Hardware Implementation of Virtual Reconfigurable Circuit for Fault Tolerant Evolvable Hardware System on FPGA

Omar E. Elnokity, Imbaby I. Mahmoud, Mohamed K. Refai, Hasan M. Farahat

Abstract- This research verify and describes a Virtual Reconfigurable Circuit (VRC) that designed and implemented for a Fault Tolerant Evolvable Hardware (EHW) system used to calculate the thermal power output of Egypt's second Training and Research Reactor (ETRR2) during operation. This circuit have three measured input signals from the reactor core: inlet temperature T_{in}, outlet temperature T_{out}, mass flow rate Q, and one output, which is the calculated thermal power. In any time the true thermal power reading should be available even one input signal get lost due to a problem in its transducer, or wire cutting, …etc. Typically, this is the function of that Fault Tolerant EHW system. The VRC design will implemented over ordinary Field Programmable Gate Array (FPGA) chip. Reducing the FPGA's configuration bits length++ is the main advantage of using VRC. Most VRCs done before used logic based function elements, while in this work, an arithmetic based elements are used, to accommodate the application nature. The design is fully synthesized on ALTERA Cyclone IV GX Family, and the design gave promising results when targeted to the EP4CGX30CF23C6 FPGA chip.

Index Terms—Egypt's second Training and Research Reactor, Evolvable Hardware, Fault Tolerant, Virtual Reconfigurable Hardware,.

I. INTRODUCTION

EHW systems combines the flexibility of computer software and the high processing performance of Hardware implementation [1]. As the programming of the FPGA changed according to the application, as the need of downloading this configuration bits into the FPGA chip. Genetic Algorithm (GA) and Application specific fitness function scoring decide the optimum configuration bits. Most of families of FPGAs configured by an external device and take relatively long time for downloading, in addition to the time needed to find the optimum bits, which finally considered as one of the significant problems of using EHW. Using VRC can help to overcome this problem by assuring a fast dynamic reconfiguration of the evolved circuit [2] utilizing conventional FPGAs (there are a new FPGA platform like Zynq platform could be partially configured, it has the potential to become the next revolutionary step in evolvable hardware design [3]).

Manuscript Received October 22, 2014.

Omar E. Elnokity, Department of System & Computer, Alazhar University, Faculty of Engineering, Cairo, Egypt.

Imbaby I. Mahmoud, Department of Engineering and Scientific Institute, Atomic Energy Authority of Egypt, Nuclear Research Center, Inshas, Egypt.

Mohamed K. Refai, Department of System & Computer, Alazhar University, Faculty of Engineering, Cairo, Egypt.

Hasan M. Farahat, Department of System & Computer, Alazhar University, Faculty of Engineering, Cairo, Egypt.

The design of VRC and its elements, explained in section 2, while sections 3, and 4 show its implementation, but section 5show the simulation results. Future work and conclusion, appears in last section.

II. THE DESIGN OF VRC

VRC is a second reconfiguration layer developed on the top of an FPGA, to reduce the length of the configuration bits and obtain fast internal reconfiguration [4]. It consists of a grid of Programmable Functional Elements (PFE) as shown in Fig. 1, the function of those elements and the interconnections between them decided by a configuration bits called chromosome, generated externally by an Evolutionary Algorithm (EA).

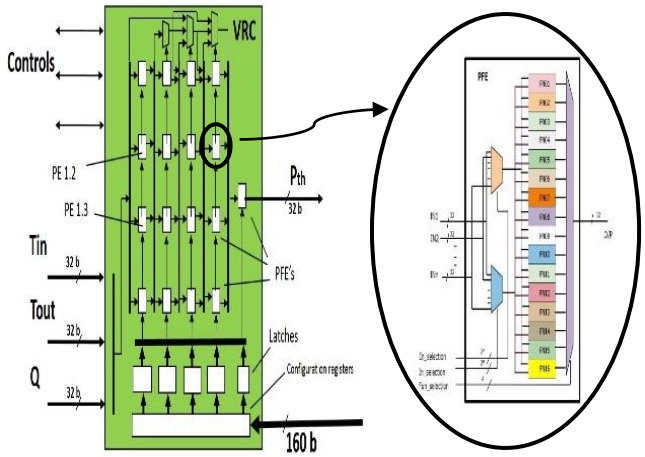

Fig. 1 Internal Diagram of VRC Utilized in the Fault Tolerant EHW System

The architecture is very similar to the representation employed in Cartesian Genetic Programming (CGP) that has been developed for circuit evolution [5]. Author in [4], reviewed the Reference research history of the VRC. Instead of download a huge number of bits to reconfigure the FPGA chip completely, a fixed structure of, "PFEs array", "shift register", "latches", and "multiplexers", will be downloaded once. In addition, in case of the need of reconfiguration, a limited number of bits should injected into the configuration register as shown in Fig. 1 to select the function of every PFE and the interconnections between them. The design consists of 4x4 array of PFEs and another one PFE for output, with 17 PFEs in total (it was planned to be 25 PFEs, but the constraints of the design aiding tools -web edition version- forced me to reduce this number). Not all the PFEs are similar, there are three types,

differs in the number of inputs only: three, four, and six inputs (PFE_3, PFE_4, PFE_6 respectively), the number of PFEs of each type are four, five, and eight respectively. There are three extra multiplexers (four, eight, and eight inputs) for interconnections between PFEs; the structure restricts the input of each PFE to come only from the PFEs of the one or two previous column at most. The configuration word length is 160 bits (4x8+5x8+8x10+2+3+3). The data word length was chosen to be 32 bits, "single precision", for the previous constraints of the design aiding tools. This design using genotype-phenotype' mapping while encoding the configuration byte word, as described in [4], i.e. every PEF have a dedicated number of bits location in the configuration byte as shown in Fig. 2.

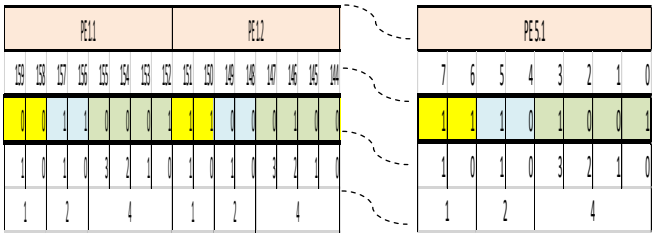

Fig. 2 Genotype-Phenotype Mapping of VRC Configuration Byte and PFEs

III. IMPLEMENTATION OF PFES

Each PFE has two multiplexer controls the connections of the two inputs with previous stage, and another one, define the function of the PFE that applied on the inputs, see Fig. 1. In table 1 it could be seen the functions used in each PFE in accordance with the 4 bits selector signal.

Table I: Functions List of Each PFE

0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
A	B	A*2	B*2	522	A/B	B/A	A*B	A-B	A+B	ABS(A)	ABS(B)	A/2	B/2	A*10	B*10

PFE circuit is fully designed, compiled, and synthesized over ALTERA, Quartus II, 64 bit, ver. 13.00, web edition. The design shown in Fig. 3 uses Altera megafunctions in arithmetic units like:

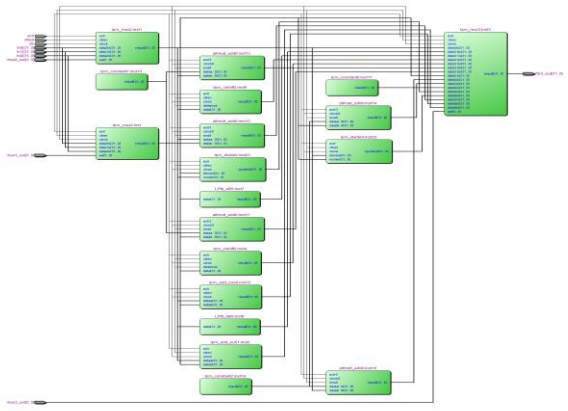

Fig. 3 Four Inputs PFE Design Structure

Multipliers, Divisors, Adders, Subtractor, Absolute, Shift Registers, and Constants. The design of this PFE and the other two PFEs are compiled and targeted to Cyclone IV GX Family chip EP4CGX30CF23C6.

As an implementation note, using the Altera's multiplexers megafunction of three inputs or six inputs, which is less than the maximum permissible number of inputs (because the selector is 2bits or 3 bits), can make troubles while simulation in 3rd party software like ModelSim. To avoid that put constants values on those not connected inputs.

IV. IMPLEMENTATION OF VRC

As mentioned in Fig. 1, VRC is an array of PFEs interconnects together through grid of controllable connections as seen in Fig. 4. The three inputs connected to a column of PFE_3 type, then the second column of PFE_4 type, the third and the fourth columns of PFE_6 type. The third and fourth columns accept inputs from previous two columns through three external multiplexers, which appears in Fig. 4 at the first row. The configuration bits of each column ordered from the output of its dedicated latch. There are five latches, one for every column, as the output PFE considered as a column. The input of each latch is feed from the 160 bits wide Shift Register (SR) output bus; this SR permits the entrance of the configuration byte to VRC. To let the other stages of EHW system knows the recent applied configuration byte, an enabled monitor module added to the design.

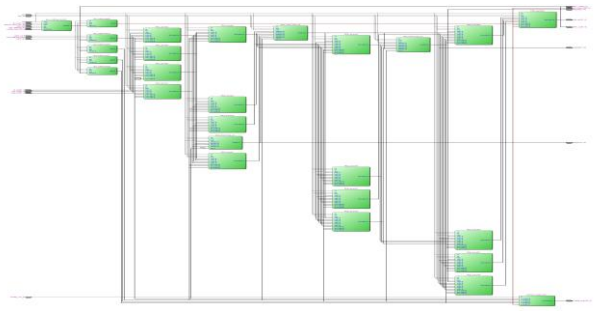

Fig. 4 Design and Implementation of VRC in ALTERA Quartus II

V. SIMULATION RESULTS FOR PFES AND VRC

ModelSim-Altera 10.1d version is used to simulate the compiled circuits. Fig. 5 shows the simulation results of PFE_3 design as an example, and valid to PFE-4, and PFE_6 modules. The output of the module matches the design criteria shown in table 1. As the results of division, operation is not so accurate due to using single precision 32 bits word, but these results could be still used properly with an acceptable percent of error, and encourage to converting all the design to a double precision 64 bits, fixed-point arithmetic, in a future work.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

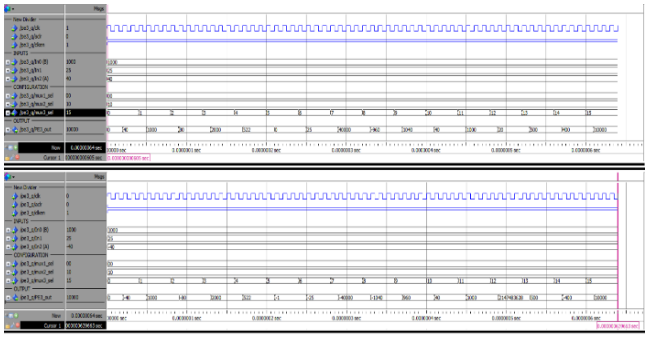

Fig. 5 Simulation Results of Implementing Three Inputs PFE

To simulate the results of VRC unit output; the following is don:

1- Form the reactor data choose a set of input values for Q, Tin, and To and mark the real thermal output of resulting from them
2- Feed those data to a hardware genetic engine build especially to these design, it will nominate the proper chromosome (configuration byte) used to configure the VRC.
3- Feed this configuration byte to VRC with the chosen input values and measure how the out thermal power reading match the desired.

As example: one chosen inputs was Q = 1000 (m^3/h), T_{in} = 25^{o}C, and T_o = 50^{o}C, those inputs according to the ETRR2 reactor data appears in our previous work [6], it should produce ~ 22 Mw. They are fed to a genetic engine to determine the configuration byte, then download the configuration byte to VRC, and apply the chosen inputs! The results was 2138500 watt, see Fig. 6, with about 0.75% error. The results appear after 180 ns of enabling it. Fig. 7 shows the internal connections tracking between the modules of VRC after applying the configuration byte to produce this result. To check the possibility to recover the VRC reading if one of the inputs lost, the last three steps is repeated again but with T_o suppressed to zero as an injected fault. The results of recovery appears in Fig. 8 is 21836000 watt, with 0.74 % of error. Fig. 9 shows the internal connections tracking between the modules of VRC after applying the reconfiguration byte. With using a single word arithmetic based calculation, it may have results with higher percent of errors, in some already tested configurations, reached to 0.9%, 1.3%, 2%, 4%, 5.8%, 7%, and 9%. It is expected that, when a fixed point based conversion -double precision word- applied to the design, it decreased to be less than ± 0.5%.

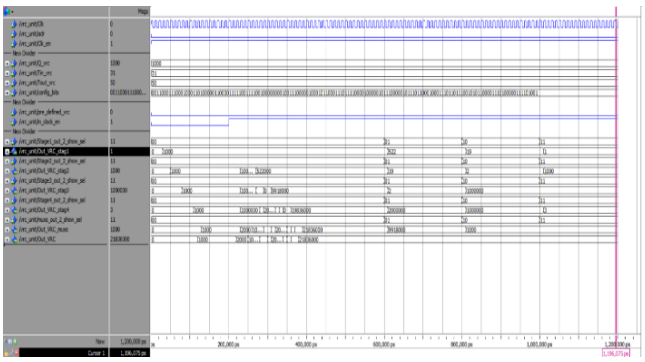

Fig. 6 Simulation Results of Applying Chosen Reactor Data Inputs into Configured VRC

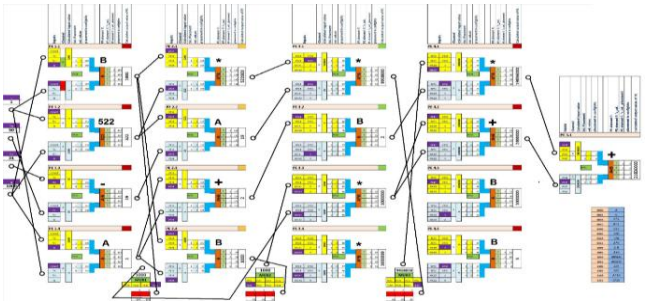

Fig. 7: Sketch of the Internal Connections Tracking between the Modules of VRC after Applying the Configuration Byte

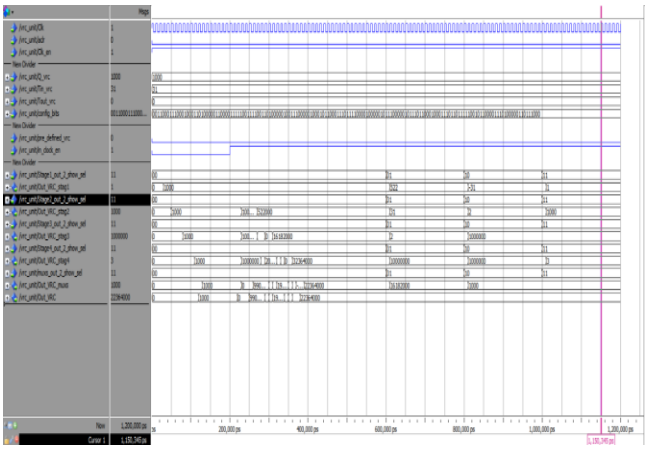

Fig. 8 Simulation Results of Applying Injecting One Data Input Fault into Reconfigured VRC

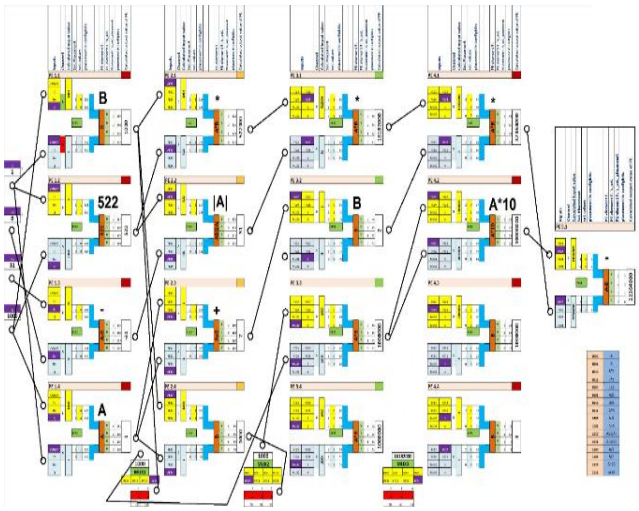

Fig. 9 Sketch of the Internal Connections Tracking between the Modules of VRC after Applying the Reconfiguration Byte

VI. EDITORIAL POLICY

In this work, promising results of VRC implementation and simulation, obtained. An ability to recover the function of VRC, demonstrated with very limited time close to real time (180 ns). A successful implementation of VRC into ordinary FPGA ALTERA EP4CGX30CF23C6 chip verified. An avenue of completing the next step after this work becomes reachable, which is build a complete Fault Tolerant Based Evolvable Hardware System applied on a nuclear process. VRC design was fully implemented and synthesized over Altera Quartus II, and

simulated using ModelSim-Altera software. The industrial standard constraints for timing performance of all logic in the design is applied. The testing data used in this work, obtained from simulating the dynamic equations of ETRR2 with its specific parameters.

REFERENCES

1. D.Dhanasekaran, K. Boopathy Bagan, E.Ravi, "Fault Tolerant System Design using Evolved Virtual Reconfigurable Circuit,"IJCSNS International Journal of Computer Science and Network Security, vol. 6 No. 5A, pp. 64-72, May 2006.
2. Chu Jie, Man Meng-Huam Liu Shang-He, Wei Miang, Yuan Liang, Ju Zheng Quan, Chang Xiao-Long, "Self-Recovery of Motor Control Circuit Based on MFNNVRC," American Journal of Engineering and Technology Research, Vol. 11, No. 9, pp. 2589-2593, 2011.
3. Roland Dobai, Lukas Sekanina," Towards Evolvable Systems Based on the Xilinx Zynq Platform," Evolvable Systems (ICES), 2013 IEEE International Conference, IEEE, 10.1109/ICES.2013.6613287, pp. 89-95, April 2013.
4. Wang, J., Incheon, Chen, Q.S., Lee, C.H, "Design and implementation of a virtual reconfigurable architecture for different applications of intrinsic evolvable hardware," IET Comput. Digit. Tech., Vol. 2, No. 5, pp. 386-400, 2008.
5. Sekanina, L., Friedl, S., "On routine implementation of virtual evolvable devices using COMBO6," Evolvable Hardware, 2004. Proceedings. 2004 NASA/DoD Conference, IEEE, 10.1109/EH.2004.1310810, pp. 63-70, June 2004.
6. Omar Elnokity, Imbaby I. Mahmoud, Mohamed K. Refai, Hasan M. Farahat, "ANN based Sensor Faults Detection, Isolation, and Reading Estimates – SFDIRE: Applied in a nuclear process," Annals of Nuclear Energy, Vol. 49, pp. 131-142, August 2012.

AUTHORS PROFILE

Omar E. Elnokity, Lecturer, PhD candidate, system & computer Dept., Alazhar University/ Faculty of Engineering, Cairo, Egypt

Imbaby I. Mahmoud, Professor Atomic Energy Authority of Egypt, Engineering and scientific Ins. Dept., Nuclear Research Center, Inshas, Egypt

Mohamed K. Refai, Professor Alazhar University, Faculty of Engineering, system & computer Dept.,Cairo, Egypt.

Hasan M. Farahat, Professor Alazhar University, Faculty of Engineering, system & computer Dept.,Cairo, Egypt.