A CMOS FPTA Chip for intrinsic Hardware Evolution of analog electronic Circuits

Jörg Langeheine, Joachim Becker, Simon Fölling, Karlheinz Meier, Johannes Schemmel
Heidelberg University,
Kirchhoff-Institute for Physics,
Schröderstr. 90,
D-69120 Heidelberg, Germany.
langehei@kip.uni-heidelberg.de
http://www.kip.uni-heidelberg.de/vision/projects/evo_tarray.html

Abstract

This paper describes and discusses an intrinsic approach to hardware evolution of analog electronic circuits using a Field Programmable Transistor Array (FPTA). The FPTA is fabricated in a $0.6\ \mu\mathrm{m}$ CMOS process and consists of 16×16 transistor cells. The chip allows to configure the gate geometry as well as the connectivity of each of the 256 transistors. Evolutionary algorithms are to be run on a commercial PC to produce the new circuit configurations that are downloaded to the chip via a PCI card. In contrast to extrinsic hardware evolution all environmental conditions present on the device under test have to be taken into account by the evolutionary algorithm. Thus a selection pressure is raised towards solutions that actually work on real dice.

1. Introduction

Automating the process of analog circuit design has long been an intriguing idea, but the difficulties that have to be overcome are manifold. On the other hand the technique of hardware evolution has recently been applied to solve some engineering design problems with promising results (see for example [1], [2], [3]).

While extrinsic hardware evolution (that is using simulation models for the hardware) is more versatile and may be easier to implement, intrinsic evolution can yield higher rates of tested individuals and includes all sorts of variations and noise present in real chips (for a more detailed argumentation the reader is referred to [4]). Since evolutionary algorithms may use any given detail of the material provided, these variations seem to be essential to obtain circuits that actually work on real dice and are robust against environmental variations (cf. [2], [5], [6]). Another or additional way to deal with varying environments may be to fine tune the behavior of the employed circuitry during operation or at least in the field, excluding the use of software models.

The project described in this paper is a first approach to design automation of transistor level circuits. The long term goal of design automation comes in two flavors: First, hardware evolution of analog circuits could be utilized as a design aid yielding design ideas and principles to a posed problem that can be used as a building block. Second, the evolved hardware itself can be used directly for the posed problem, allowing for example to fine tune the circuit according to the actual environmental conditions. While the feasibility of the former goal depends on the portability of the evolved solutions, the latter one will probably result in circuits that are slower and less effective in terms of used area and power than their counterparts directly implemented in silicon. The realistic short term goal thus is to first investigate the principal feasibility of intrinsic hardware evolution of transistor level analog electronic circuits.

According to the general purpose character of the proposed approach the hardware platform chosen is a CMOS Field Programmable Transistor Array (FPTA) consisting of 16×16 transistors programmable in their channel geometry as well as in their connectivity. The choice of programmable MOS transistors is motivated as follows: On the one hand CMOS is currently the most widely used (and therefore cheapest) technology used to produce electronic circuits. MOS Transistors are the basic devices of this technology. On the other hand the configurability of the channel geometry of the transistors is inspired by the fact that most of the analog circuits known make use of different transistor sizes. Moreover it is expected to result in a smoother fitness landscape for the design problem given. Finally it was tried

to use a topology that contains as little bias from human experience as possible to allow the utilized algorithm to freely explore the design space and that is at the same time as homogeneous and symmetric as possible.

2. Architecture of the FPTA Chip

The topology of the programmable transistor array is shown in figure 1. The array consists of 128 P- and 128 NMOS transistors that are arranged in a checkerboard pattern. As denoted by the arrows crossing the transistor symbols the channel geometry and thus the electrical properties of the transistors can be varied. Altogether one out of five different lengths ($0.6, 1, 2, 4, 8\,\mu\mathrm{m}$) and 15 widths ($1...15\,\mu\mathrm{m}$) can be chosen for each of the transistors using a total of 7 configuration bits. The terminals of this programmable transistor can be connected to the nodes at the four cardinal points of the cell border via transmission gates (depicted in figure 1 as switches). Possible impacts of the transmission gates on the performance of the circuits implemented in the transistor array are described in [7].

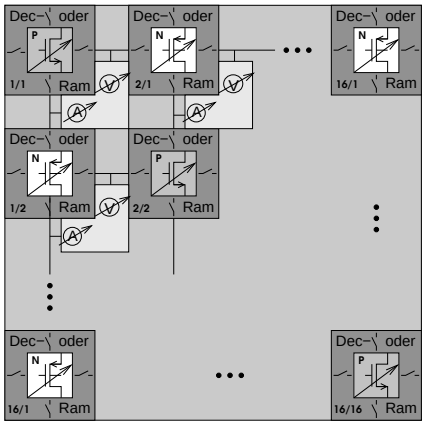

Figure 1. Schematic diagram of the 16×16 programmable transistor cells.

The chip was fabricated in a $0.6\,\mu\mathrm{m}$ CMOS technology and can be operated from a single $5\,\mathrm{V}$ power supply. The die size amounts to $33\,\mathrm{mm}^2$. The transistor array itself occupies only about $3.2\ \times 3.2\,\mathrm{mm}^2$ of silicon area while the rest is occupied by the necessary in- and output circuitry, pads and power supply lines. Figure 2 shows a micro photograph of the chip.

The block diagram of a single transistor cell is shown in figure 3. Each of the three terminals of the programmable transistor is either connected to a node of one of the four cell borders or to power or ground. In order to provide the necessary routing capabilities each of the nodes at the four

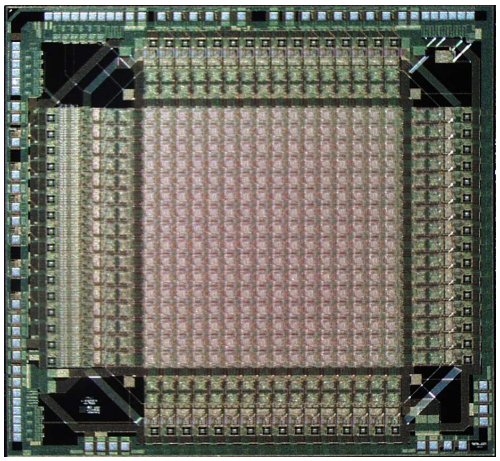

Figure 2. Micro photograph of the FPTA chip.

edges of the cell can be connected to the node of every other edge. The transistor cell contains four blocks containing six bits of SRAM to store the configuration of the cell. Six bits are used for the routing switches, three for each of the three transistor terminals and seven for the channel geometry of the transistor, which adds up to 22 used configuration bits per cell. Since all of the 24 bits (including the two unused bits) have to be set for all 256 cells, a complete configuration of the array requires $256 \times 24 = 6144$ bits (of which only 5632 contain information).

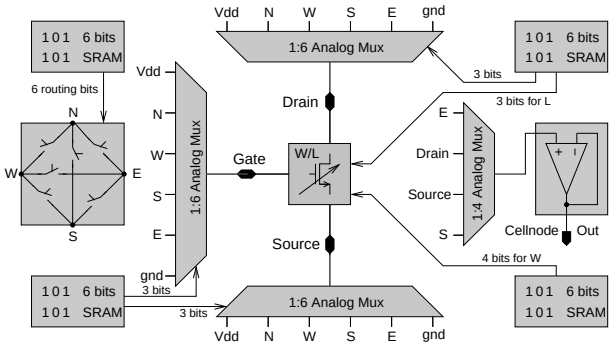

Figure 3. Block diagram of one transistor cell.

In order to serve as a design aid the hardware evolution system must allow to understand and therefore analyze the evolved circuits. The FPTA chip supports this feature by providing information about the voltage of and the currents flowing through the nodes between the transistor cells. Therefore every transistor cell contains a rail to rail unity gain buffer to read out the node voltages as well as the voltages across the transmission gates connecting the transistor cells (hidden in the multiplexers in figure 3) without af-

fecting the according signals. As shown in figure 3 this is achieved by multiplexing the Drain and Source terminals of the programmable transistor and the voltages of the nodes East and South to the unity gain buffer. To read out the voltage drops across the closed transmission gates the according two voltages have to be multiplexed to the buffer successively.

The layout of the NMOS transistor cell is depicted in figure 4, where dashed boxes are used to mark the different building blocks. Except for the programmable transistors P- and NMOS cells are identical. The transistor cell occupies

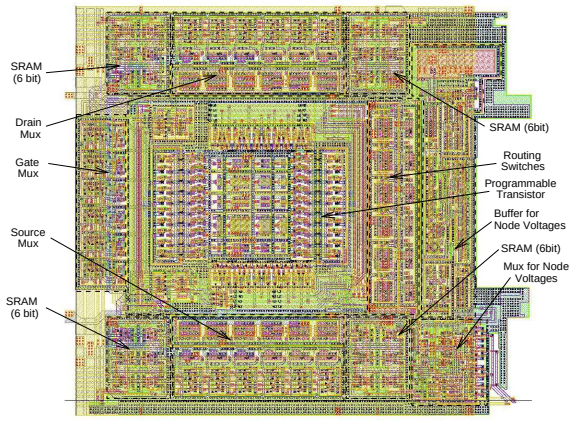

Figure 4. Layout of one complete NMOS cell.

a silicon area of about $200 \times 200\,\mu\mathrm{m}$. The area consumption is dominated rather by the necessary routing than by the devices themselves. This is partly due to the caution taken to prevent the chip from being destroyed by bad configurations: All metal lines that may have to carry high currents are made significantly wider than the minimum size width to avoid damage by electro-migration.

3 Measurement Results

First measurements of the FPTA chip show the full functionality of the transistor cell array: The SRAM can be written to and read out and the programmable transistors behave as expected, which is demonstrated by some transistor characteristics.

In order to measure the output characteristics of some of the PMOS transistor cells, the voltages at the three terminals were controlled with a HP 4155A semiconductor parameter analyzer that also measured the drain current I_d. The source terminal was held at 5 V while the drain voltage was varied between 5 and 0 V for a given gate voltage. The parameter analyzer was connected to these terminals via three pads that were connected to the two border cells 14/16 and 16/16 using one transmission gate for each connection. From there the signals were routed to the transistor

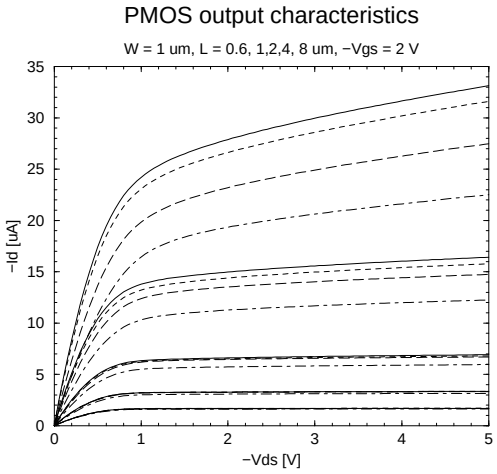

Figure 5. Measured output characteristics of four different PMOS transistor cells placed at different spots on the chip: Solid: 16/16, dashed: 15/15, long dashed: 10/10 dot-dashed 2/2

terminals through the routing switches of the array. Figure 5 shows the output characteristics of four different PMOS-transistors at the array positions 16/16, 15/15, 9/9 and 2/2. Curves are drawn for all five different lengths of the transistor.

First, it can be observed that all curves look like transistor output characteristics, vary with the programmed length as expected and look similar for different transistors. Second, it can be seen that the drain currents decrease for longer routing paths through the array. This is due to the finite resistance of the transmission gates used as routing switches.

In figure 6 the output characteristics of the PMOS transistor cell 16/16 in the lower right corner of the chip is compared to the simulation of a plain PMOS transistor as well as to one including the transmission gates used in the measurement for different transistor widths. While the more precise model of the transistor cell matches the measured curve quite well, the output currents of the transistor cell are always smaller than the ones from the simulation of the plain transistor. Again this is due to the finite resistance of the transmission gates and the discrepancy worsens for higher currents.

4 Future Plans

The next step to take is to integrate a genetic algorithm (GA) in the existing software and start evolution experiments. Candidate problems will be the evolution of specific dc behaviors as for example of an inverter, a comparator

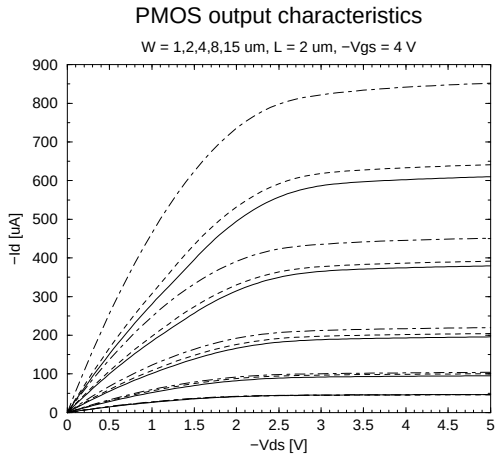

Figure 6. Comparison of the output characteristics of the measured cell 16/16 (solid line), a simulation including all transmission gates (dashed) and one of plain PMOS transistors (dot-dashed).

or simple logic gates. Depending on the success achieved, more difficult problems involving time domain constraints can be tackled. Furthermore it is planed to upgrade the printed circuit board carrying the FPTA with means to measure the die temperature as well as to control the voltage and current of the power supply of the transistor array, which is separated from the power supply of the rest of the chip. This can be used for two things. First, in case of overheating it allows to shut down the transistor array power, thereby avoiding self destruction of the FPTA chip. Second, it may be used in future experiments to evolve circuits working with low voltage power supplies, or optimized for power consumption. Eventually a larger transistor array may be required to evolve circuits with higher complexity. Therefore the FPTA chip provides the possibility to directly connect the transistor arrays of different dice by bond wires, allowing to form larger arrays.

5 Conclusions

The design of a Field Programmable Transistor Array that was fabricated in a $0.6\,\mu\mathrm{m}$ CMOS process is presented and the functionality of the transistor array is demonstrated. The chip is especially suited for hardware evolution of transistor level circuits because it provides the search algorithm with the essential devices of modern chip technology, namely CMOS transistors. The large variety of possible channel geometries that can be selected for every transistor is expected to result in relatively smooth fitness landscapes. This makes it easier for the evolutionary algorithm to find good solutions to the posed problems. The possibility to read out node voltages and currents inside the transistor array facilitates the analysis of evolved circuits, thus opening the door to a large number of interesting experiments.

6 Acknowledgment

This work is supported by the Ministerium f¨ur Wissenschaft, Forschung und Kunst, Baden-W¨urttemberg, Stuttgart, Germany.

References

[1] J. R. Koza, J. Yu, M. A. Keanne, W. Mydlowec *Evolution of a Controller with a Free Variable Using Genetic Programming.* Proceedings of the European Conference on Genetic Programming (EuroGP) 2000, R. Poli et al. Eds. pp. 91-105, Springer-Verlag, 2000.

[2] A. Thompson, P. Layzell *Evolution of Robustness in an Electronics Design.* Proc. 3rd Int. Conf. on Evolvable Systems: From Biology to Hardware (ICES2000), J. Miller et al. Eds. pp. 218-228, Springer-Verlag, 2000.

[3] M. Murakawa, S. Yoshizawa, T. Adachi, S. Suzuki, K. Takasuka, M. Iwata, T. Higuchi *Analogue EHW Chip for Intermediate Frequency Filters.* Proc. 2nd Int. Conf. on Evolvable Systems: From Biology to Hardware (ICES98), M. Sipper et al., Eds., pp 134-143, Springer-Verlag, 1998.

[4] A. Stoica, D. Keymeulen, R. Tawel, C. Sallazar-Lazaro, W. Li *Evolutionary experiments with a fine-grained reconfigurable architecture for analog and digital CMOS circuits* Proceedings of the First NASA/DOD Workshop on Evolvable Hardware, Pasadena, CA, July 19-21, IEEE Computer Society Press, pp. 76-84.

[5] A. Stoica, R. S. Zebulum, D. Keymeulen *Mixtrinsic Evolution*. Proc. 3rd Int. Conf. on Evolvable Systems: From Biology to Hardware (ICES2000), J. Miller et al. Eds. pp. 208-217, Springer-Verlag, 2000.

[6] A. Thompson *On the Automatic Design of Robust Electronics Through Artificial Evolution* Proc. 2nd Int. Conf. on Evolvable Systems: From Biology to Hardware (ICES98), M. Sipper et al., Eds., pp 13-24, Springer-Verlag, 1998.

[7] J. Langeheine, S. F¨olling, K. Meier, J. Schemmel *Towards a silicon primordial soup: A fast approach to hardware evolution with a VLSI transistor array*. Proc. 3rd Int. Conf. on Evolvable Systems: From Biology to Hardware (ICES2000), J. Miller et al. Eds. pp. 123-132, Springer-Verlag, 2000.