

REPOMO32 – New Reconfigurable Polymorphic Integrated Circuit for Adaptive Hardware

Lukas Sekanina, Richard Ruzicka, Zdenek Vasicek, Roman Prokop and Lukas Fucik

Abstract—In this paper, a new reconfigurable polymorphic chip (REPOMO32) is introduced. This chip has been developed in order to investigate the electrical properties of polymorphic circuits and demonstrate the applications of polymorphic electronics. REPOMO32 contains an array of 32 configurable logic elements; each of them can perform the AND, OR, XOR and polymorphic NAND/NOR function which is controlled by the level of the power supply voltage. REPOMO32 parameters are reported together with the analysis of polymorphic circuits implemented and evolved in REPOMO32. Potential applications of the chip are also discussed.

I. INTRODUCTION

Current work in the field of evolvable hardware can be split into the two related areas of evolutionary hardware design and adaptive hardware. While evolutionary hardware design is the use of evolutionary algorithms (EAs) for creating innovative (and sometimes patentable) physical designs, the goal of adaptive hardware is to endow physical systems with some adaptive characteristics in order to allow them to operate successfully in a changing environment or under the presence of faults. Experiments with the intrinsic evolution (i.e. when all candidate circuits are evaluated in a physical device) have shown that in order to obtain required behavior the evolution can exploit some unexpected properties of the physical device and environment. This is unreachable by means of conventional design techniques [1], [2], [3], [4]. However, novel implementations of target functions can be created. In some cases, evolved solution can also act as a new kind of sensor. For example, Bird and Layzell evolved a network of transistors sensing and utilizing the radio waves emanating from nearby PCs [5]. More advanced sensing capabilities were demonstrated in evolved polymorphic gates which are able to change the logic function according to the status of external environment [6], [7], [8], [9]. For example, the polymorphic AND/OR gate performs the AND function in its first mode (when the first environment is present, for example, temperature is 27 °C) and the OR function in its second mode (when the second environment is present, for example, temperature is 125 °C). Polymorphic gates can be considered as a new reconfigurable technology capable

Lukas Sekanina, Richard Ruzicka and Zdenek Vasicek are with the Faculty of Information Technology, Brno University of Technology, Czech Republic (email: {sekanina, ruzicka, vasicek}@fit.vutbr.cz).

Roman Prokop and Lukas Fucik are with the Faculty of Electrical Engineering and Communication, Brno University of Technology, Czech Republic (email: {prokop, fucik}@feec.vutbr.cz).

This work was partially supported by the Grant Agency of the Czech Republic under contract No. 102/06/0599 *Methods of polymorphic digital circuit design* and the Research Plan No. MSM 0021630528 – *Security-Oriented Research in Information Technology*. Cadence software was used with support through the Cadence Academic Network.

of integrating the logic function with sensing in a single compact structure.

Behavior of the polymorphic gates was demonstrated using simulations. The first example of fabricated polymorphic gate – the NAND/NOR gate controlled by power supply voltage (Vdd) – was presented by Stoica's group [8]. The six-transistor NAND/NOR gate operates as NOR when Vdd = 1.8 V and as NAND when Vdd = 3.3 V. The control of logic function via Vdd is unconventional but interesting for many applications. The HP 0.5 micron CMOS technology was used for fabrication. This gate exhibits a very unconventional structure; in particular, it does not follow the well-accepted rules for transistor-level design of CMOS gates. Then, researchers have begun to develop the methods for polymorphic circuits synthesis and integrate the polymorphic gates to various digital circuits to enhance their functionality [10], [11], [12], [13], [14], [15], [16]. Almost exclusively, the experiments were done using simulators.

Another NAND/NOR gate controlled by Vdd was developed and characterized by FIT (Faculty of Information Technology) Evolvable Hardware Group [17]. The FIT gate was fabricated using AMIS CMOS 0.7 micron technology. In addition to some differences in electrical properties, it significantly differs from Stoica's gate in the fundamental concept of required behavior. Stoica's NAND/NOR gate operates with identical voltage levels for logic signal 1 in both modes, i.e. the logic 1 is represented using 1.8 V independently whether Vdd = 1.8 V or Vdd = 3.3 V (note that the logic 0 is always at 0 V). On the other hand, the FIT gate moves logic levels with changing Vdd. When Vdd = 3.3 V, the logic 1 is represented by 3.3 V. When Vdd = 5V, the logic 1 is considered as 5 V. This property implies a different class of applications for proposed gate. The application class assumes that all circuit components (polymorphic as well as ordinary gates) are always connected to the same power supply unit. Hence ordinary gates must be able to perform correctly for Vdd = 3–5 V as well as for the logic 1 represented as 3–5 V.

Circuits composed of polymorphic and ordinary gates can be simulated in software (e.g., PSpice). However, only the physical implementation can really show and uncover all circuit properties, and possibly, serve as the proof of the concept for polymorphic electronics. In particular, it is important to investigate the effect of logic function transitions on the behavior of polymorphic circuits. Performing these experiments is impractical without using a suitable reconfigurable polymorphic chip and user-friendly interface.

The goal of this paper is to present a new reconfigurable polymorphic circuit REPOMO32 (REconfigurable POLymor-

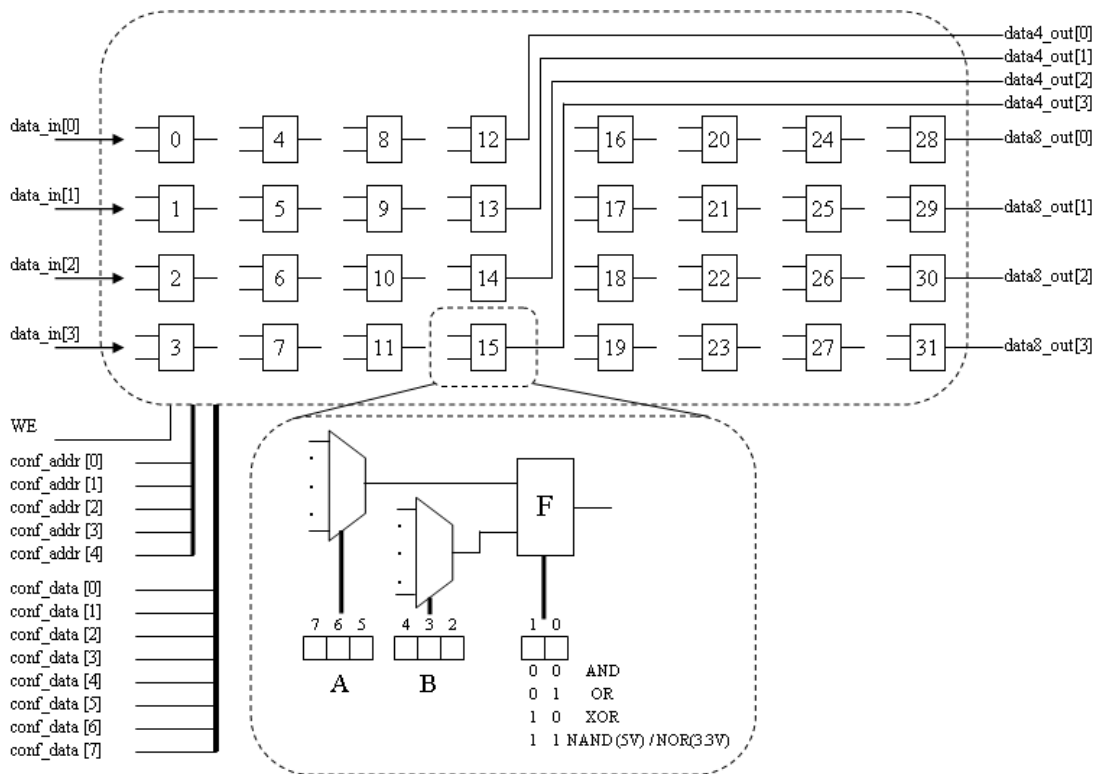


Fig. 1. Architecture of REPOMO32

pic Module) which was developed in order to investigate the electrical properties of polymorphic circuits and demonstrate the applications of polymorphic electronics. REPOMO32 contains an array of 32 configurable logic elements (CLE). In addition to ordinary gates, every CLE contains one NAND/NOR gate controlled by V_{dd} . Having 4 inputs and 8 outputs the REPOMO32 is suitable for creating small combinational polymorphic circuits. The REPOMO32 was fabricated in AMIS CMOS 0.7 micron technology.

In order to provide a user-friendly interface, the REPOMO32 is connected to the FITkit board [18] controlled by a common PC. FITkit is equipped with a small FPGA, microcontroller and various peripheral devices. FITkit is responsible for configuring the REPOMO32, generating the input signals and reading the output values. FITkit also controls the configurable power supply which generates V_{dd} for REPOMO32. Evolutionary design of polymorphic circuits is performed in such a way that while the evolutionary algorithm runs in the PC, candidate circuits are evaluated in the REPOMO32. Section II describes the architecture of REPOMO32 and the connection of the chip to PC via FITkit.

As the REPOMO32 is the first reconfigurable polymorphic circuit developed so far, it allows us to investigate properties of polymorphic electronics in a quite new way. We will show that using the chip it is possible to intrinsically evolve polymorphic circuits for given specifications. Section III-A summarizes the results of experiments arranged in order to obtain the basic parameters of the chip, for example, delays of the gates, reconfiguration time etc. In Section III-B and III-

C, the experimental evaluation of two polymorphic circuits implemented in REPOMO32 is presented. Obtained results and potential applications of REPOMO32 are discussed in Section IV. Conclusions are given in Section V.

II. REPOMO32 DESCRIPTION

The initial model of REPOMO32 (denoted as REPOMO) was presented in [19]. Then, its architecture and reconfiguration options were further analyzed [20]. On the basis of these investigations the final architecture of REPOMO32 was defined and the chip was fabricated in 2008.

A. Architecture Overview

Figure 1 shows the structure of the chip. The 32 two-input CLEs are organized in array of 4 rows and 8 columns. Each of them can be programmed to perform one of the following functions: AND, OR, XOR and polymorphic NAND/NOR (see block F in Figure 1). While the AND, OR and XOR do not change their logic functions with V_{dd} , the function of the NAND/NOR is directly controlled by V_{dd} . REPOMO32's logic behavior is defined by its configuration bits and the level of V_{dd} . The configuration bits control a set of multiplexers which are responsible for interconnecting the CLEs and selecting their logic functions. CLE's input can be connected to the output of a CLE placed somewhere in the two preceding columns. CLEs located in the column 1 or 2 can be connected to the primary inputs (`data_in`). The selection of the source point is performed using a multiplexer (3 bits for each CLE input). In total, 8 bits define the configuration

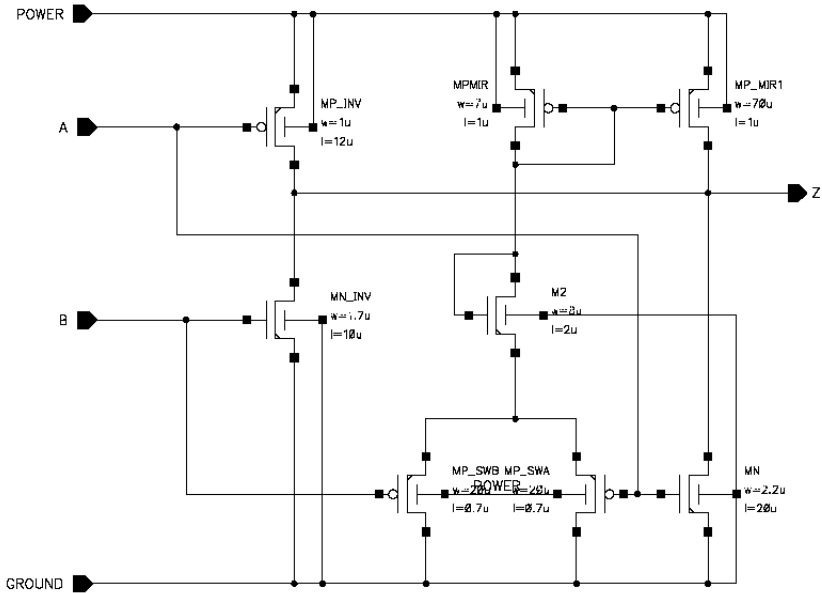
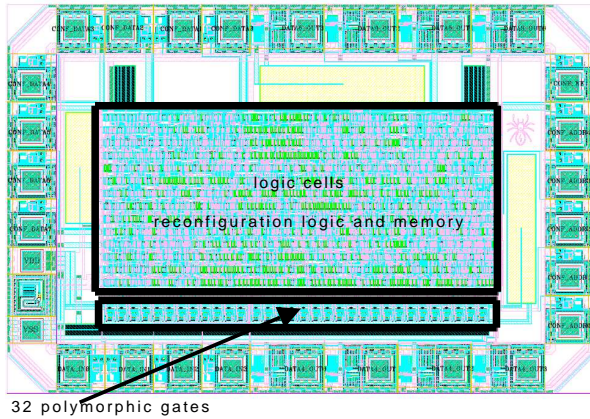


Fig. 2. Polymorphic NAND/NOR gate: the NOR when $V_{dd} = 3.3 \text{ V}$ and the NAND when $V_{dd} = 5 \text{ V}$



32 polymorphic gates

Fig. 3. Layout of REPOMO32

of a single CLE. The configuration of the chip is stored in 32 8-bit latch registers. The configuration of a single CLE is performed by supplying CLE's address ($conf_addr$) and configuration data ($conf_data$) followed by activating the WE signal. The chip can be completely reconfigured in 32 configuration steps. While the $data4_out$ outputs are connected directly to CLEs of the fourth column the $data8_out$ outputs are connected directly to CLEs of the last column. There are no synchronization registers in REPOMO32. The chip has 28 pins and occupies the area of $2900 \times 1970 \text{ um}$. Figure 3 shows the layout of the chip.

B. Polymorphic Gate

Figure 2 shows the NAND/NOR gate utilized in REPOMO32. When $V_{dd} = 3.3\text{V}$, the gate exhibits the NOR function and when $V_{dd} = 5\text{V}$, the gate exhibits the NAND function. Figure 4 shows the behavior of the gate for

two different levels of V_{dd} . Its detailed description and characterization is given in [17]. As the gate is crucial for the REPOMO32 behavior, we will briefly describe its implementation and properties.

First two transistors on inputs A and B serve as an inverter and guarantee the correct output logic levels for the 00 and 11 input combinations (note that the NOR and NAND logic functions give identical output values for these combinations). The difference in the gate behavior comes when A and B inputs differ. Then, the output level is 0 for the NOR function and 1 for the NAND function. In the NOR mode, the output level 0 is achieved by opened MN transistor (or MN_INV transistor respectively). As transistor MP_MIR1 is closed, V_{dd} is not sufficient to open MP_MIR and M2 transistors. In the NAND mode, the output level 1 is ensured by opened MP_MIR1 transistor. This transistor is much larger than all MN transistors and thus overrides their 0es. In Figure 5, the output voltage is shown for different V_{dd} s and the input combination "10". In the range of $V_{dd} = 0\text{--}3.8 \text{ V}$, the gate exhibits the NOR function (logic 0 at the output). For V_{dd} higher than 3.9 V , the gate exhibits the NAND function – the output is at logic 1 (the output voltage is close to V_{dd}). It can be seen that there is a very small hysteresis.

Measured characteristics show that the logic function is changed around $V_{dd} = 3.8 \text{ V}$. It is stable for V_{dd} below 3.7 V and over 3.9 V . The function instability range is adequately small and reduced by hysteresis. The absolute maximum ratings of the gate (shown in Table I) are inferred from characteristics of used CMOS technology.

Table II summarizes DC electrical characteristics of the gate. Because the gate exhibits two functions controlled by V_{dd} , these characteristics were measured for both levels of

TABLE I
ABSOLUTE MAXIMUM RATINGS OF THE NAND/NOR GATE

	min.	max.
Vdd	-0.5 V	7 V
input/output voltage	-0.5 V	Vdd + 0.5 V
operating temp.	0°C	+70°C
junction temp.	-40°C	+125°C

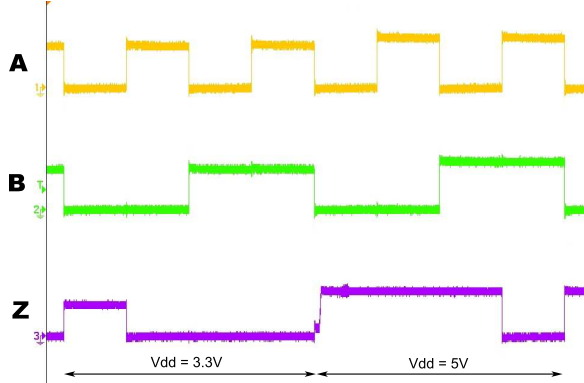


Fig. 4. Behavior of polymorphic NAND/NOR gate measured at 5 kHz (NOR for Vdd = 3.3 V, NAND for Vdd = 5 V)

Vdd. It can be seen that the input threshold voltage moves from approx. 1V in the NOR mode (Vdd = 3.3 V) to approx. 1.45 V in the NAND mode (Vdd = 5 V). Also the output logic levels are moved with Vdd (see Table II).

Table III summarizes dynamic characteristics of the polymorphic gate. Delays depend on the gate mode. Especially in the NAND mode, the delays of rising and falling edges differ by an order of magnitude. This difference is caused by different physical sizes of used transistors.

C. Interface to REPOMO32

In order to automate the process of configuration, setting the inputs and reading the outputs, the REPOMO32 was connected to the configuration/evaluation controller implemented in the FPGA (Xilinx FPGA Spartan 3 XC3S50) which is available at the FITkit board (Figure 6). The circuits of

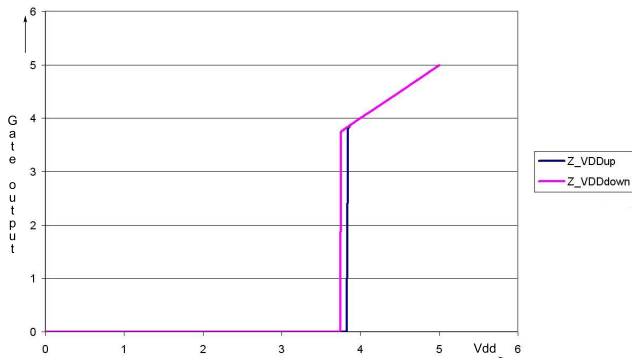


Fig. 5. Output voltage for input = "10" and different Vdd

TABLE II
DC ELECTRICAL CHARACTERISTICS OF THE NAND/NOR GATE

	Vdd = 3.3 V	Vdd = 5 V
V_{iL}	max. 0.79 V	max. 1.09 V
V_{iH}	min. 1.37 V	min. 1.78 V
V_{oL}	0.0 V	0.0 V
V_{oH}	3.29 V	4.97 V
f_{max}	38.6 MHz	56.2 MHz

TABLE III
DYNAMIC CHARACTERISTICS OF THE NAND/NOR GATE

	Vdd = 3.3 V	Vdd = 5 V
A to Z t_{pLH}	64 ns	2.8 ns
A to Z t_{pHL}	56 ns	24.4 ns
B to Z t_{pLH}	81 ns	3.0 ns
B to Z t_{pHL}	71 ns	26.4 ns

the FPGA work at 6 MHz. In our current implementation, REPOMO32 is directly connected to the FPGA which works at 3.3 V. It was verified that this voltage level of the input signals is sufficient to drive the REPOMO32 chip working with Vdd = 3.3 – 5V.

Software for PC was developed that can communicate with the FPGA using USB standard and so to generate configurations for the REPOMO32 and control the REPOMO32. The FITkit was chosen because it provides required interfaces and satisfies the requirements on the communication throughput. In addition, as the FITkit is used at FIT for the research and teaching purposes (more than 1000 pieces in use) a well established technical support is granted.

FITkit also controls the Vdd signal for REPOMO32. In order to trustworthy "simulate" real-world changes in Vdd and to provide the application-specific control of Vdd, it is necessary to generate not only two-level Vdd (i.e. 3.3 V and 5 V), but also Vdd which can be changed slowly between several voltage levels. That's achieved by a programmable power supply voltage.

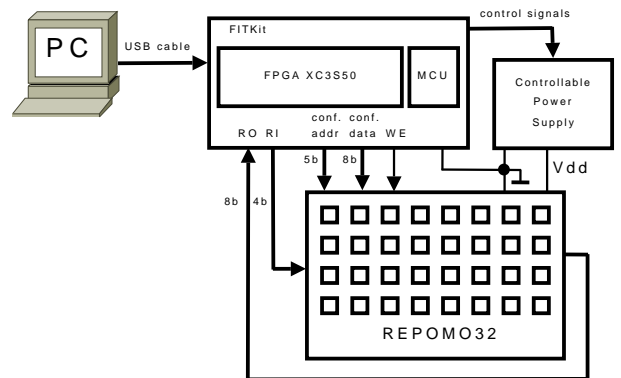


Fig. 6. Evaluation testbed for REPOMO32

D. Intrinsic Evolution in REPOMO32

Evolutionary design of polymorphic circuits is performed using Cartesian Genetic Programming (CGP) [15]. In fact, the REPOMO32 architecture resembles the representation used in CGP (with topology 8 x 4 nodes). The CGP is implemented in software; however, candidate circuits are evaluated in REPOMO32 and the fitness value is calculated in the FPGA. Every chromosome is sent to the FPGA which reconfigures REPOMO32 on the basis of the chromosome content. The FPGA also generates the input vectors for REPOMO32, collects responses from REPOMO32, calculates the fitness value and reports the fitness value to the PC. In our current implementation, the evaluation of a single test vector requires 1.3 us. A single candidate circuit can be evaluated in $2.2^4 \times 1.3$ us (if both modes of the NAND/NOR gates are considered for all possible input combinations). This is a sufficient time for PC to prepare a new candidate circuit (configuration) and send it to the FPGA (which is then sent to the REPOMO32 chip).

III. FIRST EXPERIMENTS

This section provides basic electrical parameters which were obtained by measuring the REPOMO32. Then, two polymorphic circuits implemented using REPOMO32 are presented. While the first one, self-checking adder, works correctly independently of the setting of polymorphic gates mode, the behavior of the second circuit, 3-bit parity/majority, depends on the Vdd level. In both cases, circuit responses were examined especially during Vdd transitions. In order to measure REPOMO32 parameters, we used Agilent 16823A Logic Analyzer and Agilent Infiniium DSA90254A Oscilloscope.

A. Parameters of REPOMO32

Because REPOMO32 was manufactured using the same technology as the NAND/NOR gate, the absolute maximum ratings are the same as for the NAND/NOR gate. They are inferred from used CMOS technology and shown in Table I. Also DC electrical characteristics, i.e. V_{iL} , V_{oL} , V_{iH} and V_{oH} are identical with that of the NAND/NOR gate. Selected dynamic parameters of the REPOMO32 are shown in Table IV. It can be seen that maximum operating frequency (or input-to-output delay, t_{pd}) depends on Vdd and the gates employed in the configuration. The “4 gates / 1 poly” means that a four-gate path is measured - from data_in to data4_out - and 3 ordinary AND gates and 1 polymorphic NAND/NOR gate are employed. While t_{pd} of the ordinary gate (e.g. AND) is about 10 ns (this value is relatively independent of Vdd), t_{pd} of the polymorphic gate varies from units to nearly hundred nanoseconds.

In Table V, some other parameters are shown and compared to one of the most famous small reconfigurable CMOS logic devices - Lattice GAL20V8. Operating power supply current I_{cc} of REPOMO32 is approx. $1/2I_{cc}$ of the GAL for the same input toggling frequency $f_{toggle} = 15$ MHz. This is a very good result with respect to the structure of

TABLE IV
MAXIMUM OPERATING FREQUENCY OF REPOMO32 ACCORDING TO VDD AND CIRCUIT CONFIGURATION

	Vdd = 3.3 V	Vdd = 5 V
4 gates / 1 poly	8.03 MHz	15.39 MHz
8 gates / 1 poly	5.56 MHz	10.47 MHz
4 gates / 3 poly	3.27 MHz	9.94 MHz
8 gates / 3 poly	2.50 MHz	7.73 MHz
8 gates / 5 poly	1.53 MHz	6.07 MHz
8 gates / 7 poly	1.12 MHz	5.10 MHz

TABLE V
REPOMO32 PARAMETERS IN COMPARISON WITH LATTICE GAL20V8

	REPOMO32	GAL20V8
I_{cc} at $f_{toggle} = 15$ MHz	24 mA	45 mA
t_{pd}	40-100 ns	5-25 ns
f_{max}	5.10-15.39 MHz	166 MHz

used polymorphic gates. Maximum operating frequency of REPOMO32 is lower due to the structure of polymorphic gates and sizes of used transistors. Also, the delay between the activation of the configuration Write Enable (WE) signal and the corresponding CLE reconfiguration was measured. The delay is 33 ns at Vdd = 5 V and 50 ns at Vdd = 3.3 V.

Figure 7 shows some of REPOMO32 signals during the permanent reconfiguration of CLE0 function (cyclic changes between the AND function and NOR function; no changes in Vdd allowed). The output of CLE0 is denoted out0; the first input of CLE0 is denoted in0; the second input is permanently at high (not shown). It can be seen that when configuration bit *conf*, which controls the function of CLE0, is high the NOR gate is employed - 0 NOR 1 = 0, 1 NOR 1 = 0, when *conf* bit is low the AND gate is employed - 0 AND 1 = 0, 1 AND 1 = 1. The period of reconfiguration (i.e., the period of WE signal in Fig. 7) is 3 us.

B. Self-Checking Adder

Figure 9 shows the self-checking adder that we investigated in our previous work [17]. The adder consists of 3 NAND/NORs, 2 XORs and inverter. Independently of the level of Vdd, this logic network always generates a correct

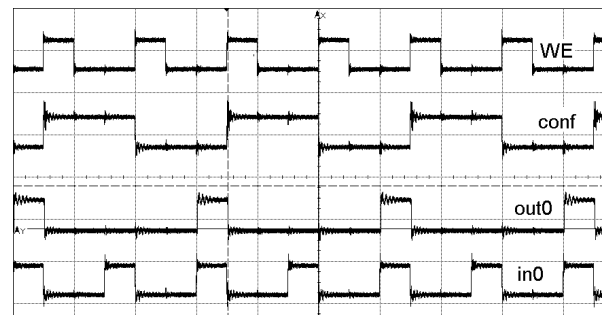


Fig. 7. Reconfiguration of CLE0 when Vdd is stable (3.3 V). The first input of CLE is denoted in0, the second input is permanently at high (not shown) and the output is denoted out0. The WE period is 3 us

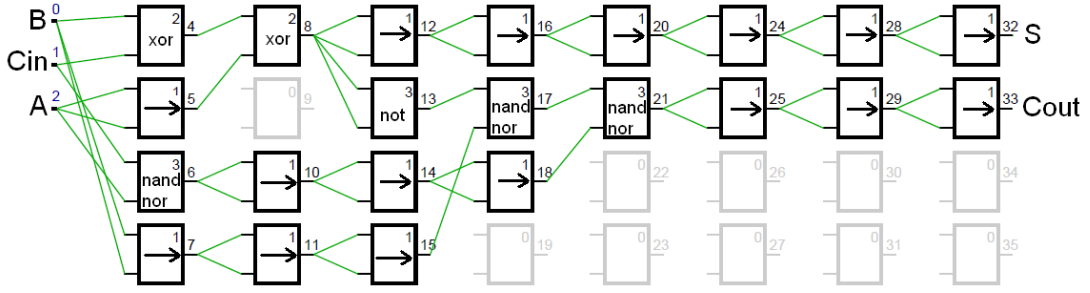


Fig. 8. Self-checking polymorphic adder in REPOMO32

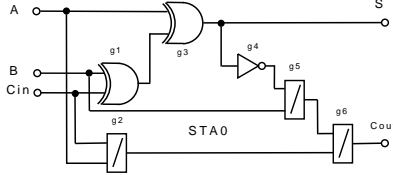


Fig. 9. Self-checking polymorphic adder

carry-out signal when there is no fault present in the circuit. However, when the stuck-at-fault is present within the adder then Cout output will oscillate between 0 and 1 at the same frequency as Vdd oscillates. In addition to its primary function, this output works as the indicator of a stuck-at-fault in the circuit. Vdd is switched whenever the circuit should be tested. Figure 8 shows an implementation of the self-checking adder in REPOMO32. Note that the OR gates with interconnected inputs serve as “wires” (see the arrows in Figure 8). The circuit behavior is shown in Figure 10. We can observe that S and Cout are correct (and identical) for both levels of Vdd. Last two signals (denoted with *) show the situation at S and Cout when a fault (stuck-at-1) is injected at the output of CLE0. Because of that fault, differences are visible at Cout when Vdd is changed.

C. Evolution of 3-bit Parity/Majority

CGP operating with 4x8 programmable elements and the 5-member population is used to evolve the 3-bit parity/majority circuit. All candidate circuits are evaluated in REPOMO32. The mutation operator randomly modifies 2 genes of the chromosome. When a perfect behavior is achieved in both modes (i.e. parity for Vdd = 3.3 V and majority for Vdd = 5 V), the process of minimizing of the number of CLEs is started. On average 875 generations are needed to find a working solution (calculated from 10 independent runs). Figure 11 shows two examples of evolved solutions. Their responses are shown in Figure 12AB for both Vdd levels.

IV. DISCUSSION

A. Summary of Experiments

Previous approaches to evolution of the polymorphic combinational circuits were based on CGP running as a program on PC [15]. Polymorphic gates were simulated using logic

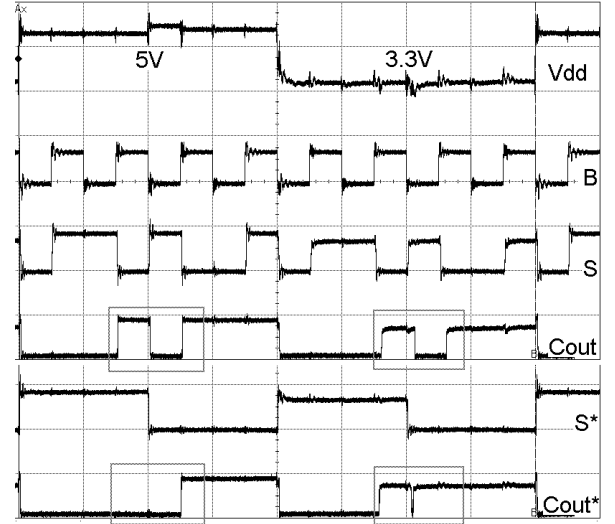


Fig. 10. Behavior of self-checking polymorphic adder ($T_B = 1$ us, $T_{Cin} = T_B/2$, $T_A = T_B/4$): a) no fault (S, Cout), b) stuck-at-fault visible as differences at Cout* when Vdd is changed. Only the least significant input bit (B) is shown.

expressions, i.e. considered as ideal digital circuits. The circuit evolution in software is relatively fast and can be even accelerated in, for example, FPGA [21]. In contrast, proposed approach allows designing and evaluating real polymorphic circuits. However, it is slow because the REPOMO32 is slow wrt fast (but simplified) software simulations.

The parity/majority circuits shown in Figure 11 are logically correct. However, it can be seen from Fig. 12 that circuit B does not exhibit so many glitches at the output as circuit A, i.e. circuit B would be better for practical use. Note that this observation could not be made only on the basis of the evaluation used in CGP. Hence the measurement on the real hardware is important.

This paper contains initial results that were obtained using REPOMO32 connected to FITkit via a breadboard. For next research, it is assumed that REPOMO32 will be connected using a PCB and a better input signal generator will be used. By optimizing the REPOMO32 controller in FPGA and the communication scheme with PC, it would be possible to decrease the time of evolution by 1-2 orders.

Because of REPOMO32’s low speed and high communication latency, it is practical to use the software implementation

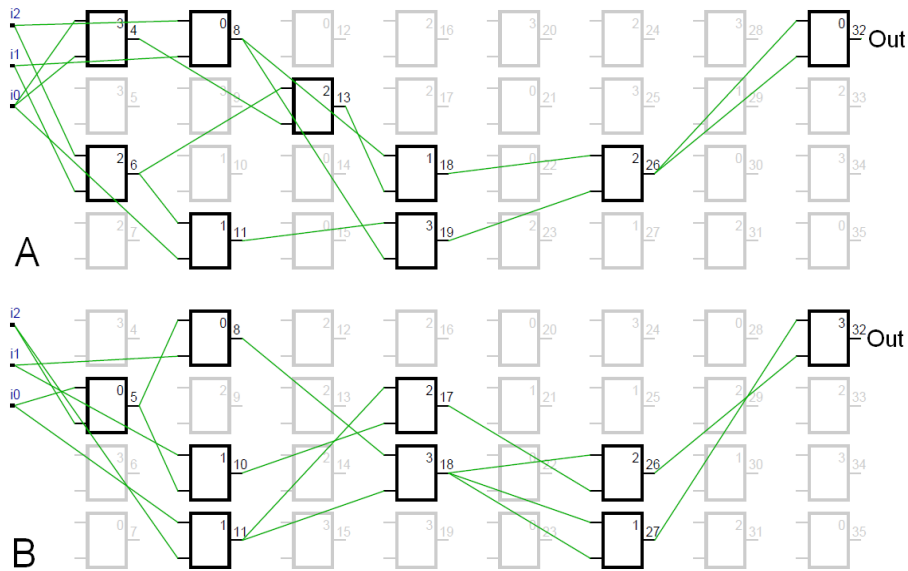


Fig. 11. Two evolved polymorphic 3-bit parity/majority circuits

of CGP (or the FPGA accelerator) to infer topologies of target circuits. REPOMO32 will be used to validate the resulting/evolved circuits in a real environment.

B. Potential Applications

Potential applications of polymorphic electronics were surveyed in [6], [7]. The REPOMO32 is considered for evaluation of real-world applications of polymorphic electronics. It can also be utilized as a reusable module for prototyping polymorphic electronics. In particular, we plan to employ REPOMO32 in the area of circuit diagnostics and security. The “second” function provided by polymorphic gates can be activated in test mode of a circuit in order to detect faults [11] or reduce the number of test vectors [10], [16]. In the area of security, polymorphic gates can implement “invisible” functions which can be activated under special conditions. Polymorphic gates can also work as sensors providing the information about the particular Vdd, system status or environment. Finally, fast reconfiguration of a reconfigurable device can be achieved using polymorphic gates controlled by Vdd because Vdd can serve as a global configuration signal.

The results presented in this paper have shown some potential problems of polymorphic electronics, in particular, the undesirable transient phenomena and higher power consumption. It seems now that the potential applications of polymorphic electronics should not change the mode of polymorphic gates frequently. The “second” mode should be activated only under specific conditions in order to cheaply perform some additional computation whose standard implementation is expensive using a separate circuit.

V. CONCLUSIONS

In this paper, a first reconfigurable polymorphic chip was introduced. Although the chip is relatively small, it

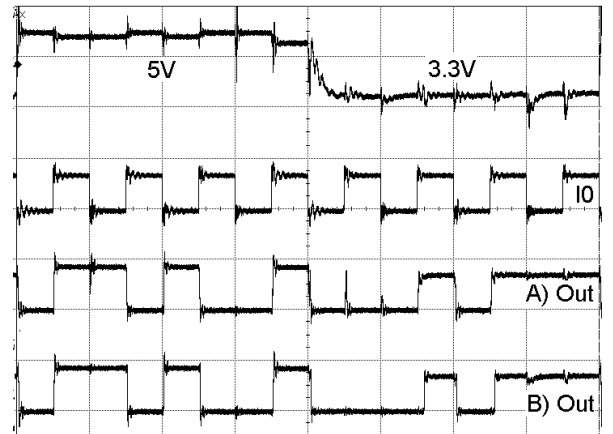


Fig. 12. Behavior of two parity/majority circuits (at 1 MHz). Only the least significant input bit (i0) is shown.

allows validating various combinational polymorphic circuits controlled by Vdd in a real environment. We presented electrical parameters of the chip and several polymorphic circuits developed using the proposed user interface. Our future research will be devoted to improving the platform and prototyping applications of polymorphic electronics using REPOMO32.

REFERENCES

- [1] A. Thompson, “Silicon Evolution,” in *Proc. of Genetic Programming GP’96*. MIT Press, 1996, pp. 444–452.
- [2] P. J. Layzell, “A new research tool for intrinsic hardware evolution,” in *Evolvable Systems: From Biology to Hardware*, ser. LNCS, vol. 1478. Springer, 1998, pp. 47–56.
- [3] A. Stoica, R. S. Zebulum, M. I. Ferguson, D. Keymeulen, and V. Duong, “Evolving circuits in seconds: Experiments with a stand-alone board-level evolvable system,” in *Proceedings of the 2002 NASA/DoD Conference on Evolvable Hardware (EH’02)*. Washington, DC, USA: IEEE Computer Society, 2002, pp. 67–64.

- [4] S. L. Harding, J. F. Miller, and E. A. Rietman, "Evolution in materio: Exploiting the physics of materials for computation," *Journal of Unconventional Computing*, vol. 4, no. 2, pp. 155–194, 2008.
- [5] J. Bird and P. Layzell, "The evolved radio and its implications for modelling the evolution of novel sensors," in *Proceedings of the Evolutionary Computation Congress*. Washington, DC, USA: IEEE Computer Society, 2002.
- [6] A. Stoica, R. S. Zebulum, and D. Keymeulen, "Polymorphic electronics," in *Proc. of Evolvable Systems: From Biology to Hardware Conference*, ser. LNCS, vol. 2210. Springer, 2001, pp. 291–302.
- [7] A. Stoica, R. S. Zebulum, D. Keymeulen, and J. Lohn, "On polymorphic circuits and their design using evolutionary algorithms," in *Proc. of IASTED International Conference on Applied Informatics AI2002*, Innsbruck, Austria, 2002.
- [8] A. Stoica, R. Zebulum, X. Guo, D. Keymeulen, I. Ferguson, and V. Duong, "Taking Evolutionary Circuit Design From Experimentation to Implementation: Some Useful Techniques and a Silicon Demonstration," *IEE Proc.-Comp. Digit. Tech.*, vol. 151, no. 4, pp. 295–300, 2004.
- [9] R. S. Zebulum and A. Stoica, "Four-Function Logic Gate Controlled by Analog Voltage," *NASA Tech Briefs*, vol. 30, no. 3, p. 8, 2006.
- [10] L. Sekanina, L. Starecek, Z. Kotasek, and Z. Gajda, "Polymorphic Gates in Design and Test of Digital Circuits," *Int. J. Unconventional Computing*, vol. 4(2), pp. 125–142, 2008.
- [11] L. Sekanina, "Evolution of Polymorphic Self-Checking Circuits," in *Proc. of the 7th Conf. on Evolvable Systems: From Biology to Hardware*, ser. LNCS, no. 4684. Wuhan, China: Springer, 2007, pp. 186–197.
- [12] R. S. Zebulum and A. Stoica, "Multifunctional Logic Gates for Built-In Self-Testing," *NASA Tech Briefs*, vol. 30, no. 3, p. 10, 2006.
- [13] W. Luo, Z. Zhang, and X. Wang, "Designing polymorphic circuits with polymorphic gates: a general design approach," *IET Circuits, Devices & Systems*, vol. 1, no. 6, pp. 470–476, 2007.
- [14] R. S. Zebulum and A. Stoica, "Ripple Counters Controlled by Analog Voltage," *NASA Tech Briefs*, vol. 30, no. 3, p. 2, 2006.
- [15] L. Sekanina, "Evolutionary design of gate-level polymorphic digital circuits," in *Applications of Evolutionary Computing*, ser. LNCS, vol. 3449. Lausanne, Switzerland: Springer Verlag, 2005, pp. 185–194.
- [16] L. Starecek, L. Sekanina, and Z. Kotasek, "Reduction of test vectors volume by means of gate-level reconfiguration," in *Proc. of 2008 IEEE Design and Diagnostics of Electronic Circuits and Systems Workshop*. IEEE Computer Society, 2008, pp. 255–258.
- [17] R. Ruzicka, L. Sekanina, and R. Prokop, "Physical demonstration of polymorphic self-checking circuits," in *Proc. of 14th IEEE International On-Line Testing Symposium*. IEEE, 2008, pp. 31–36.
- [18] "FITkit – A Board with FPGA and MCU developed at Faculty of Information Technology, Brno University of Technology," 2008, URL: <http://www.fit.vutbr.cz/fitkit>.
- [19] R. Ruzicka and L. Sekanina, "Evolutionary circuit design in repomo - reconfigurable polymorphic module," in *Proceedings of the Second IASTED International Conference on Computational Intelligence*. ACTA Press, 2006, pp. 237–241.
- [20] Z. Vasicek, L. Capka, and L. Sekanina, "Analysis of reconfiguration options for a reconfigurable polymorphic circuit," in *Proc. of the 2008 NASA/ESA Conference on Adaptive Hardware and Systems*. IEEE Computer Society, 2008, pp. 3–10.
- [21] L. Sekanina, T. Martinek, and Z. Gajda, "Extrinsic and intrinsic evolution of multifunctional combinational modules," in *2006 IEEE World Congress on Computational Intelligence*. IEEE Computational Intelligence Society, 2006, pp. 9676–9683.