ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL, 2013, Vol. 42, No. 1

Implementing a Unique Chip ID on a Reconfigurable Polymorphic Circuit

Lukas Sekanina, Richard Ruzicka, Zdenek Vasicek, Vaclav Simek, Petr Hanacek

Brno University of Technology, Faculty of Information Technology,
IT4Innovations Centre of Excellence
Bozetechova 2, 612 66 Brno, Czech Republic
e-mail: sekanina@fit.vutbr.cz

crossref http://dx.doi.org/10.5755/j01.itc.42.1.925

Abstract. The need for secure physical implementations of cryptography functions has become urgent in the recent years. In particular, a unique unclonable chip ID has been implemented using various techniques. In this paper, we investigate the use of polymorphic gates as a new mechanism for implementing a unique chip ID in systems already containing some polymorphic gates. The proposed solution exploits the fact that switching time of polymorphic gates (controlled by V_{dd}) is slightly different even for neighboring gates on the same die because of fabrication variations. We applied a partial reconfiguration in order to generate 48-bit IDs on the reconfigurable polymorphic REPOMO32 chip that we have developed in our previous research. In some application scenarios, we achieved 94.44% stable bits which is reasonably close to existing approaches.

Keywords: Unclonable ID; Polymorphic Gate; Process Variation; Reconfigurable Circuit; Security.

1. Introduction

The requirement for security has become as much important as the functionality, cost or maintainability even for ordinary applications and systems. In this context, the security means protecting data and HW/SW systems from unauthorized access, use, modification or destruction. Classical cryptography which is based on computationally hard to break algorithms provides various solutions against some of these problems. Physical attacks such as side-channel attacks on cryptographic devices have shown that even computationally hard to break functions are not sufficient in protecting secure information when their physical implementation does not take into account the security issue. Hence we can observe an increased importance of physical security and mechanisms ensuring physical unclonable functions and data [26].

Nowadays, it is often required for some embedded systems to provide a unique unclonable identification number (ID). Typical applications where the ID could be utilized include integrated circuit (IC) identification and authentication, encryption of communication channels and intellectual property (IP) protection in case of field programmable gate arrays (FPGAs).

On-chip IDs are usually implemented either by writing a unique number to a non-volatile memory or post-fabrication modification of a chip or introducing circuits which derive unique information from inherent fabrication process variations. The last approach has attracted a considerable attention in the recent years mainly because it represents a reasonably secure and cheap solution (see Sect. 2).

In our previous work we have developed and fabricated a small reconfigurable polymorphic chip called REPOMO32 (REconfigurable POlymorphic MOdule) [17]. The REPOMO32 chip can be programmed similarly to other digital programmable chips, by means of a configuration bit stream which defines the functionality of REPOMO32's elementary reconfigurable blocks. In addition, however, REPOMO32 contains polymorphic gates – new electronic components whose logic function can be switched unconventionally as response to various means including power supply voltage (in our case). The main feature which distinguishes polymorphic logic from ordinary programmable logic is that a polymorphic gate does not contain any switch or multiplexer to select one of k different functions. The multifunctionality is inherently embedded in the structure of each polymorphic gate. This concept has been introduced by Stoica et al. [22] and later explored by others, e.g. [6, 18].

In the REPOMO32 chip the logic function of polymorphic gates depends on level of the power supply voltage (V_{dd}). When V_{dd} = 3.0 - 3.8V the NAND/NOR gates used in REPOMO32 exhibit the NOR function and when V_{dd} = 3.9 - 5V the gates exhibit the NAND function. Hence the behavior of

REPOMO32 is not defined by a configuration bit stream solely. It also depends on the level of power supply voltage.

If an electronic system contains a reconfigurable polymorphic device (such as REPOMO32) it may be programmed to perform various functions that combine standard logic functionality with capabilities of sensing in *one* compact structure. Such systems usually contain electronic circuits enabling controlled changes of V_{dd} that are required in particular applications. Various applications have been proposed utilizing this concept. For a survey, see Section 3.

In this article, we present a new application of polymorphic electronics. We investigate the use of polymorphic gates as a new mechanism for implementing an unclonable ID in systems already containing some polymorphic gates. The proposed solution exploits the principle that the switching time of polymorphic gates (controlled by V_{dd}) is slightly different even for neighboring gates on the same die because of fabrication variations. We propose a simple circuit which compares the output signals of a pair of polymorphic gates and determines which of them has switched its logic function earlier as a response to the V_{dd} change. A single bit of ID is then generated. More bits can be generated using additional polymorphic gates and comparators.

Therefore, in addition to their original function, various polymorphic systems could be equipped with a new functionality (i.e. unclonable ID) almost for free. The experimental evaluation of the proposed method has been carried out using 21 (universal) REPOMO32 chips, i.e. we have not designed a new chip to perform the experiments. The results show that the proposed scheme is able to produce unique and reasonably non-colliding IDs.

The rest of the paper is organized as follows. Section 2 briefly introduces the area of on-chip IDs. The REPOMO32 chip, which will be used for experimental evaluation, is described in Section 3. Section 4 deals with the proposed implementation of on-chip ID using the REPOMO32 chip. The results, which include the ID collision analysis and ID stability analysis, are summarized in Section 5. Conclusions are given in Section 6.

2. Unique On-Chip ID

Obtaining a unique ID from a chip is a problem very similar to the unique cryptographic key generation problem. We need a number which is unique to a particular chip and does not change during time. The first implementations of the on-chip IDs were based on writing a unique number to a non-volatile memory or post-fabrication modification of the chip using lasers or fuses. In 2000, Lofstrom et al. proposed a new method utilizing inherent process variations that exist from die to die for obtaining a unique ID [8]. Unlike post-fabrication creating of IDs, the ID codes are assigned randomly here. Recently published papers have shown that a unique ID can be extracted from each chip by comparing variations in transistors characteristics [23], digital path delays [10, 24] or random initialization states of static RAM cells [4], i.e. by utilizing the features that are undesirable in common digital circuits [2]. However, traditional delay-based solutions exploit only IC's process variations for ID generation.

More advanced approaches have utilized equivalent resistance variations in the power distribution system [5] and variations of environment (temperature, power supply noise and crosstalk) [27]. A detailed survey and classification can be found in [12].

As physical variations creating the ID are below the resolution of the manufacturing equipment it is difficult for an attacker to make a reasonably good clone of the chip. Unfortunately, the output values can not directly be used as a unique ID because approximately 4-5% bits are unstable [28]. Hence any implementation must ensure that each particular chip is identified with a very high probability. In particular, determining a sufficient number of bits for each ID and ensuring high stability in an operational environment (noise, unstable V_{dd}) are crucial for obtaining highly stable ID codes. An analysis of the probability of misidentification has been performed for different unstable bit percentages in [23].

2.1. ID Collisions

There is a finite probability of the ID code collision within a given number of chips. Following the analysis given in [23], ID collision between any two chips can be expressed as $1/2^X$ where X is the number of bits in an ID code. The total probability of ID collision across Y chips can be expressed as

$$P_c = 1 - \prod_{n=1}^{Y} \left(1 - \frac{n-1}{2^X}\right) \tag{1}$$

Figure 1 shows the ID collision probability versus different number of chips Y for various ID lengths. It can be seen that the use of randomly assigned IDs might be highly reliable for many applications. The collision probability is less than 10^{-9} even for short IDs (64 bits) when 10 thousand chips are fabricated.

2.2. Requirements on IDs

It is a well known fact that subsequent readings of the ID of the same chip may lead to different codes. The Hamming distance measured on the ID codes reflects the number of unstable bits. By using of error correcting codes it is possible (under some assumptions such that the code length is sufficient for a given implementation) to positively identify the chip even if some bits of ID are unstable.

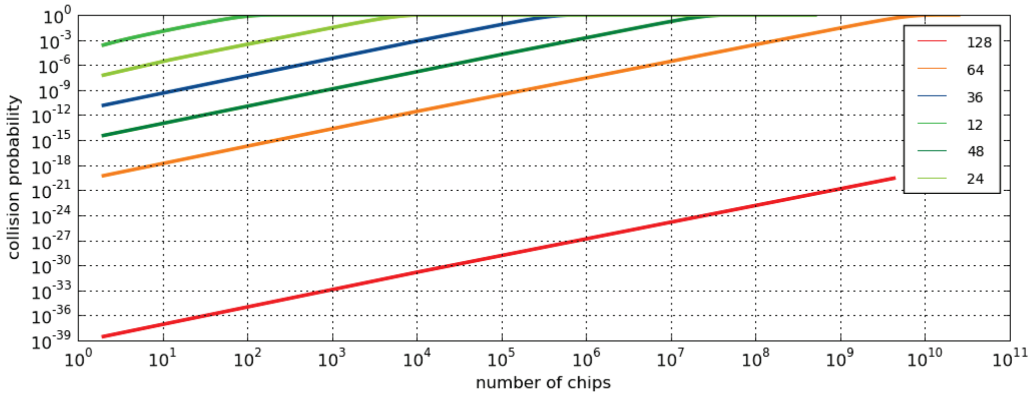

Figure 1. Probability of ID collision versus different number of chips for various ID lengths

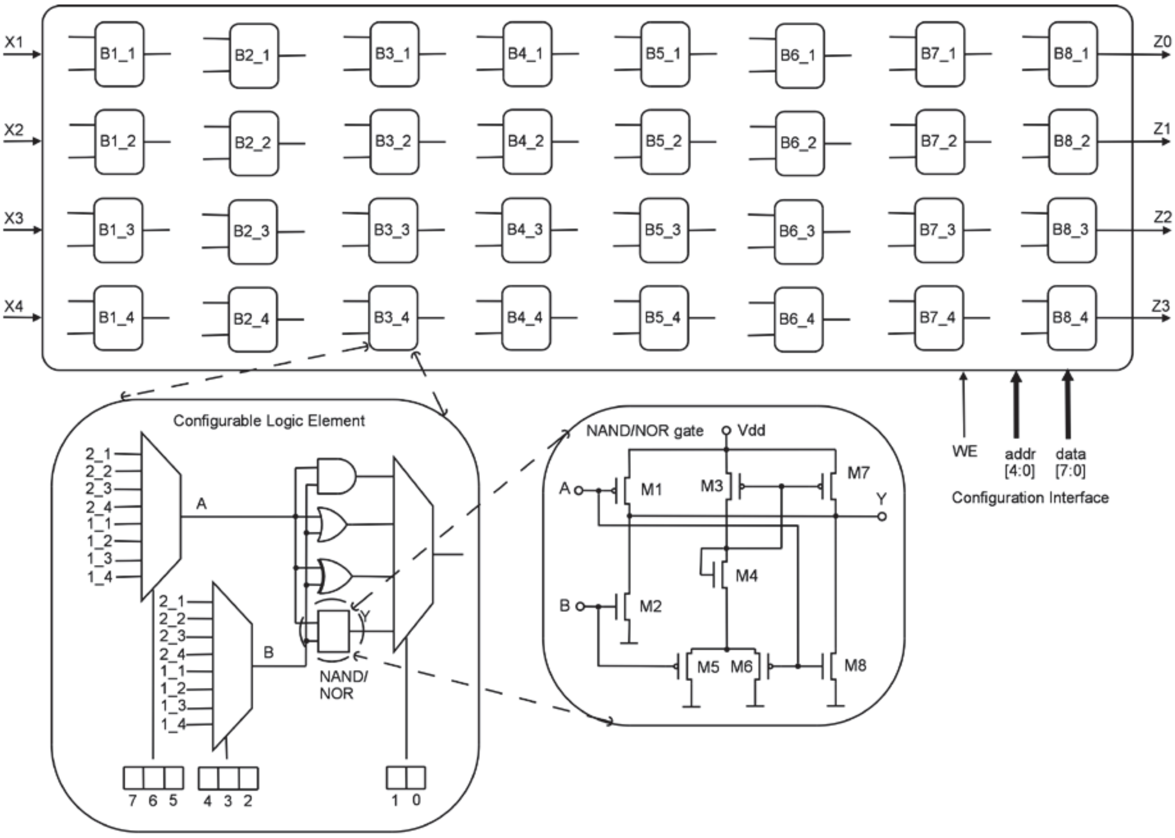

Figure 2. The REPOMO32 chip shown together with the implementation details of a single CLE block and NAND/NOR gate

The Hamming distance between the ID codes obtained from two different chips should be, on average, half of the total number of bits in the ID. The average Hamming distance significantly below $X/2$ indicates that there are many collisions and the underlying variations of physical process are not sufficient. In order to make a potential attack even more difficult the number of zeros has to be as close as possible to the number of ones in every ID.

Performing of the analysis of the average Hamming distance, determining of the number of unstable bits and measuring of the zero-one distribution is important for evaluation of any proposal in this area.

3. Reconfigurable Polymorphic Chip

Since 2001 several papers have been published dealing with implementations of polymorphic gates. Initial experiments, carried out at NASA JPL, were performed with polymorphic circuits embedded into a filed programmable transistor array [22] and later with polymorphic gates directly implemented in HP 0.5 µm CMOS technology [21]. The REPOMO32 chip, which is presented in this paper, employs polymorphic gates originally proposed in 2008 [15].

The main motivation for development of polymorphic electronics is potentially very fast and cheap reconfiguration of polymorphic gates, e.g. by means of V_{dd}. Current applications of polymorphic electronics include multifunctional circuits (counters [13] and filters [16]), test vector reduction [18] and self-testing circuits [11, 29]. Since conventional tools do not support synthesis of polymorphic circuits, various design and optimization methods have been proposed in this domain [3, 9, 19]. They typically combine classic approaches (such as Binary Decision Diagrams or satisfiability problem solving) with evolutionary computing. Theoretical research in this area has lead to a new completeness theory for polymorphic Boolean networks [7].

A recent work that falls into this context has shown a novel graphene reconfigurable logic device based on the control of p-n doping configurations using split gates [25]. By using split gates to change the grapheme properties, multi-function logic gate was obtained and dynamically reconfigured. This kind of devices is considered as a new way for implementing some reconfigurable electronic systems. We can speculate that, for example, Tabula's 3D Spacetime FPGAs [1] could benefit from it in the future.

In order to demonstrate crucial features of elementary polymorphic circuits (such as small polymorphic combinational circuits), the REPOMO32 chip has been developed and tested [17].

3.1. REPOMO32

REPOMO32 is primarily intended for implementation of polymorphic four-input/four-output combinational circuits. As Fig. 2 shows, the chip consists of 32 two-input Configurable Logic Elements (CLEs) organized in an array of 4 rows and 8 columns. A CLE can be programmed to perform one of the following functions: AND, OR, XOR and polymorphic NAND/NOR (controlled by V_{dd}). When V_{dd} = 3.0 - 3.8 V the NAND/NOR gate exhibits the NOR function and when V_{dd} = 3.9 - 5V the gate exhibits the NAND function. Remaining gates do not change their logic functions with changes of V_{dd} within the range of 3 - 5 V.

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. In total, 8 bits define the configuration 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 (*addr)* and configuration data (*data)* followed by activating the *WE* signal. The chip can completely be reconfigured in 32 configuration steps. The primary outputs *Z0* … *Z3* 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 × 1970 μm. It was fabricated using AMIS CMOS 0.7 μm technology.

The REPOMO32 chip is considered as a small module which may be embedded into a larger system. Note that REPOMO32 is not intended for implementing an unclonable ID. However, we will show that it can be programmed to obtain a basic unclonable ID.

3.2. REPOMOKit

A REPOMO32/kit is a dedicated experimental board which has been designed with the aim to conveniently facilitate test, evaluation and measurement procedures of all the features available inside the REPOMO32 chip [20]. This evaluation platform contains a DIL-28 socket for an integrated circuit with polymorphic gates (REPOMO32 chip), which may be configured by means of external switches or signals. A set of fast buffers is deployed in order to separate available input ports from the surrounding environment. The REPOMOKit also contains a Xilinx XC9572XL CPLD which may be used as a configuration controller for REPOMO32 or to customize an additional application-specific logic.

A special attention has been paid to the power generation and distribution across the board. Flexible power system is virtually divided into several independent branches, which deliver power supply within a range of 3.3–5V to the individual components onboard. It consists of dedicated rails for on-board auxiliary logic and a set of programmable power sources for the REPOMO32 chip itself. For example, the user may choose between stabilized 3.3/5V voltage in either manual way or by means of using a signal from external controller or logic. Another feasible alternative how to obtain V_{dd} is based on using programmable power supply, which makes it possible to generate an eligible level of the output voltage within the range of 3.3-5V. A digital potentiometer and integrated microcontroller are employed for this task. The REPOMO32/kit contains a dedicated heated chamber which is used to create special kind of working environment, where the temperature may reach up to 150 °C. The whole conception is based around metal box with dimensions of 76 mm × 48 mm × 19 mm (length × width × height). Spiral heating wire is placed inside this tightly enclosed space where it is responsible for heat generation task. Necessary energy for the process of thermal management is delivered by an external source (12 V with maximum current of 1.5 A). An adequate shielding of this heated chamber ensures that the eventual temperature effects on functionality of other components on REPOMO32 kit are minimized to high degree. The parameters of a concrete thermal analysis test are fully configurable (e.g. duration of heating phase, specific temperature, regulation hysteresis). The whole procedure is subsequently managed by an 8-bit MCU, in particular MC9S08QE8 manufactured by Freescale. Its tasks include processing of test parameters, temperature measurement with sensor ADT7301 and heating regulation by means of switching power to the spiral heating wire. Temperature measurement can be carried out with a resolution of 0.03125 °C, which is more than sufficient for the character of REPOMO32 chip thermal analysis.

3.3. Measurement on Polymorphic Gates

Dynamic characteristics of polymorphic gates of REPOMO32 chip were investigated for normal temperature in [20]. Fig. 3 shows the output of the NAND/NOR gate when the chip is exposed to supply voltage ramp up from 3.3 V to 5 V level. In order to observe a change on the output logic signal, the NAND/NOR gate has different logic values assigned at its input ports (i.e. the 01 or 10 combination).

Fig. 4 shows a delay of one particular critical path created using four CLEs (one of them configured as the NAND/NOR) in the REPOMO32 chip as a function of temperature. It can be seen that the length of critical path t_{pd} is almost constant to approx. 100°C. After this limit, t_{pd} rises significantly. It means that the chip goes slower for high temperatures. Note that the experiment was performed for V_{dd} = 3.3 V on four randomly selected chips. The results of experiments show that there are no significant differences among chips for traditional temperature range [14].

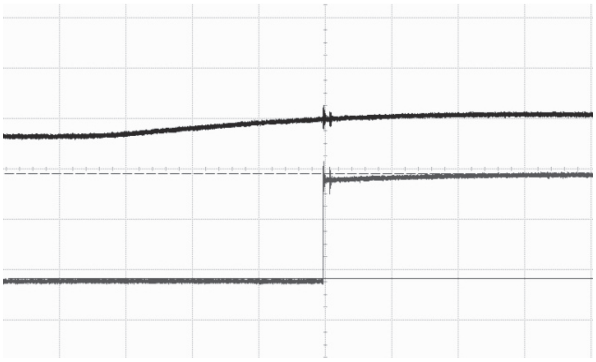

Figure 3. Response of a polymorphic NAND/NOR gate (down trace) exposed with supply voltage ramp up from the 3.3V to 5V level (upper trace). The inputs of the NAND/NOR gate are set at '0' and '1'. Time base is 1 μs/div

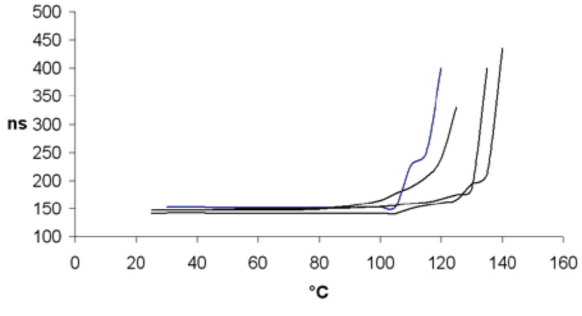

Figure 4. Critical path length vs. temperature for four selected chips

4. Unclonable ID on REPOMO32

Implementing of unclonable ID on the REPOMO32 chip is a challenging task since REPOMO32 has been designed for completely different purposes. The proposed solution is based on the fact that the switching time of polymorphic gates (controlled by V_{dd}) is slightly different even for neighboring gates on the same die because of fabrication variations. There are several constraints that have to be considered when polymorphic gates of the REPOMO32 chip are utilized for unique ID generation.

- Since there are only four outputs it is possible to produce only a few bits of a particular ID in one step.
- Since there are not direct connections from primary inputs to CLEs that are situated in columns 2-8 it is necessary to configure some CLEs as wires (i.e. identity function) to make connections from the primary inputs to particular CLEs.
- Since no data registers exist in the REPOMO32 chip the resulting ID must be stored in an external register.

The proposed solution overcomes these problems by a partial reconfiguration of the REPOMO32 chip. The resulting ID is then generated in several steps. As only four primary outputs are available, we use only four polymorphic gates (one column of CLEs) to generate a part of the ID in one step. Fig. 5b shows that the outputs of four polymorphic gates which are configured as NAND/NORs are connected to a set of S-R (set reset) latches (Fig. 5c). The S-R latches may be implemented either on an external chip (on the CPLD in our case) or as a part of our future polymorphic reconfigurable device. An S-R latch determines which of two polymorphic gates has switched its logic function earlier as a response to the V_{dd} change (3V → 5V.). The resulting bit is then considered as one bit of the ID. It is necessary to ensure that the inputs of the NAND/NOR gates are fed by mutually inverse logic values to invert the output value of the gate with the change of V_{dd}. These values come from the primary inputs via CLEs (situated in columns 1-7) that must be configured to operate as wires (e.g. using the A AND $A = A$ setting).

Therefore, the unique bits are obtained when V_{dd} is changed and under assumption that physical characteristics (delays during switching) of pairs of gates differ from die to die.

Fig. 5 shows that a 6-bit ID can be produced using our four-output REPOMO32 chip. In general, the number of bits is $n(n\text{-}1)/2$, where n is the number of available (outputs of) polymorphic gates that are implemented in a given chip. In order to obtain longer IDs we sequentially reconfigure CLEs and read the outputs of polymorphic gates in columns 1–8. In this scenario, a single column of REPOMO32 is configured as four NAND/NORs and the

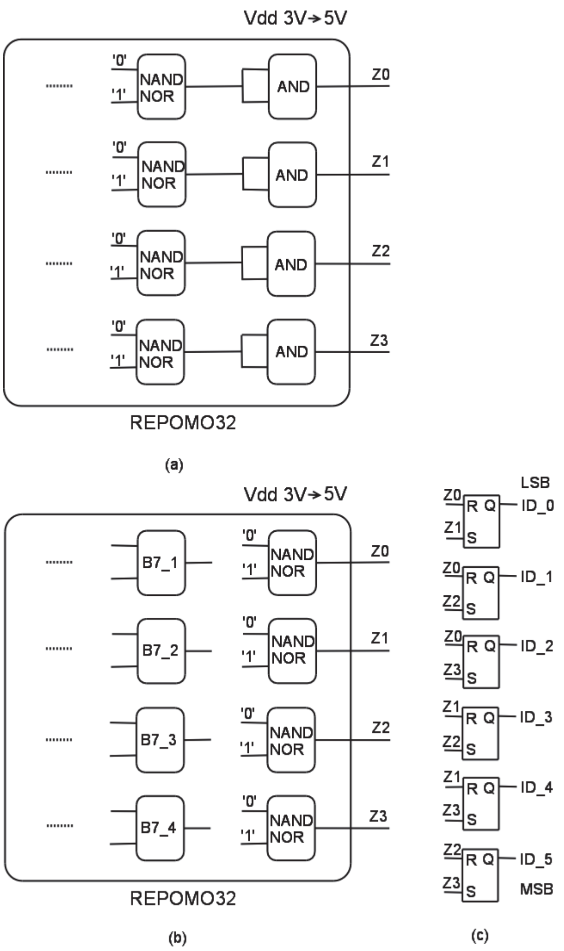

Figure 5. The on-chip ID on REPOMO32: (a) The NAND/NOR gates configured to column 7; (b) The NAND/NOR gates configured to column 8; (c) The S-R circuits used to generate and store the ID

remaining columns are configured as “wires”. The S-R latches are reused. This strategy is illustrated for columns 7 and 8 in Fig. 5a, b. The initial configuration requires 32 configuration steps (all CLEs have to be configured). A 6-bit ID is extracted using the first column of CLEs (configured as the NAND/NOR function) while columns 2–8 operate as “wires”. Then, in order to obtain next 6 unique bits, 8 CLEs (located in columns 1 and 2) have to be reconfigured (the first column is now the “wire”; the second column performs the NAND/NOR functions). The same strategy is repeated for remaining columns. In total, eight 6-bit IDs can now be produced using a single REPOMO32 chip.

5. Measured Results and Discussion

In order to evaluate the proposed method, IDs were measured for 21 REPOMO32 chips available to us. Table 1 gives the 6-bit IDs obtained for all columns of all the chips. We repeated the measurement 1000 times for each chip to establish the number of unstable bits. The resulting average number of unstable bits varies from 5.56% (column 8) to 17.94% (column 2) as shown in the last section of Table 1. We hypothesize that lower values measured for columns 5-8 are due a closer connection of polymorphic gates to the S-R circuits; i.e. delays which may introduce errors are more stable. The average number of unstable bits is 11.08 % when the whole 48-bit ID is considered.

Table 1. The 6-bit ID obtained from each column for 21 REPOMO32 chips (A-U)

chip/column	1	2	3	4	5	6	7	8
A	35	0F	25	01	07	3F	27	1E
B	3F	27	25	35	30	1A	21	3F
C	27	27	38	37	38	3F	3C	37
D	0B	27	21	25	38	27	08	1F
E	27	34	35	27	0A	1E	25	3E
F	0A	3E	07	07	18	03	3E	20
G	21	20	21	07	3C	27	1A	18
H	1F	38	35	3F	34	0F	25	0A
I	34	21	1F	0B	27	18	30	0B
J	0B	00	25	03	34	30	0B	1F
K	1E	37	27	00	35	3C	20	1A
L	3E	3F	25	1F	03	01	1F	01
M	27	34	3E	27	37	08	35	0B
N	27	03	08	0F	00	18	3F	37
O	00	37	18	21	3C	35	21	1E
P	35	3F	0F	03	0F	3F	1A	3F
Q	20	3F	3F	21	21	37	0B	01
R	3F	3E	34	21	21	0B	07	07
S	0F	1F	21	3F	34	0A	20	21
T	3F	07	20	21	35	30	3C	07
U	0B	1F	25	1F	1F	25	03	01
unstable bits [%]	12.68	17.94	11.64	15.33	7.72	9.41	8.40	5.56

Table 2. Measured Hamming distance of the ID codes for all combinations of 21chips (A-U)

chip	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
A	0	19	25	24	18	30	24	23	23	22	18	23	26	23	22	14	21	18	28	20	22
B	19	0	16	15	15	27	27	18	24	19	17	24	21	20	23	25	26	17	15	17	27
C	25	16	0	15	17	25	21	24	28	27	27	28	21	16	19	21	24	23	21	17	29
D	24	15	15	0	22	22	14	23	33	16	20	27	28	27	16	24	21	22	20	18	20
E	18	15	17	22	0	24	24	17	25	24	24	27	16	21	22	22	27	20	24	28	28
F	30	27	25	22	24	0	20	23	31	26	26	17	26	25	28	24	23	24	20	28	20
G	24	27	21	14	24	20	0	23	27	16	26	27	26	27	22	22	23	30	26	26	22
H	23	18	24	23	17	23	23	0	24	23	21	22	17	30	25	31	28	17	15	27	21
I	23	24	28	33	25	31	27	24	0	25	19	24	15	22	25	19	24	27	27	23	31
J	22	19	27	16	24	26	16	23	25	0	20	27	24	21	22	26	25	24	24	18	20
K	18	17	27	20	24	26	26	21	19	20	0	27	22	31	18	22	23	24	22	18	24
L	23	24	28	27	27	17	27	22	24	27	27	0	23	22	35	23	20	17	21	23	13
M	26	21	21	28	16	26	26	17	15	24	22	23	0	23	24	26	25	18	22	20	26
N	23	20	16	27	21	25	27	30	22	21	31	22	23	0	27	23	30	25	23	19	29
O	22	23	19	16	22	28	22	25	25	22	18	35	24	27	0	24	19	26	28	22	26
P	14	25	21	24	22	24	22	31	19	26	22	23	26	23	24	0	19	24	30	26	24
Q	21	26	24	21	27	23	23	28	24	25	23	20	25	30	19	19	0	17	27	25	21
R	18	17	23	22	20	24	30	17	27	24	24	17	18	25	26	24	17	0	22	18	24
S	28	15	21	20	24	20	26	15	27	24	22	21	22	23	28	30	27	22	0	20	16
T	20	17	17	18	28	28	26	27	23	18	18	23	20	19	22	26	25	18	20	0	26
U	22	27	29	20	28	20	22	21	31	20	24	13	26	29	26	24	21	24	16	26	0

Table 2 shows the Hamming distances obtained for 48-bit IDs. The average Hamming distance is 22.83±4.11 (see the histogram in Fig. 6) which is a value reasonably close to the theoretical value 24.

Figure 7 shows the average Hamming distance for IDs containing 6, 7…24 bits that were obtained from columns 1-4 and 5-8. It can be seen that more reliable IDs were obtained using columns 5-8. In order to obtain collision free IDs, 10 bits are sufficient for our set of 21 chips. The ratio of the number of “1” vs the number of “0” is 1.25 in the 48-bit IDs.

The proposed implementation was significantly influenced by the fact that REPOMO32 provides four output pins only. Hence only six-bit IDs were extracted during one change of the power supply voltage and thus only the 48-bit ID was obtained in

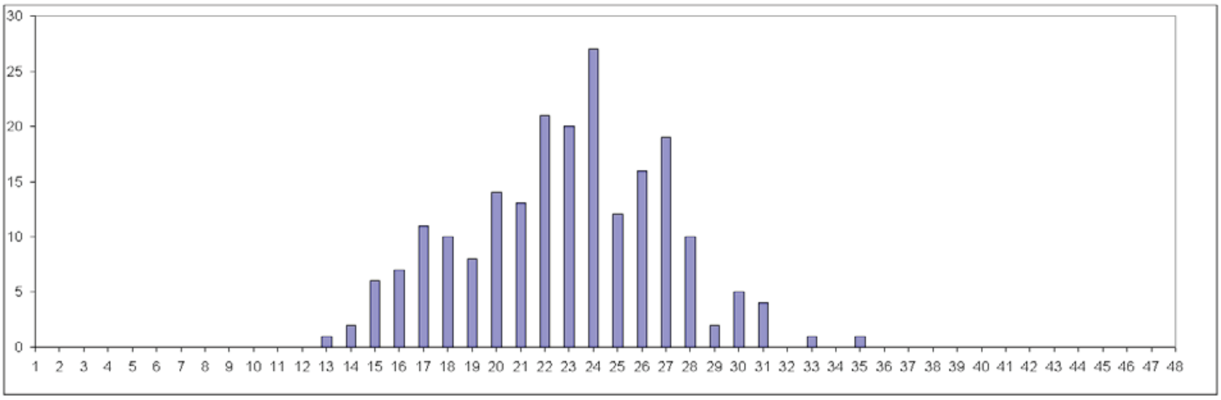

Figure 6. Histogram of Hamming distances for 48-bit IDs

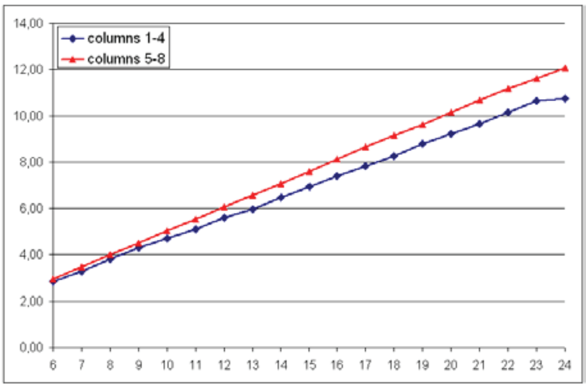

Figure 7. The average Hamming distance of ID codes vs the number of bits in IDs for columns 1-4 and 5-8

total. We observed different quality of IDs read from different columns of the chip. The quality of generated IDs may be increased by introducing various error correction codes as proposed e.g. in [28].

A straightforward approach to implementing a *k*-bit signature in a future polymorphic ASIC is the utilization of a sufficient number of independent polymorphic gates and their direct connection with the S-R latches.

In comparison with the implementation proposed in this article, that approach would lead to a much faster ID generation (no reconfiguration will be needed) and increasing the quality of IDs as the number of unstable bits would be around 5.56% (see column 8 in Table 1).

6. Conclusions and Future Work

In this paper, we showed that polymorphic electronics can be utilized to implement unique on-chip IDs. We applied a partial reconfiguration in order to generate 48-bit IDs on the REPOMO32 chip. If IDs were obtained from polymorphic gates connected directly to the S-R circuits (i.e. from column 8 of REPOMO32) only 5.56% bits are unstable which is a result reasonably close to existing approaches achieving 4% on ASIC [23] and FPGA [4].

Future work will consists of replacing our reconfigurable chip by a new chip specialized for this application where a sufficient number of polymorphic gates will be directly connected to the S-R latches. We will also introduce a suitable error correcting scheme.

Acknowledgements

This work was partially supported by the Czech Science Foundation under project *Natural Computing on Unconventional Platforms* P103/10/1517, the research program *Security-Oriented Research in Information Technology* MSM 0021630528, the BUT project FIT-S-11-1 and the *IT4Innovations Centre of Excellence* CZ.1.05/1.1.00/02.0070.

References

[1] Abax Product Family Overview. URL: www.tabula.com, 2012

[2] **E. Bareiša, V. Jusas, K. Motiejūnas, R. Šeinauskas.** Delay Fault Models and Metrics. In: *Information Technology And Control*, 2005, Vol. 34, No. 4, pp. 307-317.

[3] **Z. Gajda, L. Sekanina.** On evolutionary synthesis of compact polymorphic combinational circuits. In: *Journal of Multiple-Valued Logic and Soft Computing*, 2011, Vol. 17, No. 6, pp. 607-631.

[4] **J. Guajardo, S. S. Kumar, G. J. Schrijen, P. Tuyls.** FPGA intrinsic PUFs and their use for IP protection. In: *Cryptographic Hardware and Embedded Systems Workshop*, LNCS 4727, Springer, 2007, pp. 63-80.

[5] **R. Helinski, D. Acharyya, J. Plusquellic.** A physical unclonable function defined using power distribution system equivalent resistance variations. In: *Proc. of Design Automation Conference*, ACM, 2009, pp. 676-681.

[6] **D. Hentrich, E. Oruklu, J. Saniie.** Polymorphic Computing: Definition, Trends, and a New Agent-Based Architecture. In: *Circuits and Systems*, 2011, Vol. 2, No. 4, pp. 358-364.

[7] **Z. Li, W. Luo, L Yue, X Wang.** On the completeness of the polymorphic gate set. In: *ACM Transactions on Design Automation of Electronic Systems*, 2010, Vol. 15, No. 4, pp. 1-20.

[8] **K. Lofstrom, W. R. Daasch, D. Taylor.** IC identification circuit using device mismatch. In: *Proc. of the IEEE International Solid State Circuits Conference, IEEE*, 2000, pp. 372–373.

[9] **W. Luo, Z. Zhang, X. Wang.** Designing polymorphic circuits with polymorphic gates: a general design approach. In: *IET Circuits, Devices & Systems,* 2007, Vol. 1, No. 6, pp. 470-476.

[10] **A. Maiti, N. Raghunandan, A. Reddy, P. Schaumont.** Physical unclonable function and true random number generator: A compact and scalable implementation. In: *Proceedings of the 19th Great Lakes Symposium on VLSI*. ACM, 2009, pp. 425-428.

[11] **M. Mashayekhi, H. H. Ardakani, A. Omidian.** A new efficient scalable bist full adder using polymorphic gates. In: *World Academy of Science, Engineering and Technology Journal*, 2010, Vol. 61, pp. 283-286.

[12] **U. Rührmairy, S. Devadas, F. Koushanfar.** Security based on Physical Unclonability and Disorder. In: *Introduction to Hardware Security and Trust*, M. Tehranipoor and C. Wang (ed.), Springer, 2012, pp. 65-102.

[13] **R. Ruzicka.** Gracefully degrading circuit controllers based on polytronics. In: *Proc. of 13th Euromicro Conference on Digital System Design*. IEEE Computer Society, 2010, pp. 809-812.

[14] **R. Ruzicka, V. Simek, L. Sekanina.** Behavior of CMOS Polymorphic Circuits in High Temperature Environment. In: *Proceedings of the 2011 IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems. IEEE CS*, 2011, pp. 447-452.

[15] **R. Ruzicka, L. Sekanina, R. Prokop.** Physical demonstration of polymorphic self-checking circuits. In: *Proc. of the 14th IEEE Int. On-Line Testing Symposium.* IEEE Computer Society, 2008, pp. 31-36.

[16] **L. Sekanina, R. Ruzicka, Z. Gajda.** Polymorphic FIR filters with backup mode enabling power savings. In: *2009 NASA/ESA Conference on Adaptive Hardware and Systems*, *IEEE*, 2009, pp. 43-50.

[17] **L. Sekanina, R. Ruzicka, Z. Vasicek, R. Prokop, L. Fujcik.** Repomo32 – New reconfigurable polymorphic integrated circuit for adaptive hardware. In: *2009 IEEE Workshop on Evolvable and Adaptive Hardware*, *IEEE CIS*, 2009, pp. 39-46.

[18] **L. Sekanina, L. Starecek, Z. Kotasek, Z. Gajda.** Polymorphic gates in design and test of digital circuits. In: *Int. Journal of Unconventional Computing*, 2008, Vol. 4, No. 2, pp. 125-142.

[19] **L. Sekanina, Z. Vasicek.** A SAT-based Fitness Function for Evolutionary Optimization of Polymorphic Circuits, In: Proc. of the *2012 Design, Automation and Test in Europe*, Dresden, EDAA, 2012, pp. 715-720.

[20] **V. Simek, R. Ruzicka, L. Sekanina.** On analysis of fabricated polymorphic circuits. In: *Proc. of the 13th Int. IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems*, IEEE, 2010, pp. 281-284.

[21] **A. Stoica, R. Zebulum, X. Guo, D. Keymeulen, I. Ferguson, V. Duong.** Taking Evolutionary Circuit Design From Experimentation to Implementation: Some Useful Techniques and a Silicon Demonstration. In: *IEE Proc.-Comp. Digit. Tech*., 2004, Vol. 151, No. 4, pp. 295-300.

[22] **A. Stoica, R. S. Zebulum, D. Keymeulen.** Polymorphic electronics. In: *Proc. of Evolvable Systems: From Biology to Hardware Conference*. LNCS 2210, 2001, pp. 291-302.

[23] **Y. Su, J. Holleman, B. P. Otis.** A digital 1.6 pj/bit chip identification circuit using process variations. In: *IEEE Journal of Solid-State Circuits*, 2008, Vol. 43, No. 1, pp. 69-77.

[24] **G. E. Suh, S. Devadas**. Physical unclonable functions for device authentication and secret key generation. In: *Proc. of Design Automation Conference.* ACM, 2007, pp. 9-14.

[25] **S. Tanachutiwat, J. U. Lee, W. Wang, C. Y. Sung.** Reconfigurable multi-function logic based on graphene p-n junctions. In: *Design Automation Conference*. ACM, 2010, pp. 883-888.

[26] **M. Tehranipoor, C. Wang (Eds.).** *Introduction to Hardware Security and Trust*. Springer Verlag, Berlin, 2012, pp. 195-229.

[27] **X. Wang, M. Tehranipoor.** Novel physical unclonable function with process and environmental variations. In: *Proc. of Design, Automation and Test in Europe*, 2010, pp. 1065-1070.

[28] **H. Yu, P. H. W. Leong, H. Hinkelmann, L. Möller, M. Glesner, P. Zipf.** Towards a unique fpga-based identification circuit using process variations. In: *19th International Conference on Field Programmable Logic and Applications*. IEEE, 2009, pp. 397-402.

[29] **R. S. Zebulum, A. Stoica**. Multifunctional Logic Gates for Built-In Self-Testing. *NASA Tech Briefs*. Vol. 30, No. 3, 2006, p. 10.

[30] **D. P. Agrawal, Q. A. Zeng.** Introduction to Wireless and Mobile Systems, *Thomson Brooks*, 2003, ISBN 0534-40851-6.

Received June 2011.