage attribute. A single output bit (bit 55) in round 1 of the AES decrypt function has a state-dependent power signature. Figure 1 shows an overlay of two average traces of 80,000 power measurements for each of two sub-sets composing the full data set. The power trace data window is positioned around the location in time of the leakage attribute (sample 153). The selection function used to sort the data set is based on a "hamming distance" leak, where bit 55 of the 128 bit AES state vector updated through different power consumption profile when round 1 has a change in the state occurs versus when the register value remains constant through the processing step. Figure 2 shows an enlarged scale, with the sample window centered on the sample with the largest absolute difference between the two averages. A circle highlights the specific leakage attribute location in time. Figure 3 shows the overall t-test statistic for all sample points presented in Figure 1. It shows a point-wise difference, for each time instant, between the two data sets in Figure 1 according to equation (1), where X is the average of all of the traces in the group, S is the sample standard deviation in the group, and N is the total number of traces collected per group.

$$\frac{X_A - X_B}{\sqrt{\frac{S_A^2}{N_A} + \frac{S_B^2}{N_B}}}$$

EQ 1

proposes a passing t-test statistic score of $|t| \leq 4.5\sigma$, which implies a confidence factor of 99.999% that the two groups are sampled from the same distribution. Given the selection function described for Figure 1, the t-test statistic score crosses the 4.5σ threshold for the leakage attribute previously identified, failing the TVLA test. The large value of the t-statistic at time sample 153 ($|t|>15\sigma$) indicates a very high confidence that the difference in the mean traces highlighted in Figure 2 is statistically significant and not the result of random chance. Not all leakage exposed by the TVLA methodology may immediately lead to a working key extraction attack, however, this is more frequently the case when a specific intermediate calculation leaks such as in this example. In this case, a complete working key extraction attack was demonstrated based in part on this specific leakage attribute of the AES hardware, affirming the vulnerability indicated by the TVLA result.

As of this of writing, there are no known successful key extractions on systems where the leakage statistic is less than 4.5σ. As one validation point, Figure 3 shows the t-test statistic for sample 153 as a function of the number of traces collected. The number of traces where the t-test statistic exceeds 4.5σ is approximately 10,000 traces. A successful key extraction in this specific example took 80,000 power traces, which is on the order of a single bitstream load. With the attack used in this example, TVLA pass/fail criteria is an order of magnitude more conservative than was absolutely necessary.

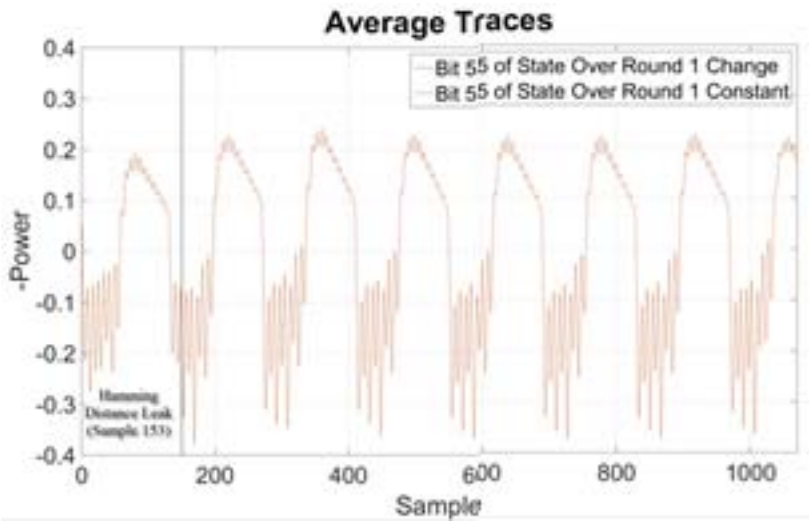

***Figure 1* • Two power trace averages (80,000 traces each) collected over a Xilinx bitstream decryption. The collection window captures the "Hamming distance" leak. Output of AES decrypt ro nd 1 bit 55, shows different power consumption when changing vs. when constant.**

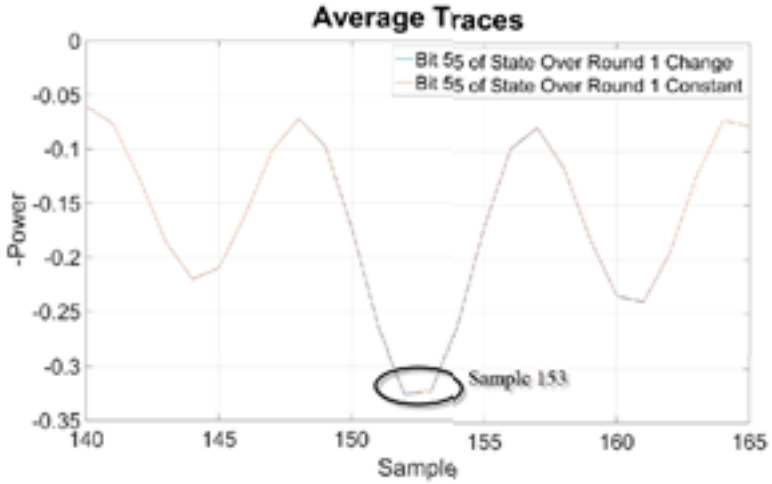

***Figure 2* • Average Traces from Figure 1, centered on the value of a Hamming distance leakage attribute of the decryption engine on the Xilinx Virtex 5 FPGA with an enlarged scale. Note the slight differences between the two curves at sample 153.**

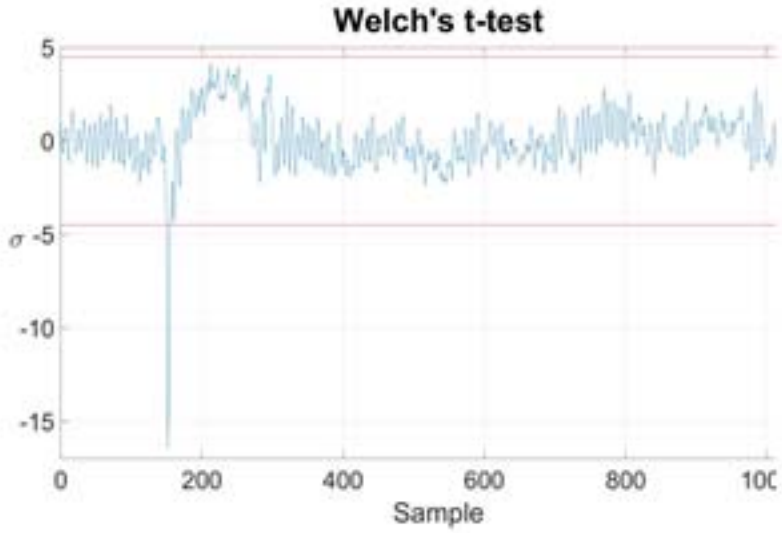

***Figure 3* • Welch's T-Test over Virtex 5 power trace data Leakage location identified in Figures 1, 2 crosses |t| = 4.5σ indicating with high confidence that the difference in means observed is significant.**

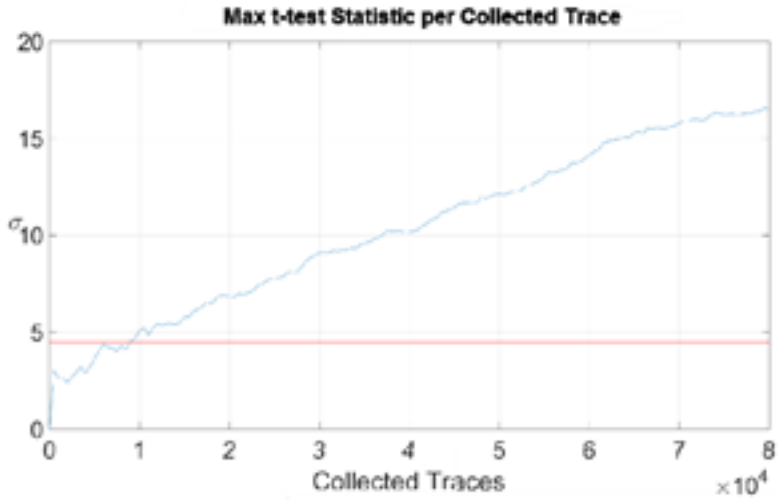

***Figure 4* • T-test Statistics as a Function of the number of power traces collected at the point of maximum leakage**

Microsemi Secure Boot FPGA Methodology

The Microsemi secure boot solution involves building a chain of trusted processing steps, with the end goal of provided confidential and authenticated SRAM FPGA loads in a manner that is resistant to side-channel analysis and other board-level attacks such as monitoring, replay, modification, and man-in- the-middle attacks. The primary requirements to enable a chain-of-trust include the following:

- A suitable root-of-trust, based on an immutable protected secret value that can be verified by a challenge-response protocol as a means of validating authenticity.
- Extension of the root-of trust with the ability to transfer a verifiable secret to the vulnerable target in an un-trusted environment.

SmartFusion2 FPGAs as a Root-of-Trust

SmartFusion2 FPGAs are a suitable root-of-trust as the devices themselves can be considered as having protected cryptographic boundaries. An intelligent system component or user has various ways to determine both the authenticity of the device and the design that it contains, forming the foundation for a secure hardware system.

Each SmartFusion2 FPGA is uniquely keyed with a number of secret keys. The SmartFusion2 FPGA provides an attestation protocol, where a programmer can interrogate the device to prove that it knows the factory secret based on the serial number of the device. In larger parts, containing the Elliptical Curve Cryptographic (ECC) engine and an SRAM Physically Unclonable Function (PUF), the key attestation protocol can be based on the factory enrolled ECC public-private key pair. As the private key is protected by the SRAM PUF, this represents an especially strong binding to the underlying physical silicon. Further, all devices contain a digital X.509 certificate signed with a Microsemi factory key. Bound to the certificate is the serial number, the model number with optional speed grade information, as well as a cryptographic value based on the device's secret key. The certificate, in tandem with the attestation protocol described previously, provides strong evidence that the host device in question is authentic.

User enrolled symmetric key material and the host design may also be verified. The enrolled symmetric keys (typically a root key used to protect the bitstream) can be verified with a challenge-response protocol by the programmer, similar to the procedure used to validate the device unique factory keys. After SmartFusion2 is loaded with a user design, the user may verify that the correct design has been loaded by checking the Certificate-of-Conformance (C-of-C) generated by the device. The C-of-C is a keyed digest based on the programming bitstream loaded and the serial number of the device.

Beyond verifying the authenticity of the design, all cryptographic processing of the device and user for the purposes of design security, included key derivations, keyed digests, and challenge response protocols are implemented with DPA countermeasures in place. All key materials are loaded onto the device in a protected manner. All devices have a Non-deterministic Random Number Generator (NRBG) to support key generation and nonces. The NRBG is built with multiple random entropy sources, and conditioned by a deterministic random number generator that is certified to NIST SP 800-90A.

Once sensitive information is loaded on the device, information is protected from physical tampering, with active and passive countermeasures such as glitch resistance and an active metal mesh over the area that stores security policy. Key materials are stored encrypted, with redundancy and integrity checks.A user-defined security policy includes granular access controls to restrict access to programming ports and prevent modifications to security parameters. Locked parameters may either be temporarily unlocked for changes based on an authorization pass-code, or they can be permanently locked as defined by the security policy. Furthermore, countermeasures are available as a system response including zeroization of user flash and the stored design in the case unexpected faults are encountered.

Root-of-Trust Extension

Trust in the target device requires establishing a verifiable secret in the target platform. This initial secret can then be used to authenticate the target platform, and transfer other secret key materials that can be application bitstream.

The secure boot FPGA solution takes a three stage approach, as summarized in Figure 5.

- Establishing Authenticity of the target hardware and the initial boot loader image
- Establishing an ephemeral session key between the host and target devices and the transfer of the long-term payload key to the target device
- Configuring the final application image. This includes authenticating, decrypting using the target boot loader image.

The solution is designed to have a minimal impact on board level architecture of existing designs with large SRAM FPGAs, as most board designs with a large SRAM FPGAs will likely have an external configuration controller to host the configuration load. A hosted configuration load for SRAM FPGAs enables use of a parallel interface for faster configuration loads and denser non-volatile memories. Many of the configuration pins on SRAM FPGAs are dual-use as configuration data pins during configuration load and as GPIO once a configuration image is successfully loaded.This solution uses as many as the existing configuration pins as possible as the communication interface between the host and target

device to minimize the impact on board level pin changes as compared to normal layout guidelines for a hosted configuration load.

Upload and Authenticate Boot Loader

Upon system boot-up, the SmartFusion2 host configures the target SRAM FPGA with an initial image as a boot loader (Boot-0). This initial image is sent as an unencrypted plaintext image, or if desired by system integrator, may use the SRAM FPGA's native bitstream encryption as an extra encryption wrapper. This image contains the logic that ties into the internal configuration port of the target SRAM FPGA and the cryptographic processing elements to decrypt the application image. Upon successful loading, Boot-0 calculates a CBC-MAC based over an internal read back of the Boot-0 configuration bitstream, a random nonce generated by the non-deterministic random number from the SmartFusion2 host and a device unique attribute. The MAC tag is sent back to the SmartFusion2 host for authentication. Binding over these three attributes authenticates Boot-0, the target device and provides protection against man-in-the middle/replay attacks. The initial reference design binds to a device unique ID from a Series 7 device (“Device DNA”), with a stronger form of binding based on a butterfly-PUF in the commercial versions of the solution.

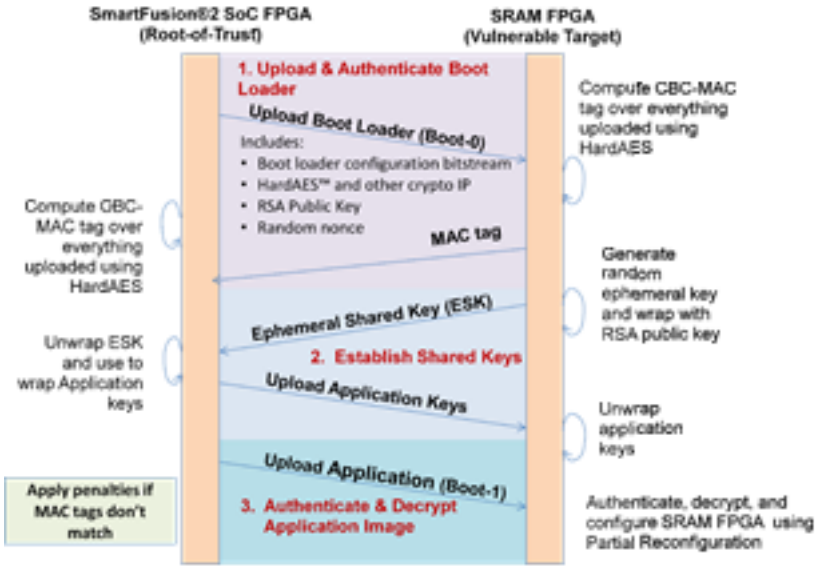

Figure 5 • **Secure Boot FPGA Process Overview**

HardAES, a Microsemi developed IP that is included in the Boot-0 load, provides the verifiable secret used to generate the MAC tag for validation. HardAES is a DPA-resilient gate-level implementation of the AES algorithm that uses white-box cryptographic techniques.

In a white-box model, an underlying assumption is that a target is open to inspection and physical tampering [12]. Examples are set-top boxes where an attacker can access memory through open ports to obtain key material. White-box techniques decompose the underlying algorithm into a much larger execution space, with the key transformed and expanded into a much larger key space. The decryption or encryption operation of the underlying algorithm is then a function of the input ciphertext or plaintext, respectively. Hence, the key is a function of the execution of the white-boxed algorithm and not an explicit input. The classical key material is never exposed or re-assembled in memory.

Boot-0 contains an implementation of HardAES that is based on AES-256. The algebraic decomposition of the AES algorithm in HardAES creates a key space expansion from 256 bits to something on the order of many kilobytes of data. The decomposition factor is tunable based on end system requirements to emphasize security or performance. Since, the classical AES algorithm key itself is never reassembled, there is a measure of DPA resiliency built-in to the HardAES algorithm. Any side-channels, even if there is information leakage, is not directly correlated with the classical AES key. In the commercial solution, HardAES is further strengthened with the ability to roll AES sub-keys as an additional countermeasure for DPA.

The SmartFusion2 host is enrolled with the same HardAES key and with the device unique attribute of the target device so that it may perform the same MAC tag calculation in order to authenticate the SRAM target. If the MAC tag does not validate properly, the SmartFusion2 host halts the target and applies system level penalties as appropriate.

Exchange of Ephemeral Shared Key and Application Load

After the target device and Boot-0 are authenticated, Boot-0 on the target FPGA and the SmartFusion2 root-of-trust FPGA exchange a session-based ephemeral AES 256 key. Boot-0 generates a random number that is encrypted with a 1024 bit RSA public key. A CBC-MAC tag based on HardAES is included with the ephemeral key as a means to ensure the authenticity of the ephemeral key.

The SmartFusion2 host decrypts the encrypted packet using the private RSA key. The RSA algorithm on the host must have DPA countermeasures in place, as it uses a secret RSA key. Algorithmic level countermeasures are employed to protect against DPA. The SmartFusion2 host uses the unwrapped ephemeral key to encrypt the long term payload keys for safe transmission to the target FPGA. This is the same long term payload key used to encrypt the main application image off line, and to decrypt it within the target FPGA during the DPA-resistant secure boot phase. The application image is stored encrypted in external NVM memory. The encrypted long term static key is stored in a protected sector in private non-volatile memory within the SmartFusion2 host device.

After the target device has the long term payload key, the SmartFusion2 host retrieves the application image from NVM and sends it to the Boot-0 application running on the target. The target uses a leakage resistant mode of decryption and authentication proposed by Cryptography Research Inc. [15].

Validation Of DPA Mitigation Schemes

There are three general strategies that may be employed to mitigate DPA vulnerabilities of an underlying system[6]:

- Leakage Reduction: reducing the signal-to-noise ratio of the leakage signal either through signal reduction or increasing the noise floor.
- Incorporation of Randomness: de-correlating side channel emanations through blinding and masking secret intermediates with random data.
- Protocol Level Countermeasures: limiting the number of transactions that can be performed with any given key.

A combination of all three elements is used in the cryptographic processing IP that is part of the Secure Boot FPGA solution. The SmartFusion2 SoC FPGA and the DPA-Resistant IP elements of the solution use patented techniques licensed by Microsemi from Cryptography Research, Inc. (a division of Rambus).

The integrated solution, shown in Figure 6 consists of a SmartFusion2 advanced development kit based on a M2S150T-1FCG1152 [10], paired up with a SASEBO GIII (Sakura-X), based on a Kintex 7 XC7K160T FPGA [18]. The SASEBO GIII board is a suitable platform for its ease of connectivity between the two boards, flexibility with respect to the configuration modes for the Kintex 7 device, and ease of power signature evaluation.

For evaluation testing of the three critical cryptographic implementations used in the secure boot solution, the current work uses the SmartFusion2 starter kit, based on a M2S010-FGG484, for the RSA decryption algorithm. The SASEBO GII board is used for evaluation of HardAES, and the conventional AES decryption algorithm for configuration load. The SASEBO GII is based on a Xilinx Virtex V XC5VLX50.

The SASEBO GII board is chosen in the current study primarily for ease of initial evaluation, as the platform setup was readily available and lent itself well to timely collection of data. The results of analysis and countermeasures implemented are portable between the two families, as the countermeasures do not directly rely on the technological level attributes between the two devices.Rather the countermeasures are logical and architectural attributes of the underlying algorithms. Validation on the SASEBO III board will form the basis of future work.

Measurements are taken on the SASEBO GII board using a Tektronics DPO7104C oscilloscope. Supply side voltage measurements on the Virtex 5 core voltage supply line are taken directly using an active probe. The design operates at 24 MHz. The acquisition sampling rate is set at 125 MSPS, with the probe bandwidth of 20 MHz. On the SmartFusion2 advanced development kit, the system clock is 166 MHz, with the acquisition sampling rate at 250 MSPS. Supply side voltage measurements on the 1.2 V core supply voltage are used to characterize the SmartFusion2 device.

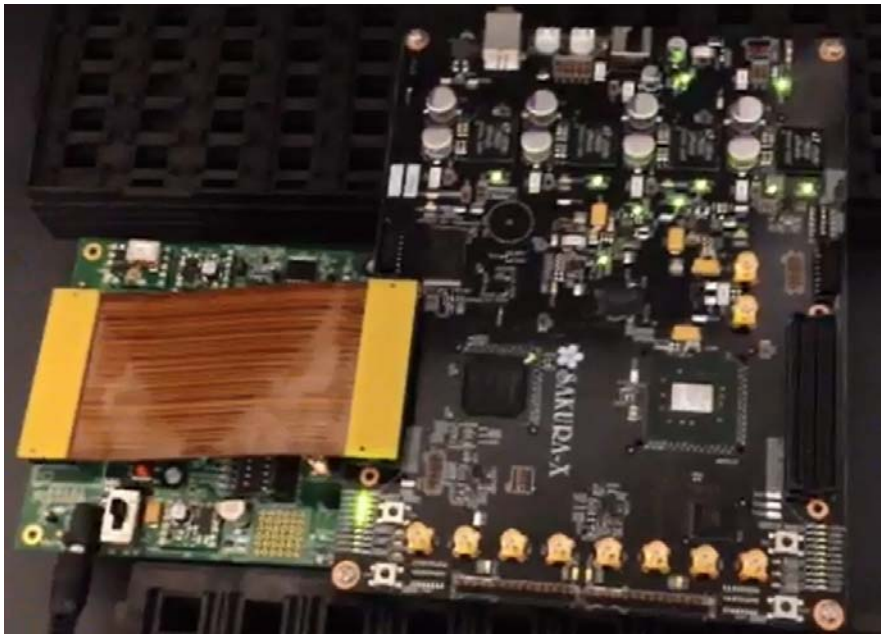

Figure 6 • **Microsemi Secure Boot FPGA reference platform using a SmartFusion2 Advanced development kit and a SASEBO III board with a Kintex 7 device**

For the evaluation benchmark, this study uses the "fixed versus random" (FVR) test proposed in [3]. In the FVR test, the algorithm under test uses a fixed set of keys. Two populations of voltage measurements are collected with the algorithm running repeatedly using a set of fixed input data for the first group, and randomly varying input data for the second group. The FVR test does not target a specific leakage attribute, rather it looks at aggregate information leakage at each point in time during the cryptographic operation. The resulting measurement traces are compared under Welch's T-test to derive a passing score based on $|t| \leq 4.5\sigma$.

TVLA Test for HardAES

Figure 7 shows the overall TVLA results for HardAES. In the secure boot solution, HardAES is used in the computation of MAC tags used to authenticate the as-loaded Boot-0 image and the ephemeral secret session key. A total of 25,000 traces were collected for each data set, that is, fixed inputs and random inputs. The key rolling countermeasure used in the solution (described later) was disabled and a fixed key was used for all 25,000 traces. Plot b shows Welch's t-test against the fixed and random input data sets. The two horizontal lines demarcate $t = \pm 4.5\sigma$.

The plot b shows that HardAES does have information leakage points that are significant. However, as noted previously, the structure of HardAES is that the key itself is never an explicit input into the function, but rather a function of the execution of the HardAES algorithm. The side channel leakages, though they do cross 4.5σ, do not directly correlate with the key material. Second, in the integration of the secure boot FPGA solution, HardAES is used for a limited number of rounds. The configuration read-back is compressed with SHA-256 prior running the HardAES algorithm for CBC-MAC generation as a measure to decrease processing time. Lastly, as the primary measure to provide a very high level of DPA immunity, sub-keys in the HardAES algorithm are updated as a protocol level countermeasure ("key rolling").

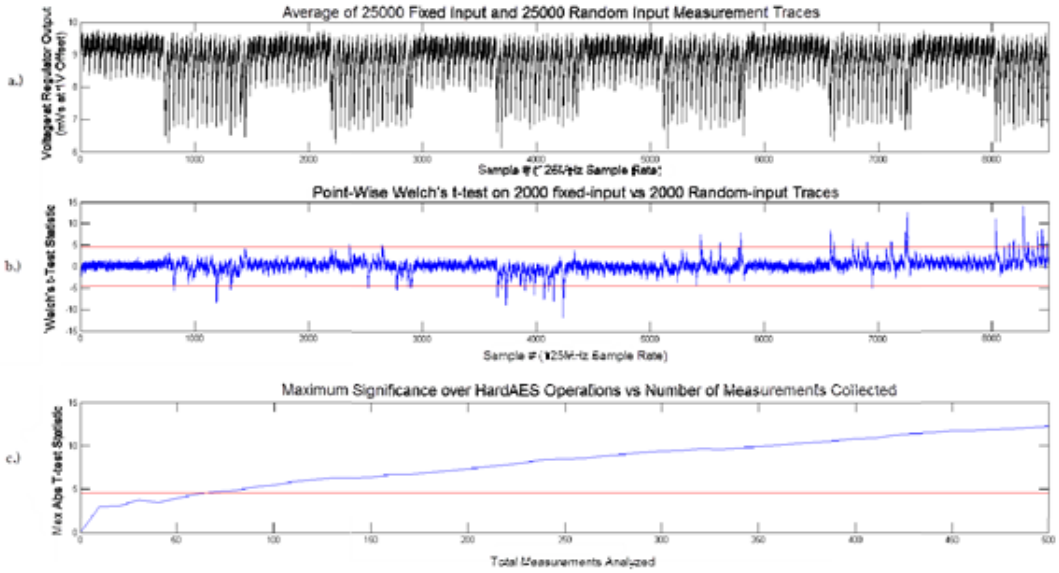

***Figure 7* • TVLA test results, HardAES. Plot a - shows the time-aligned average voltage measurement for the 25000 measurements using fixed and random input data. Plot b -Welches t-test statistic on the difference between the fixed input data set and the random data set. Plot c - Max t-test statistic as a function of the number of traces collected. Data presented is over the leakage point with the largest absolute value in plot b**

Plotting the t-test statistic as a function of the number of traces collected with a fixed key provides a guideline on the design margin inherent in the key rolling algorithm. Since, the pass/fail threshold is |t| = 4.5σ, the crossing point where |t| = 4.5σ for the number of traces collected infers a metric for the number of rounds the underlying algorithm can use a single instance of a key before the information leaked becomes statistically significant. Plot c in Figure 7 shows the max t-test statistic as a function of traces collected for the highest valued t-test statistic (largest information leakage) point depicted on plot b. Approximately 60 traces would be statistically significant according to the t-test statistic crossover point in plot c. Given that the key update schedule is at a sub-key level, that is, multiple sub-key updates per each AES block decrypted, the HardAES algorithm key schedule provides over two decades of design margin for DPA.

TVLA Test for RSA

In the secure boot solution, the RSA algorithm is used to encrypt the ephemeral secret session key on the target FPGA using a public RSA key for transmission to the SmartFusion2 FPGA where it is decrypted using the private key. RSA decryption on the SmartFusion2 SoC FPGA employs a combination of strategies to mitigate side channel leakages. Firstly, to eliminate SPA leaks during the RSA exponentiation operation, a constant-time square-and-always-multiply operation is performed for each key bit. Both input ciphertext message blinding and exponent blinding are employed using random numbers generated by the built-in NRBG on the SmartFusion2 device. In the exponent blinding operation, the secret exponent is dynamically split into multiple shares. Lastly, temporal noise is added through random clock delays.

Figure 8 shows the overall TVLA analysis on the RSA decryption algorithm. 3500 traces are collected over the first 16 square and multiply operations. Plot b in Figure 8 shows a definitive passing result over all traces collected.

TVLA Test for AES

Conventional AES is used to encrypt the final configuration image that is stored in PROM and subsequently uploaded and decrypted by the Boot-0 boot-loader upon each power-up cycle where it is used to program the target SRAM FPGA using partial reconfiguration. The AES decryption algorithm uses protocol level countermeasures based on key updates and key trees described in [15]. Key updates are scheduled every 4 AES blocks, as per the suggestion in [15]. This study performs TVLA on the underlying AES block cipher with key-rolling disabled to characterize aggregate information leakage in an unprotected implementation. Given the unprotected implementation, the threshold where the t-test statistic crosses 4.5σ provides a design margin metric. The TVLA data for AES decryption is provided in Figure 8.

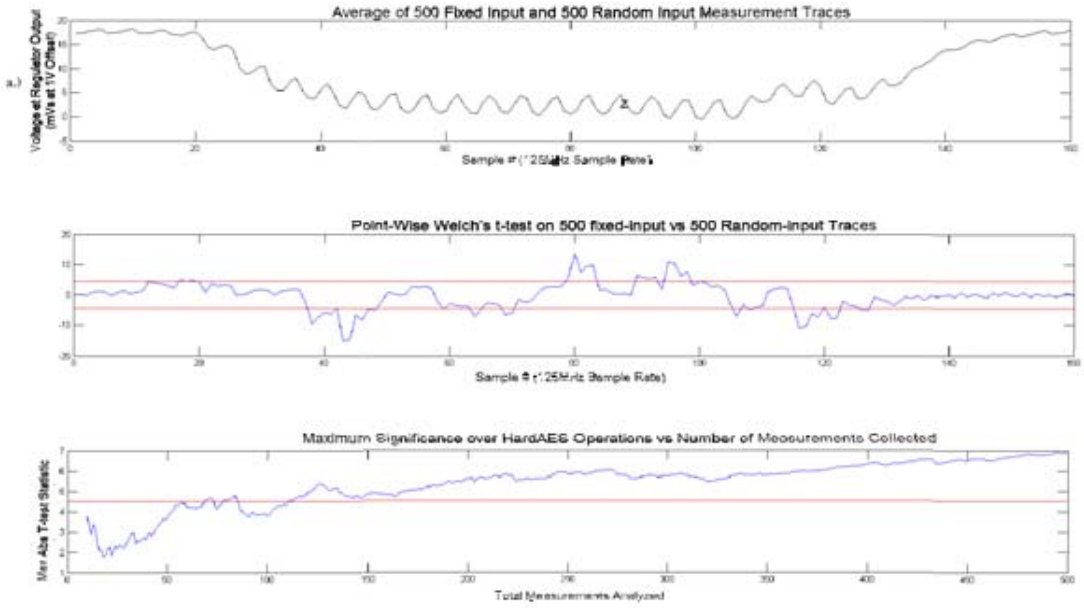

***Figure 8* • TVLA test results on classical AES block. Plot a -voltage measurement of the average of 500 fixed inputs and 500 random inputs. Plot b -Welch's t-test statistic between the fixed input data set and the random data set. Plot c - Max t-test statistic as a function of the number of traces collected. Data presented is over the leakage point with the largest absolute value in plot b**

The data indicate that the AES leakage becomes statistically significant with somewhere between 50 and 100 measurements of AES decryption with a fixed key. Since the key updates are scheduled every 4 AES blocks, there is sufficient design margin to prevent a successful DPA attack.

Tamper Resilience

The underlying algorithms presented in this solution so far provide a confidential and authenticated load of the SRAM FPGA bitstream. Providing additional tamper protection mechanism can increase the robustness of the solution, as well as increase security of the overall system as a part of a layered approach. The HardAES algorithm supports an input for a secret activation key computed from an enrolled PUF along with a random mask. The key modifies the interpretation of the underlying HardAES key. In this way, an enrolled PUF makes each specific instance of the HardAES algorithm unique. Microsemi has licensed a Butterfly-PUF from Intrinsic-ID to provide a "silicon biometric" ID on the target SRAM FPGA. Every device has unique credentials, with the underlying secret protected by the PUF. The secret activation key value is enrolled with the SmartFusion2 host, so that it may perform the same MAC tag calculation for authentication purposes. This binds the SmartFusion2 and the target FPGAs together without ever having to disclose the PUF secret, preventing undetected substitution of either of these devices by an adversary, and preventing relay-type attacks directing protocol messages to another device or system.

Future work on the Secure Boot solution also includes timing windows, where the SmartFusion2 host will enforce a timeout period on the calculation of credential information. This would be another layer of protection against a replay or relay attack.

In addition to the current solution, SmartFusion2 can be employed as a system level anti-tamper monitor that can exact penalties system wide. A common use case for SmartFusion2 devices is for power management.The Mixed Power Manager (MPM) is an IP developed for SmartFusion2 to manage digital point-of-load regulators for the purposes of power sequencing. A Microsemi developed IP, the EnforcIT® Security Monitor [9], contains independent clock and JTAG port monitors in addition to tying together the built-in tamper monitors that are a part of SmartFusion2 and any board level sensors the user may have.

The EnforcIT Security Monitor will coordinate tamper penalties in response to any unexpected fault conditions that occur during operational run-time. The tamper response layered with IPs like MPM and EnforcIT Security Monitor enables the system designer to exact a penalty system wide in response to tamper events including gracefully powering down a system, denying I/O or critical computations, or zeroization of critical security parameters.

Conclusions

This work shows that a SmartFusion2 FPGA, layered with a solution integrating DPA-mitigated cryptographic IP, yields a strong and effective solution for addressing DPA vulnerable endpoints. TVLA provides an objective score as to the vulnerability of unprotected cryptographic implementations contrasted to the effectiveness of the solution presented in this paper. Furthermore, in addition to root-of-trust services, SmartFusion2 is suitable as a part of a layered anti-tamper response system wide, where system-level penalties can be exacted in response to active tampering.

References

[1] A. Moradi, A. Barenghi, and C. Paar, "On the Portability of Side-Channel Attacks," Cryptology ePrint Archive, Report 2009/391.

[2] A. Moradi, A. Barenghi, T. Kasper, and C. Paar, "On the Vulnerability of FPGA Bitstream Encryption against Power Analysis Attacks - Extracting Keys from Xilinx Virtex-II FPGAs," In the 18th ACM Conference on Computer and Communications Security - CCS 2011. ACM, (2011).

[3] Goodwill, G., Jun, B., Jaffe, J., Rohatgi, P, "A testing methodology for side channel resistance validation," NIAT (2011)

[4] Hori, Y., Katashita,T., Sasaki,A., Satoh,A., "A First Report on Electromagnetic and Power Analysis Attacks against a 28-nm FPGA Device," Research Institute for Secure Systems (RISEC) (2013)

[5] Kocher,P, Jaffe,J., Jun, B. "Differential power analysis," In Advances in Cryptology - CRYPTO '99, LNCS 1666, pp. 388-397, Springer-Verlag, (1999)

[6] Kocher,P., Jaffe, J., and Jun, B. "Introduction to differential power analysis and related attacks, " http://www.cryptography.com/public/pdf/IntroToDPA.pdf (2011)

[7] Kocher, P., Jaffe, J., Jun, B., "Using unpredictable information to minimize leakage from smartcards and other cryptosystems," US Patent 6,327,661

[8] Kumar,S.S. et al., "The butterfly PUF protecting IP on every FPGA," IEEE Int. Workshop on Hardware-Oriented Security and Trust, pp. 67- 70, (2008).

[9] Microsemi Corporation. EnforcIT Product Brief. http://www.microsemi.com/index.php?option=com_docman&task=doc_download&gid=132856

[10] Microsemi Corporation. SmartFusion2 Advanced development Kit. http://www.microsemi.com/products/fpga-soc/design- resources/dev-kits/smartfusion2/smartfusion2-advanced- development-kit#overview

[11] Microsemi Corporation. SmartFusion2 SoC and IGLOO2 FPGAs Security Features. http://www.microsemi.com/document-portal/doc_download/134394-smartfusion2-soc-and-igloo2- fpgas-security-features

[12] Microsemi Corporation. WhiteboxCRYPTO Strength of Security. http://soc.microsemi.com/documents/reg/Whitebox_Crypto_Stre ngth_Security_WP_2.pdf

[13] Swierczynski P, Moradi A, Oswald D, Paar C, "Physical Security Evaluation of the Bitstream Encryption Mechanism of Altera Stratix II and Stratix III FPGAs." (2015)

[14] Tiri, K., Verbauwhede, I. "Synthesis of Secure FPGA Implementations. In Proceedings of International Workshop on Logic and Synthesis," IWLS (2004)

[15] Rohatgi, P., "Leakage Resistant Encryption and Decryption", http://www.cryptography.com/public/pdf/leakage-resistant- encryption-and-decryption.pdf (2011)

[16] Rohatgi, P., "Protecting FPGAs from Power Analysis", http://www.cryptography.com/public/pdf/FPGASecurity.pdf (2010)

[17] SASEBO GII : http://www.rcis.aist.go.jp/files/special/SASEBO/SASEBO-GII- en/SASEBO-GII_Spec_Ver1.01_English.pdf

[18] SASEBO GIII : http://www.risec.aist.go.jp/project/sasebo/download/SASEBO-GIII_Spec_v1_1_English.pdf

Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA

Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996

E-mail: sales.support@microsemi.com

© 2015 Microsemi Corporation. All rights reserved. Microsemi and the Microsemi logo are trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners.

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; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 3,600 employees globally. Learn more at **www.microsemi.com**.

Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer's responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided "as is, where is" and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this document or to any products and services at any time without notice.