SAND2018-0214C

A Rad-Hard eFPGA Core for Reconfigurable System-On-Chips

John Teifel, Jeff Black, Bill Cavanaugh, Kevin Clark, Jeff McCasland, Richard Owen, Moslema Sharif
Sandia National Laboratories
Albuquerque, NM 87185, U.S.A.
jteifel@sandia.gov

***Abstract*— Radiation hardened embedded FPGA (eFPGA) cores are desirable for System-on-Chip applications to reduce costs and increase reuse potential. To meet this need, Sandia has licensed the EFLX(TM)-4K eFPGA logic core from Flex Logix Technologies for use in its 180-nm radiation hardened process node. This paper describes the radiation hardened version of this eFPGA core and its application to System-on-Chip designs.**

Keywords*— *Rad-hard; embedded FPGA; System-on-Chip

I. Introduction

While there have been a number of efforts to implement commercial Field Programmable Gate Array (FPGA) devices in radiation hardened (rad-hard) process nodes [1][2], these have had limited success in replacing rad-hard Application Specific Integrated Circuits (ASICs) in government applications. The primary challenges have included the FPGA's lower logic density and performance, relative to an ASIC, and the difficulty in modifying the architecture and software tools of the commercial FPGA device to meet unique government application requirements. Emerging embedded FPGA (eFPGA) technology, where FPGA cores can easily be integrated within a larger System-on-Chip (SoC), provides a potential solution to these previous challenges and presents a new design opportunity for rad-hard government applications. This enables the System-on-Chip designer to partition logic functions between fixed ASIC blocks and flexible eFPGA cores, and to be able to customize the behavior of the eFPGA cores to meet specific government application needs [3]. In this paper, we describe the implementation of Flex Logix's commercial EFLX™-4K eFPGA logic core into Sandia's 180-nm radiation hardened process node, and illustrate its usage in a rad-hard reconfigurable SoC design.

II. Background

Traditional FPGA device development requires many man-years of engineering effort to optimize custom circuit layouts for the smallest area and maximum speed, and to modify the bitstream generation software to support the new device. Even different sized devices in the same FPGA family and process fabrication technology typically require significant re-engineering efforts to, for example, re-insert clock trees and to modify the configuration logic and bitstream software to account for the different resource sizes. Likewise, porting an FPGA device between different chip manufacturers or re-

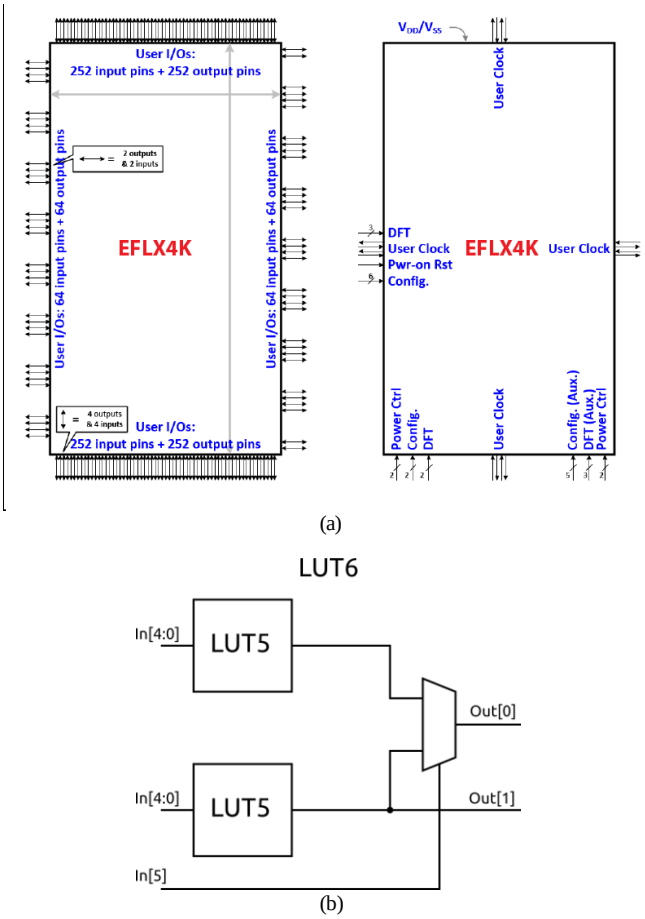


Fig. 1. EFLX-4K eFPGA logic core: (a) interface diagram and (b) six-input look-up-table (LUT6) programmable logic circuit [7].

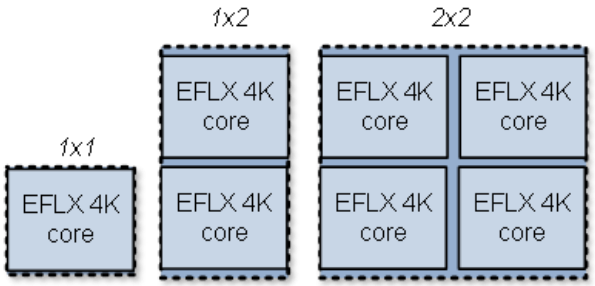


Fig. 2. Radiation hardened eFPGA array configurations that are available for use in reconfigurable System-on-Chip applications and application-specific FPGA devices.

designing it for radiation tolerance requires a large capital investment [4]. It is even more cost prohibitive to realize an application-specific FPGA device, where chip resources such as programmable logic elements, I/O pads, memory, and other hard IP (intellectual property) blocks are optimized for a specific range of applications. This is especially true for government applications, where low-volume production quantities make it difficult to amortize the very high development cost of such a custom FPGA device.

An alternative development approach for realizing an application-specific FPGA device, is to leverage automated (fast turn, lower cost) ASIC design flows and embed an FPGA block inside a larger System-on-Chip. While this embedded FPGA (eFPGA) concept has been attempted in the past [5], critical challenges still need to be addressed, including how to cost effectively develop the FPGA layout block, how to make the layout easily portable across different fabrication technologies, how to scale the eFPGA resources on an application-specific basis, and how to implement the software required to configure the eFPGA. Given the strong industry demand for a robust eFPGA technology, there are currently over half-a-dozen commercial companies working to solve these issues, each with different approaches [6], ranging from full custom 'hard' layouts (similar to a traditional FPGA design flow) to fully 'soft' synthesized solutions (similar to a traditional standard-cell ASIC auto-place-and-route design flow).

To realize application-specific FPGAs in its 180-nm rad-hard fabrication process, Sandia has licensed the EFLX-4K eFPGA logic core from Flex Logix Technologies for use in reconfigurable System-on-Chip government applications. The EFLX-4K uses a middle-ground development approach, where the eFPGA layout is implemented with standard layout cells to keep development costs manageable, but with those cells compacted together into a 'hard' layout to minimize impacts to density and performance (versus a full custom eFPGA layout). Fig. 1a shows the interface diagram for the EFLX-4K eFPGA logic core [7], where a large number of user I/O, clock, configuration, and test pins are available for the SoC designer to highly customize the operation of the eFPGA core. Similar to other SRAM-based FPGA technologies, the main programmable logic element in the EFLX-4K is a high-performance 6-input lookup-up-table (LUT6), as shown in Fig. 1b, which is also capable of implementing dual five-input logic functions. In addition, the EFLX-4K eFPGA logic core can be arbitrarily arrayed to create larger embedded FPGA resources [7]. The array configurations available for Sandia's 180-nm process are the 1x1, 1x2, and 2x2 configurations, as illustrated in Fig. 2. The EFLX-4K core and array configurations utilize the Synopsys Synplify tool for logic synthesis and the Flex Logix EFLX Compiler tool for logic mapping, place-and-route, timing analysis, and bitstream generation.

III. Radiation Hardened Embedded FPGA Core

The rad-hard version of the EFLX-4K eFPGA logic core, whose features are listed in TABLE I. , has been designed for Sandia's strategically radiation hardened, 7 metal layer, dual-gate oxide (3.3V/1.8V), 180-nm Silicon-on-Insulator (SOI) process. All of the eFPGA circuit elements have been hardened against radiation effects, including the configuration memory, user registers, and user memory. The radiation hardening has been implemented by process or by cell design, and is transparent to the user (i.e., no user-level scrubbing, triple-modular-redundancy, or error-detection-and-correction is required). The rad-hard EFLX-4K eFPGA core is functionally equivalent to its implementations in commercial technologies.

TABLE I. Rad-hard EFLX-4K eFPGA Features

Feature	Array Configuration		
	1x1 Array	*1x2 Array*	*2x2 Array*
Fabrication Process	Sandia 180-nm rad-hard SOI		
Operating Voltage	1.8V		
Metal Utilization	6 metal layers		
User I/O pins	632 inputs 632 outputs	760 inputs 760 outputs	1264 inputs 1264 outputs
Look-up Tables (6-input, or dual-5-input)	2520	5040	10080
Registers (flip-flops)	6304	11600	22688
Distributed Memory	20Kbits	40Kbits	80Kbits

TABLE II. Rad-hard eFPGA Performance Comparison

Benchmark	Clock Period [ns]	
	Xilinx Virtex-E[a] *[8]*	*Rad-hard EFLX-4K*[b]
16-bit counter	3.7	5.7
64-bit counter	9.6	9.2
64:1 multiplexer	6.6	5.5

[a.] XCV50e, -7 speed grade, 1.8V, 25C, Xilinx ISE tool

[b.] TT corner, 1.8V, 25C, Synopsys Synplify and Flex Logix EFLX Compiler tools

While it is often difficult to benchmark across different manufacturing processes and FPGA architectures, we have chosen to compare the performance of the rad-hard eFPGA core against a Xilinx Virtex-E device built in a commercial 180-nm process [8]. For these performance comparisons, it is important to note that the Virtex-E device uses 4-input LUTs, whereas the EFLX-4K uses 6-input LUTs. Although this can result in sizable performance differences [9], there is no known commercial FPGA device with a 6-input LUT that is implemented in a 180-nm process for which to make a more fair comparison. TABLE II. lists the clock period results for three basic benchmarks as implemented on each FPGA device type. On the 64-bit benchmarks, the rad-hard eFPGA is slightly faster than the Virtex-E due its ability to more efficiently pack logic into its 6-input LUTs and reduce its critical path. On the 16-bit benchmark, the rad-hard eFPGA is slower than the Virtex-E, partially due to the 16-bit benchmark failing to utilize 6-input LUTs, and partially due to the speed differences between the full custom layout implementation of the Virtex-E and the standard layout cell implementation of the rad-hard eFPGA. Overall, the speed performance of the rad-hard EFLX-4K eFPGA core versus a commercial FPGA device, in a similar feature size fabrication technology, is promising.

IV. DragonFly: A Reconfigurable Rad-Hard SoC

To demonstrate the usefulness of the rad-hard eFPGA core described in the previous section, we developed a reconfigurable System-on-Chip architecture called DragonFly. As illustrated in Fig. 3, the DragonFly architecture includes a 1x1 EFLX-4K eFPGA core connecting together dual ARM7TDMI-S™ processors, block SRAM, general-purpose I/O (GPIO), and high-speed serial communication nodes. One challenging SoC design aspect was determining how to split the eFPGA user I/O pins across the SoC resources. The eFPGA user I/O allocation is shown in Fig. 4a, with the majority going to block SRAM (36%), off-chip GPIO (25%), and processor interfaces (21%). A surprising amount of user I/O pins went unused (13%), and that is because interfaces like the block SRAMs used more eFPGA output pins than inputs pins, causing an imbalance in the available pin types (Fig. 4b). The chip layout of the DragonFly rad-hard reconfigurable SoC is shown in Fig. 5, and it is currently in fabrication. Additional design details on the DragonFly Soc are provided in the subsections below.

A. *eFPGA Configuration and Operation*

The eFPGA core can be programmed in multiple ways, including through an SPI (Serial Peripheral Interface) bus, JTAG (Joint Test Action Group) port, or through the processors. The configuration SPI port can be configured as a master, to read the configuration bitstream out of an external memory at power up, or a slave, to have the configuration bitstream be written by an external SPI master. The ARM7 processors can also configure the eFPGA by reading configuration data stored in their external memories. When being programmed through the processors, hardware CRC (Cyclic Redundancy Check) and software-based encryption can be enabled to ensure the eFPGA bitstream data is valid. The JTAG programming interface is intended for hardware debug and software development. Except for the processors, all the other SoC IP blocks must be held in reset until the eFPGA configuration has completed. Multiple programmable clock sources are available throughout the SoC to run individual SoC blocks at different clock rates, and all blocks, including the processors, can be disabled when they are not needed for a given application.

B. *ARM Processor Operation*

The ARM7 processors can run in lockstep or independently. They both have SPI, discrete (GPIO), and high-speed AMBA™ AHB interfaces into the eFPGA core. While the ARM7 processors have their own dedicated SRAM memory, an eFPGA application design could decide to use the dual-port SRAM blocks attached to the eFPGA core for extra data storage or buffering.

C. *Dual-port SRAM Operation*

There are three 1Kx32 dual-port SRAM blocks attached to the eFPGA core for general purpose use. These block SRAMs are fully dual ported and support multiple clock domain configurations.

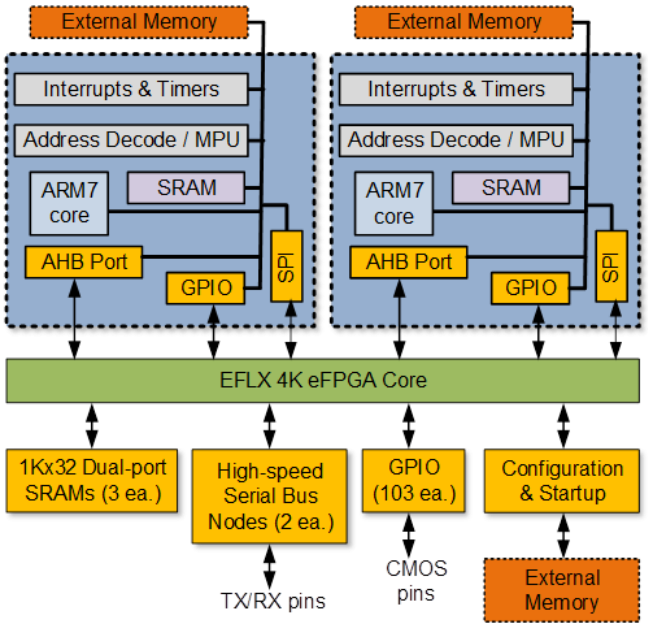
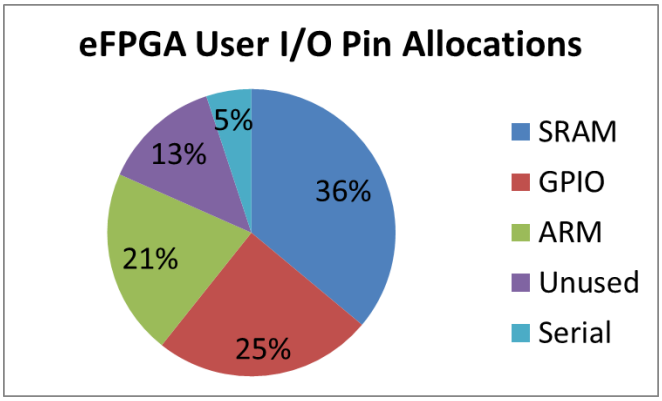
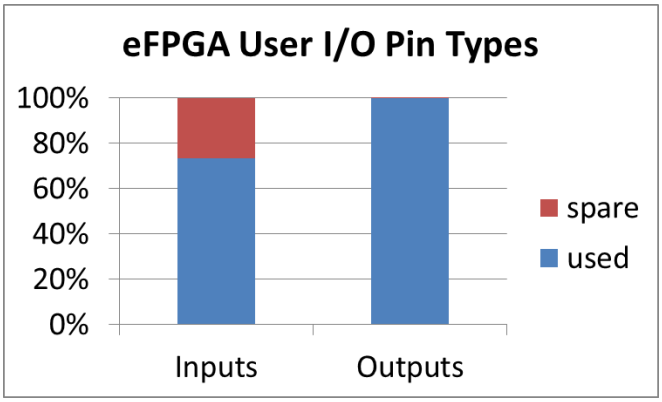


Fig. 3. DragonFly reconfigurable System-on-Chip architecture using the rad-hard 1x1 EFLX-4K eFPGA logic core.



(a)



(b)

Fig. 4. Embedded FPGA user I/O pin statistics for the DragonFly System-on-Chip: (a) user I/O pin allocations by IP block, and (b) user I/O pin allocations by type.

D. Off-chip Programmable I/O Operation

The DragonFly's 103 GPIO pins have user-programmable drive strength, slew rate, pull-up, pull-down, hysteresis, and bus-holder settings. The circuit diagram for the GPIO pin is shown in Fig. 6. While the data_in, data_out, and out_enable GPIO signals are directly controlled by the eFPGA's user I/O pins, the other control signals (drive_strength, slew_ctrl, pull_up, pull_down, input_enable, hysteresis_enable, bus_holder_enable) are statically programmed in latches located outside of the eFPGA core. This was done to reduce the number of eFPGA user I/O pins required to support each GPIO pin. During eFPGA configuration, when the GPIO pins are in an undefined state, there are user-controllable options to float the GPIO pins, pull them high, or to pull them low, which are similar to options available on commercial FPGA devices. In addition, there are two high-speed serial bus nodes attached to the eFPGA to provide LVDS differential packet communication.

V. SUMMARY

We described the design of the rad-hard EFLX-4K embedded FPGA logic core, and showed how it can be leveraged for the development of System-on-Chip applications for Sandia's 180-nm rad-hard fabrication process. These rad-hard embedded FPGA cores can significantly improve the re-configurability of rad-hard System-on-Chips, thus improving their reuse and cost savings potential for government applications.

ACKNOWLEDGMENT

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

REFERENCES

[1] The Free Library, "Atmel and Honewell Team to Develop Radiation Hardened Reconfigurable FPGAs", https://www.thefreelibrary.com/Atmel+and+Honeywell+Team+to+Develop+Radiation+Hardened+Reconfigurable...-a020935112, 1998.

[2] L. Rockett, D. Patel, S. Danziger, B. Cronquist, J.J. Wang, Radiation Hardened FPGA Technology for Space Applications, IEEE Aerospace Conference, 2007.

[3] J. Teifel, M. E. Land, R. D. Miller, Improving ASIC Reuse with Embedded FPGA Fabrics, Government Microcircuit Applications & Critical Technology Conference, 2016.

[4] C. Gordon, The Single Event Immune Reconfigurable FPGA (SIRF) Program: Development of the Radiation Harened V5QV, AIAA Infotech@Aerospace Conference, 2011.

[5] P.S. Zuchowski, C.B. Reynolds, R.J. Grupp, S.G. Davis, B. Cremen, B. Troxel, A hybrid ASIC and FPGA architecture, IEEE/ACM International Conference on Computer Aided Design, 2002.

[6] Flex Logix Technologies Inc., "eFPGA IP Density, Portability & Scalability", https://www.flex-logix.com/s/2017-11-dense-scalable-portable-eFPGA-IP.pdf, 2017.

[7] Flex Logix Technologies Inc., "T16FFC/FF+/12FFC EFLX", 2017+10+EFLX4K+TSMC16FF+FFC+12FFC+product+brief.pdf, http://www.flex-logix.com/t16ff-eflx-1, 2017.

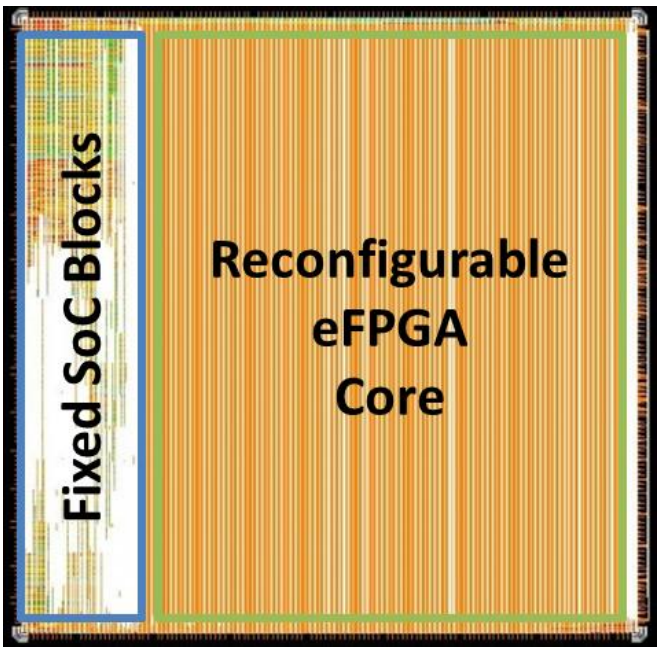


Fig. 5. Chip layout of the DragonFly rad-hard System-on-Chip, showing fixed IP blocks and a reconfigurable embedded FPGA core.

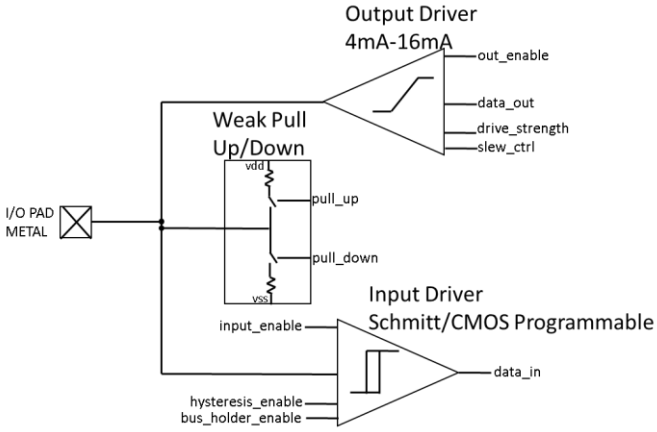


Fig. 6. User-programmable general purpose input/ouput (GPIO) circuit diagram for DragonFly rad-hard System-on-Chip.

[8] Xilinx Inc., Virtex™-E 1.8V Field Programmable Gate Arrays Production Product Specification, https://www.xilinx.com/support/documentation/data_sheets/ds022.pdf, 2014.

[9] Flex Logix Technologies Inc., "6-LUT FASTER DENSITY", http://www.flex-logix.com/6lut-faster-denser/, 2017.