Hardware Platform for Testing Performance of TRNGs Embedded in Actel Fusion FPGA

Michal VARCHOLA[1], *Miloš DRUTAROVSKÝ*[1], *Robert FOUQUET*[2], *Viktor FISCHER*[2]

[1] Dept. of Electronics and Multimedia Communications
Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic
[2] Laboratory Hubert Curien CNRS UMR 5516, 42000 Saint-Etienne, France

{Miso.Varchola, Milos.Drutarovsky}@tuke.sk, {robert.fouquet, fischer}@univ-st-etienne.fr

Abstract. *The paper presents hardware SoPC platform for testing performance of various TRNGs embedded in Actel FPGAs. The SoPC was implemented in the recent Actel Fusion ARM enabled FPGA device. It consists of four main blocks - CoreMP7 (Actel's soft-core industry standard ARM7 processor) for managing the SoPC, SRAM and Flash memories embedded inside the FPGA for storage and execution of CoreMP7 embedded software, custom TRNG design and Hi-Speed USB 2.0 interface based on a Cypress USB microcontroller for fast download of generated data to the computer. TRNG and USB interface peripherals are directly connected to the CoreMP7 peripheral bus to improve flexibility of using it. The results of NIST statistical testing of experimental PLL-based TRNG implementation demonstrate potential of designed hardware platform as well as new feature of Actel Fusion FPGAs - internal RC oscillator that could enhance security of the implemented TRNGs.*

Keywords

True random number generator, PLL, statistical tests, NIST, ARM7, Actel Fusion, USB interface.

1. Introduction

Security of various cryptographic systems [1] depends upon the generation of nonrecurring and unpredictable quantities that should remain unknown to adversary. For example common cryptosystems use keys that must be generated randomly. Many cryptographic protocols also require random inputs at various points, e.g., in generating digital signatures, padding of plain text messages, or in authentication protocols too.

Random number generators (RNGs) are hardware devices or software algorithms that are employed to produce random data. Generally, there are two main groups of RNGs: True Random number generators (TRNGs) and Pseudo-Random Generators (PRNGs). Latter mentioned are based on software algorithms having good statistical parameters, but on the other hand the output is deterministic, predictable and unusable for most high - security cryptographic applications. To obtain serious unpredictable values is possible only as a result of the random physical phenomenon.

Recent Field Programmable Gate Array (FPGA) based System on Programmable Chip (SoPC) can not implement natural physical processes such as direct amplification of a resistor or a PN junction noises typically used in external TRNGs. One of possible way, how to create complete cryptographic SoPC solution with TRNG inside FPGA is to obtain randomness by sampling jitter of internal oscillators. There are several concepts to produce timing noise including jitter of a Phase Locked Loop (PLL) output [2] and jitter of free running ring oscillator(s) [3]. Thus other components implemented inside SoPC including soft-core processor or cryptographic accelerators can use generated random numbers as they need.

There are three main objectives of a RNG implementation for cryptographic applications: good statistical properties, unpredictability, and good security parameters. Unpredictability means that any knowledge of previous random sequence or knowing of internal state of generator should not enable anyone to predict next sequence of bits. Good security parameters or attack immunity depends on generator's implementation. To ensure RNG immunity it is recommended to implement circuitry or algorithm for on-chip verification of RNG output quality. This circuitry or algorithm could detect whether RNG operates correctly or not. Statistical parameters are usually evaluated from recorded random sequences of certain length. It is possible to use statistical test suites such as: NIST [4], Diehard [5], or basic FIPS 140-2 [6]. To receive adequate result of statistical tests it is necessary to store large records (typically 1 Gb or even larger). Records are usually stored and tested in PC because of embedded hardware constraints. TRNGs could work at several megabits bit-rates (including some additional data necessary for TRNG state evaluation). Those are reasons why high speed connection between embedded device and PC is necessary to ensure safe error-free data transfer.

This paper describes SoPC platform that was designed in order to test performance implementation of various em-

bedded TRNGs. As SoPC base Actel Fusion ARM FPGA was chosen because its important advantages which are discussed in the article. To demonstrate advantages of this platform a TRNG based on internal PLL [2] was implemented, but TRNGs based on other principles can be also tested .

The paper is organized as follows: Section 2 presents the used hardware platform. Basic principle of tested PLL-based TRNG is described in Section 3. Section 4 briefly introduces NIST test suite. Results of proposed TRNG testing are included in Section 5. Conclusions and future development ideas are presented in Section 6.

2. Actel Fusion FPGA Hardware

Design was implemented into latest Actel's Fusion FPGA family [7]. Design was implemented into Actel System Management Board [8] with a M7AFS600 chip.

2.1 Description Actel Fusion FPGA

Fusion family combines all advantages of standard FPGAs (configurable, rapid prototyping, In System Programming (ISP)) and mixed-signal ASICs (highly integrated, application optimized, secure, low power, live at power-up, on chip flash, low unit cost, monolithic, nonvolatile and configurable analog components). As a single chip solution, Fusion reduces power, footprint, board complexity, risk and component costs. Fusion's FPGA fabric is based on the nonvolatile Flash technology that means no external Flash memories are necessary to store configuration as in the case of SRAM-bsased FPGAs. Thanks to Flash technology, Fusion device is live at power-up without requirement of another memory chip with critical data-paths on the board.

Used Actel FPGA is rich equipped by on-chip SRAM (up to 10kB) and Flash (up to 512kB), clock condition circuits (PLLs, Dividers), and internal RC oscillator (tuned at 100MHz) and of course there is a possibility to synthesize a CoreMP7. Actel's CoreMP7 [9] is the only soft IP ARM7 core that has been optimized for use in FPGAs. Using Actel's CoreConsole software tool it is possible to add IP blocks around the CoreMP7 and then program the whole design into the ARM7 enabled FPGA device. One of the great benefits of the CoreMP7 is that no ARM7 license or royalty fees are necessary. That means the CoreMP7 could be used even in very low cost designs. CoreMP7 executes the ARMv4T instruction set architecture and implements all 32-bit ARM7 instructions and all 16-bit Thumb instructions. The processor has a 3-stage pipeline, 32-bit ALU, 32-bit register file, a 32-bit external address and data bus interface, and JTAG debug interface. Fusion family provides internal RC oscillator and analog front-end also. Last mentioned includes: temperature measuring, analog-to-digital converter (ADC), and MOSFET gate drivers. Each of these advantages (e.g. CoreMP7, RC oscillator, or analog front-end) could help to enhance security level of TRNG or even complete SoPC implementation.

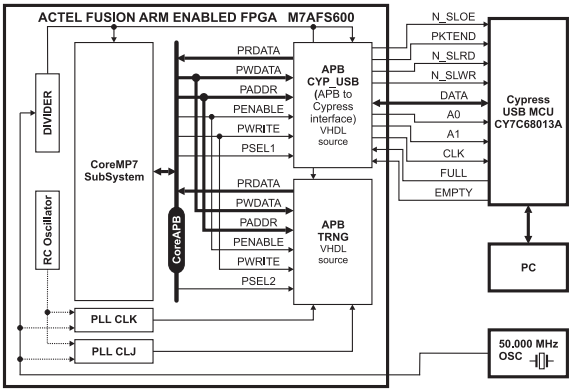

Fig. 1: Implemented hardware platform used for generation and recording random numbers based on Actel Fusion ARM Enabled FPGA,USB 2.0 interface and PC.

2.2 Description Actel Fusion FPGA

Block diagram of a system that was used for random data recording is depicted in Fig. 1. It consists of Actel Fusion FPGA, Cypress USB microcontroller CY7C68013A [10] and PC. The main component of the FPGA design is a CoreMP7 - industry standard soft-core ARM7 processor.

The CoreMP7 subsystem was configured in Actel's CoreConsole software. It consists of the following blocks: CoreMP7 (ARM7 core), SRAM and Flash embedded memories, and two standardized AMBA buses: Advanced High-Performance Bus (CoreAHB) and Advanced Peripheral Bus (CoreAPB). CoreAHB ensures connection between CoreMP7 and high-speed peripherals such as memories. CoreAPB ensures connection with lower-speed peripheral and it is taken out of subsystem for connection of custom IPs: APB TRNG and APB CYP_USB (APB to Cypress chip interface). APB TRNG was written in VHDL and utilizes two clocks CLK and CLJ for its operation. Both clocks are synthesized in internal FPGA PLLs. PLL reference clock could by taken from internal FPGA 100 MHz RC oscillator or 50.000MHz external crystal oscillator. CoreMP7 runs up to 25MHz approximately and its clock is generated by dividing 50.000MHz reference clock by factor 2, 3 or 4 in the Divider. APB CYP_USB translates APB signals to Cypress chip signals and vice versa. Reading from TRNG and sending data to PC via USB is controlled by CoreMP7 software.

Basic algorithm of CoreMP7 software is depicted in Fig. 2. This algorithm ensures storing random data in PC, in other words it performs reading 32-bit random word from TRNG and writing it as two 16-bit words via Cypress chip into USB bus because of Cypress' 16-bit parallel data bus. Algorithm waits for a request from the PC after hardware initialization . If start command is received a random data sequence of user defined length (e.g. 1 Gbit) is generated and send via USB to PC for performing statistical evaluation (e.g. by NIST test suite).

Great potential of CoreMP7 is that it can execute embedded (on-line) TRNG tests to detect incorrect operation

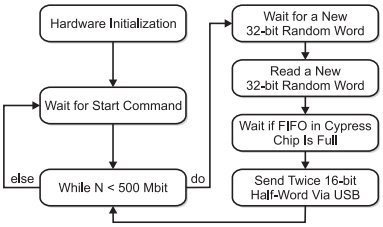

Fig. 2: Basic CoreMP7's algorithm which controls reading random values from TRNG and sending it via USB to PC.

of TRNG directly in the application. Tests could consist of simplified statistical tests or, thanks to analog front-end there are options to measure chip temperature, voltage of power supply and current consumption of FPGA. Either PLL based TRNGs or ring oscillator based ones are strongly depended on physical phenomenon such as temperature changes or noise in the power supply lines. Each of these phenomenon could by used to perform attack on cryptographic hardware. Great advantage of Actel Fusion FPGA is the possibility to create whole system with on-chip analog measurements. Proposed hardware platform enables to create an test such cryptographically strong system and it will be presented in next articles.

3. Basic Principle of PLL based TRNG

The basic principle behind the TRNG shown in Fig. 3 is to extract the randomness from the jitter of the clock signals synthesized in the embedded analog PLLs [2]. The jitter is detected by the sampling of a reference signal *CLJ* using a rationally related (clock) signal *CLK* synthesized in the on-chip analog PLLs with frequencies

$$F_{CLJ} = \frac{M_{CLJ}}{D_{CLJ}} F_{OSC} \tag{1}$$

$$F_{CLK} = \frac{M_{CLK}}{D_{CLK}} F_{OSC} \tag{2}$$

where F_{OSC} is a reference clock signal and parameters $K_M = M_{CLJ}D_{CLK}$, $K_D = D_{CLJ}M_{CLK}$ are related to the PLL structures. The signal *CLJ* is sampled into the first D flip-flop using a clock signal with frequency F_{CLK}. There are K_D rising edges of *CLK* signal and $2K_M$(rising and falling) edges of a *CLJ* waveform during the time period

$$T_Q = \frac{1}{R} = K_D T_{CLK} = K_M T_{CLJ} \tag{3}$$

where R is the bit-rate of the output TRNG sequence.

Internal Actel Fusion FPGA RC oscillator was used as clock source. It is great benefit of Actels Fusion FPGA, because there is no clock track needed outside of device which

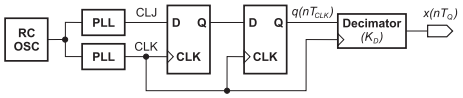

Fig. 3: Basic structure of implemented PLL-based TRNG.

could be influenced by attack. Because there is a probability that the first D flip-flop could become metastable, the second D flip-flop is cascaded in order to decrease influence of metastability on the following logic behavior. Delay line proposed in [2] was not used in order to simplify detection of TRNG's defects. The output $x\,(nT_Q)$ is connected to additional interface logic in order to interface with CoreMP7. CoreMP7 can use TRNG to generate secure keys or primes for its cryptographic algorithms. Additional logic ensures bits forming into 32-bits words readable by CoreMP7 subsystem and a new word indication.

4. NIST Test Suite description

The NIST statistical test suite [4] is a statistical package consisting of 15 tests (1: Frequency, 2: Block Frequency, 3: Cumulative Sums, 4: Runs, 5: Longest Run, 6: Binary Matrix Rank, 7: Discrete Fourier Transform, 8: Non-overlapping Template Matching, 9: Overlapping Template Matching, 10: Universal, 11: Approximate Entropy, 12: Random Excursions, 13: Random Excursions Variant, 14: Serial, and 15: Linear Complexity) that were developed to test the randomness of arbitrary long binary sequences produced by either hardware or software based cryptographic random or pseudorandom number generators. These tests focus on a variety different types of non-randomness that could exist in a sequence.

To perform NIST tests, version 1.8 was used and default parameters were used. Random data record had 500 Mbit length and it was divided into 500 sequences with 1 Mbit length of each. Consequently a set of P-values and proportion of passing sequences were evaluated at the significance level $\alpha = 0.01$.

5. Experimental Results

5.1 Implementation details

CoreMP7 is available with or without the on-chip debugging circuitry. The debug circuitry is about one third size of the CoreMP7. Ability to remove it after debugging leads to the possibility of the design to be implemented in smaller device or to add next custom logic instead of it. Tab. 1 shows resource requirements of implemented SoPC, where CoreMP7 Subsystem consists of CoreMP7 (core only), AMBA system buses and memory controllers. According to Fig. 1 Whole Design consists of Core MP7 subsystem, APB TRNG, and APB CYP_USB.

Entity	Used Tiles	
	–degug	+debug
APB TRNG	349	349
APB CYP_USB	120	120
CoreMP7 (core only)	6083	8090
CoreMP7 Subsystem	7797	9804
Whole Design	8266	10273

Tab. 1: Resource requirements of the design, where one Tile is equal to one D flip-flop.

Test	40 kbps TRNG		1 Mbps TRNG	
	P-value	Prop.	P-value	Prop.
1	0.100109	0.9940	0.000000*	0.0000*
2	0.090936	0.9960	0.000000*	0.0000*
3	0.440975	0.9920	0.000000*	0.0000*
4	0.786830	0.9920	0.000000*	0.0000*
5	0.502247	0.9920	0.000000*	0.0680*
6	0.935716	0.9900	0.222480	0.9880
7	0.743915	0.9820	0.000000*	0.0080*
8	0.387264	0.9840	0.000000*	0.0000*
9	0.624627	0.9920	0.000000*	0.0000*
10	0.657933	0.9900	0.000000*	0.6200*
11	0.691081	0.9880	0.000000*	0.0000*
12	0.237192	0.9908	0.000000*	0.0000*
13	0.428666	0.9908	0.000000*	0.0000*
14	0.632955	0.9880	0.000000*	0.0000*
15	0.763677	0.9920	0.148653	0.9940

Tab. 2: NITS P-values and proportions of passed blocks for 2 PLL based TRNG configurations ('*' means that test was not passed).

It was observed that 16-bit data could be transmitted each 2 periods of CopreMP7 system clock by ModelSim simulation. CoreMP7 is significant bit-rate limiting factor because it works approximately up to 25MHz and it has to execute other instructions e.g. testing end of loop or reading new value. Real algorithm inside CoreMP7 could utilize performance of USB 2.0 between 8 - 16 Mbps according to its clock frequency.

5.2 NIST tests results

Two configurations of TRNG have been tested operating at following bit-rates - 40 kbps and 1 Mbps. Both TRNG configurations were tested on three different boards in order to check if certain TRNG configuration is not board-depended. Results of NIST tests were approximately the same for each board. Tab. 2 shows results for only one board for each configuration. Slower 40 kbps TRNG pass the test very well but faster 1 Mbps does not (as it was expected and will be analyzed in next paper).

6. Conclusion

A flexible TRNG testing hardware platform has been described and evaluated within paper. It was embedded into Actel's Fusion ARM nonvolatile Flash FPGA. Resources of this device enable to embed whole reprogrammable design inside, even the external oscillator is not necessary if internal RC one is used. Thus it is possible to minimize off-chip tracks in order to prevent attacks. Only power lines could be exposed, but with powerful analog front-end it is possible to monitor power lines or chip temperature to detect manipulation and to prevent incorrect TRNG operation. Another option how to improve reliability of TRNG implementation is to perform on-line statistical test of generated TRNG data. The system is controlled by Actel's industry - standard ARM7 based CoreMP7. The soft-core ARM7 processor suits for this purpose very well, only firmware replacement is necessary to change testing or monitoring algorithms. TRNG and USB connection are added as peripherals. Analog front-end can be connected as additional peripheral. Implemented architecture provides high level of flexibility. Our future research and development will concentrate on observing jitter characteristic and on on-chip monitoring system to generate alarms, if output of TRNG is not correct.

Acknowledgement

This work has been done in the frame of the Slovak scientific projects VEGA 1/4088/07, and project KEGA 3/5238/07 of Slovak Ministry of Education.

References

[1] MENEZES, J .A., OORSCHOT, P. C., VANSTONE, S. A. Handbook of Applied Cryptography. *CRC Press, New York*, 1999.

[2] FISHER, V., DRUTAROVSKY, M. True Random Number Generator Embedded in Reconfigurable Hardware. In B.S. Kaliski, Jr., C.K. Koc, C. Paar, editors, *Cryptographic Hardware and Embedded Systems, 4th International Workshop (CHES)*. volume 2523 of LNCS, pages 415–430. Redwood Shores, CA, USA, Springer-Verlag, 2002.

[3] VALTCHANOV, B., AUBERT, A., BERNARD, F., FISHER, V. Modeling and observing the jitter in ring oscillators implemented in FPGAs. Submitted In *Workshop on Design and Diagnostics of Electronic Circuits and Systems (DDECS)*, 2008.

[4] RUKHIN, A., et al. NIST Special Publication 800-22. A statistical test suit for random and pseudorandom number generators for cryptographic applications, 2001

[5] MARSAGLIA, G. Diehard: a battery of tests of randomness, 1997. Online. Available: www.stat.fsu.edu/pub/diehard/

[6] FIPS Publication 140-2. Security requirement for Cryptographic Modules, January 1994

[7] Actel. Fusion Family of Mixed-Signal Flash FPGAs with Optional Soft ARM Support, October 2007. Online. Available at: www.actel.com/documents/Fusion_DS.pdf

[8] Actel. System Management Board User's Guide, December 2006. Online. Available at: www.actel.com/documents/SysMgmtBrd_UG.pdf

[9] Actel. CoreMP7 Data Sheet, July 2007. Online. Available at: www.actel.com/documents/CoreMP7_DS.pdf

[10] Cypress. CY7C68013A EZ-USB FX2 USB Microcontroller, June 2002. Online. Available at: download.cypress.com.edgesuite.net /design_resources/datasheets/contents/cy7c68013a_8.pdf