PAnDA: A Reconfigurable Architecture that Adapts to Physical Substrate Variations

James Alfred Walker, *Member*, *IEEE*, Martin A. Trefzer, *Senior Member*, *IEEE*,
Simon J. Bale, *Member*, *IEEE*, and Andy M. Tyrrell, *Senior Member*, *IEEE*

Abstract—Field programmable gate arrays (FPGAs) are widely used in applications where online reconfigurable signal processing is required. Speed and function density of FPGAs are increasing as transistor sizes shrink to the nanoscale. As these transistors reduce in size intrinsic variability becomes more of a problem and to reliably create electronic designs according to specification time consuming statistical simulations become necessary; and even with accurate models and statistical simulation, the fabrication yield will decrease as every physical instance of a design behaves differently. This paper describes an adaptive, evolvable architecture that allows for correction and optimization of circuits directly in hardware using bioinspired techniques. Similar to FPGAs, the programmable analog and digital array (PAnDA) architecture introduced provides a digital configuration layer for circuit design. Accessing additional configuration options of the underlying analog layer enables continuous adjustment of circuit characteristics at runtime, which enables dynamic optimization of the mapped design's performance. Moreover, the yield of devices can be improved postfabrication via reconfiguration of the analog layer, which can overcome faults induced due to variability and process defects. Since optimization goals are generic, i.e., not restricted to reducing stochastic variability, power consumption or increasing speed, the same mechanisms can also enhance the device's fault tolerant abilities in the case of component degradation and failures during its lifetime or when exposed to hazardous environments.

Index Terms—Reconfigurable architectures, intrinsic variability, bio-inspired algorithms, fault tolerance, evolvable hardware, adaptive hardware

1 Introduction

Application specific integrated circuits (ASICs) are the building blocks of many electronic systems. Over the past 20 years, these devices have rapidly improved in performance and function density, enabled by the continuous shrinking of technology sizes. Current devices can implement entire digital systems including multiple microprocessors, peripherals, and application specific blocks on a single die. As a consequence, the mind set of both users and manufacturers of such devices have settled at a point, where it is almost taken for granted that higher and higher densities of reliably operating logic can be achieved in perfect accord with Moores Law [1].

Device sizes are now approaching the atomistic scale, where the presence or absence of single doping atoms and structural irregularities at the atomistic scale are likely to affect the behavior of the device in a random manner. Time-consuming statistical SPICE simulations using specific, statistically enhanced device models become necessary to create reliable electronic designs that behave according to specification. Unfortunately, due to the statistical nature of these variations, the fabrication yield still decreases and failure rates increase significantly, because every physical instance of a design behaves in a stochastically different manner [2], [3], [4]. Even when verifying designs using accurate device models and statistical SPICE simulation, this will in the first instance only allow for a more accurate yield prediction, rather than overcome the effects of random variability, unless the additional information gained can be successfully used to improve the designs [5]. However, this is a highly complex task and it is not straightforward to decide how to change circuit topologies and device sizes to improve the performance of a large design or to bring the performance of the design back into specification.

The work presented in this paper proposes an approach that introduces mechanisms to overcome the effects of intrinsic stochastic variability by automatically optimizing designs postfabrication. In particular, this work focuses on enhancing field programmable gate array (FPGA) architectures for the following reasons: first, FPGAs are widely used in applications where online reconfigurable signal processing is required. Current devices feature high logic densities and programmable application specific macro blocks, i.e., multipliers, ALUs, memory, and can, therefore, be configured to implement customized digital systems comprising of processors, peripherals, and high-density logic, which places them between microprocessors and ASICs. Their versatility and the fact that they incorporate reconfiguration options already makes them suitable candidates for the proposed research. Second, the design most affected by intrinsic variability has been SRAM [6]. Since SRAM is used for storing configuration data and forming Lookup Tables (LUTs) in reconfigurable devices (it is worth noting that SRAM memory blocks are also present in FPGAs for system implementation), and hence can be

- *The authors are with the Intelligent Systems Group, Department of Electronics, University of York, Heslington, York, North Yorkshire YO10 5DD, United Kingdom. E-mail: {james.walker, martin.trefzer, simon.bale, andy.tyrrell}@york.ac.uk.*

Manuscript received 22 Aug. 2012; revised 15 Jan. 2013; accepted 25 Feb. 2013; published online 19 Mar. 2013.
Recommended for acceptance by K. Benkrid, D. Keymeulen, U.D. Patel, and D. Merodio-Codinachs.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TCSI-2012-08-0581.
Digital Object Identifier no. 10.1109/TC.2013.59.

0018-9340/13/$31.00 © 2013 IEEE Published by the IEEE Computer Society

operated at relatively low speeds compared with the actual applications, FPGA fabric has not been as severely affected by this kind of intrinsic variability as other ASICs, such as memory and processors. However, it is projected that the next "victim" of variability after SRAM will be latches, and this will have direct impact on FPGA architectures, which comprise of a large number of Flip-Flops of which latches are an essential part. This makes them a challenging architecture when developing and investigating mechanisms to overcome the effects of intrinsic variability.

Although conventional FPGAs offer mechanisms to replace and reroute designs postfabrication to achieve performance on an chip-by-chip basis, this is purely constrained to the digital layer (i.e., cannot modify the analog properties of the underlying devices) and is relatively coarse grained, as no slight performance adjustments can be made. Also, it is not always possible to perform such modifications during runtime without disrupting the operation of the mapped design. To be able to overcome these issues and achieve our goal of overcoming the effects of intrinsic stochastic variability, reconfiguration options are included in the design of the device (in addition to those already found in an FPGA), which allow for alterations of the characteristics of devices and components once they are fabricated and during operation (i.e., at the analog layer). This provides an access point for optimization algorithms to find configurations that may improve the circuit's performance and bring it back into specification.

Although introducing (additional) configuration options into a design generates considerable area overhead compared with conventional FPGA architectures, the overall benefit of continuing to use parts of the device that otherwise would have to be disabled because they do not work according to specification, or even worse, not being able to use the whole device make it worthwhile, especially for certain applications, such as timing-critical circuits, where the conventional FPGA reconfiguration mechanisms are not always adequate for recovering functionality to meet timing requirements.

This paper introduces the Programmable Analog and Digital Array (PAnDA) architecture, which proposes a novel reconfigurable variability tolerant architecture (illustrated in Fig. 5), which allows variability aware design and rapid prototyping by exploiting the configuration options of the architecture. Hence, PAnDA represents an adaptive, evolvable architecture that allows for correction and optimization of circuits directly in hardware using bioinspired techniques. Similar to FPGAs, the PAnDA architecture provides a digital configuration layer for circuit design. However, access to additional configuration options of the underlying analog layer enables continuous adjustment of circuit characteristics at runtime, which enables dynamic optimization of the mapped design's performance.

It is possible to see these additional features of the PAnDA architecture could have a number of potential benefits and/or applications. For example, the power versus performance tradeoff of a mapped design could be optimized autonomously and dynamically, so that it can be reconfigured to a high-speed mode that consumes more power when the mapped design is under a heavy load and alternatively dynamically reconfigured to a power saving mode when the mapped design is idle. The yield of devices can be also improved postfabrication via reconfiguration of the analog layer, which can overcome faults induced due to process variability. Furthermore, as the performance and reliability of the device degrades over its lifetime due to ageing effects [7], the same mechanisms could can once again be utilized to dynamically reconfigure the PAnDA architecture to improve performance and/or recover operation. Since optimization goals are generic, i.e., not restricted to reducing simply variability, power consumption or increasing speed, the same mechanisms have a great potential to be more generic and can also enhance the device's fault tolerant abilities in the case of component degradation and failures during its lifetime or when exposed to hazardous environments. Examples of previous work where postfabrication optimization has been shown to be beneficial in terms of yield, fault tolerance, and/or performance can be found in [8], [9], [10].

Results presented in this paper are obtained from statistical SPICE simulation using blocks of the PAnDA architecture. A prototype PAnDA chip is currently being fabricated in 40-nm process technology and is expected to be delivered in April 2013.

This paper is organized as follows: Section 2 discusses the problem of stochastic variability in cutting edge CMOS processes, Section 3 discusses the limitations of current FPGA architectures and how they are affected by stochastic variability, followed by how inspiration can be taken from FPTA architectures and variation aware design to overcome these limitations in Section 5. The proposed PAnDA architecture is introduced in Section 6 before two case studies that exploit the features of the PAnDA architecture to overcome stochastic variability are discussed in Section 7. Finally, conclusions and future work are discussed in Section 8.

2 Stochastic Variations in Deep Submicron CMOS Processes

The precision of individual device and interconnect parameters has traditionally been dependant on constraints within the manufacturing process (e.g., distortions in the photo-lithography process and strain) and has been considered relatively deterministic in nature [11]. As channel lengths shrink below $50\ \mathrm{nm}$, unavoidable stochastic variability due to the discrete location of individual dopant atoms within the device channel are becoming increasingly significant [12]. Many advances have been made to reduce the loss of precision caused by the manufacturing process (e.g., optical proximity correction [13], uniformly dense layout [14]). However, the fundamentally granular nature of matter cannot be overcome and the impact will increase as the technology continues to shrink further [2], [15], [16].

Device variability occurs in both the spatial and temporal domains, and for each domain this can additionally be split into deterministic and stochastic fluctuations. Spatial variability occurs when the produced device shape differs from the intended design, including uneven doping profiles, nonuniformity in layer thickness, and polycrystalline surfaces [17]. This variability is found at all levels: over the lifetime of a fabrication system (i.e., wafer-to-wafer), across a wafer of chips (i.e., die-to-die), between cells within

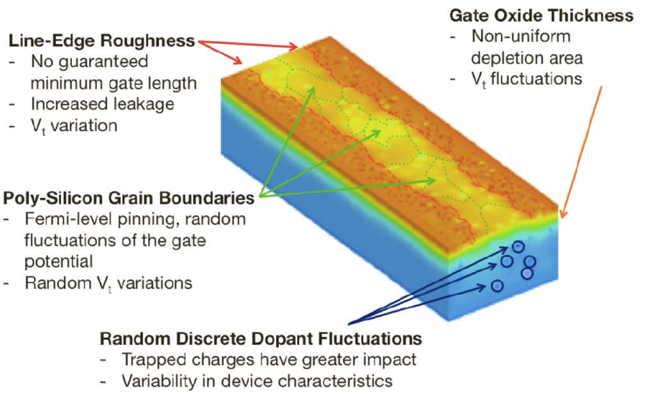

Line-Edge Roughness
- No guaranteed minimum gate length
- Increased leakage
- V_t variation
Gate Oxide Thickness
- Non-uniform depletion area
- V_t fluctuations
Poly-Silicon Grain Boundaries
- Fermi-level pinning, random fluctuations of the gate potential
- Random V_t variations
Random Discrete Dopant Fluctuations
- Trapped charges have greater impact
- Variability in device characteristics

Fig. 1. Intrinsic parameters fluctuations within a simulated 35-nm device [2].

a VLSI chip, and between individual devices within that cell (i.e., within-die) [18]. Temporal variability includes the effects of electromigration, gate-oxide breakdown, and the distribution of NBTI.[1] Such temporal variability has been estimated and the effects combined to give an expected lifetime calculation for an individual device, or simulated to determine the compound effect across a whole chip [19], [20]. While deterministic variability can be accurately estimated using specific design techniques, intrinsic parameter fluctuations (summarized in Fig. 1) can only be modeled statistically and cannot be reduced with improvements in the manufacturing process [21].

3 Limitations of Current FPGA Architectures

Current FPGA architectures have been continuously improved over the past 20 years. They have directly benefited from the advancements in process technology, which made it possible to significantly increase logic density as a direct result of reducing feature sizes (see Fig. 2) without the need for creating conceptually new FPGA architectures. As a consequence, programmable logic elements of all models and vendors generally consist of LUTs, MUXes, and Flip-Flops that are arranged in similar topologies.

Continued demands of increasing speed and decreasing power consumption of more complex designs, such as processors, memory, and high-speed interfaces, are traditionally addressed via incorporating an increasing number of custom building blocks (hard macros) that are embedded into the FPGA fabric by the FPGA industry. This has been a viable approach, because FPGA technology has not been affected by intrinsic variability in fabrication due to the way SRAM is used in FPGAs: although SRAM is used for storing configuration data and LUTs in reconfigurable devices, the impact of variability on the FPGA fabric has not been as severe when compared with the affects on ASIC designs, such as memory and processors, due to the fact that FPGAs are generally operated at relatively low speeds compared with ASIC designs. In particular, the SRAM used for building LUTs is only rewritten at the time of programming the FPGA and is only read during operation. However, there may be severe limitations to this strategy when faced with the challenges of electronic design when shrinking device sizes to the atomistic scale, even when moving to new device technologies such as SOI and FinFETs, given the likely affect of variability on latches in the future [6]. It is anticipated that this will have a direct impact on FPGA architectures, which comprise of a large number of Flip-Flops, of which latches are an essential part. In addition, SRAM cells may become unstable even in the static case, which would randomly change the configuration of an FPGA and effectively change the logic function.

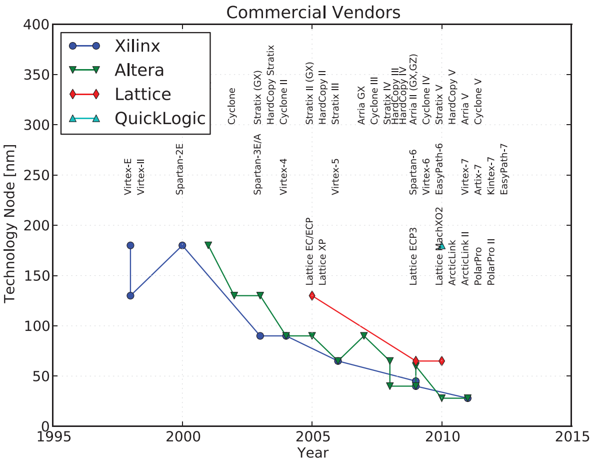


Fig. 2. The adoption trend of smaller semiconductor technology nodes by major FPGA vendors.

Hence, to accommodate the increased variability of individual device characteristics there is a need for novel device architectures and circuit design methodologies. To further advance reconfigurable architectures, it is essential to identify and review current and future challenges both from an electronic design, and from an emergent technologies point of view. To overcome these challenges, it is necessary to take a more holistic approach to electronic design problems, i.e., understanding and tackling problems on multiple levels rather than considering the device, analog, digital or system levels separately. Particularly in the case of reconfigurable architectures, there are great opportunities to include mechanisms that allow for inherent correction and optimization of shortcomings due to variability or other parameters, because facilities to reconfigure the device are already present and the additional area overhead introduced when adding more configuration options can be more easily alleviated via yield recovery and postfabrication performance optimization.

4 Current FPTA Architectures

The design of analog circuits, particularly creating new topologies, is a nontrivial task and there exists no automatic mapping that can, for instance, translate a transfer function into an optimized transistor circuit. Hence, there are a number of fine-grained, reconfigurable architectures aimed at rapid circuit prototyping and design automation via optimization algorithms. Most examples come from the fields of evolutionary computation and evolvable hardware, which provides model free optimization algorithms that are the most suitable approaches when there is no formal design methodology. The most relevant

[1] Negative-Bias Temperature Instability.

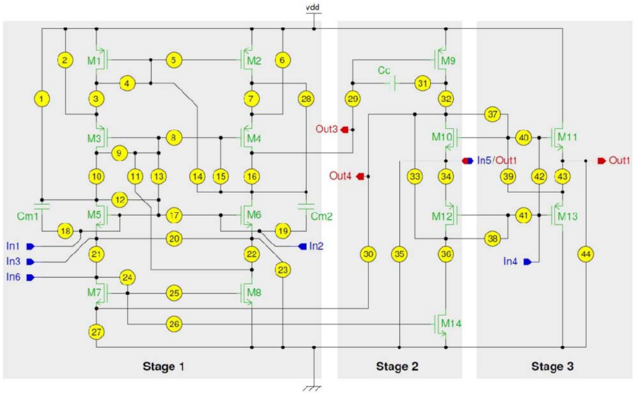


Fig. 3. The architecture of the transistor cell from the JPL FPTA2 [9]. The encircled numbers denote switches.

of these reconfigurable architectures are the two Field Programmable Transistor Array (FPTA) architectures—the FPTA0, FPTA1, and FPTA2 from NASA's Jet Propulsion Laboratory (JPL) and the FPTA and FPTA-2 from the University of Heidelberg.

The FPTA0 and FPTA1 [22] from JPL are predecessors to the FPTA2 [9]. FPTA0 consists of a single cell that comprises eight transistors connected by 24 switches. The latter two chips (FPTA1 and FPTA2) feature the same cell architecture (shown in Fig. 3) based on a design of an OPAMP with two output stages, but consist of a different number of cells. The FPTA1 features 12 cells and can be thought of as a prototype for the FPTA2, which features an array of 8×8 cells. However, each connection between transistors in a cell is replaced with a switch, which enables fine-grained reconfiguration on the transistor level. In addition to this, the cells of FPTA2 also feature a number of programmable photo diodes, resistors and capacitors. The cells in the FPTA2 are connected to their four nearest neighbors.

The applications tackled by the JPL FPTAs concentrate on recovering functionality in the case of harsh environments (i.e., extreme temperatures and radiation impact as it would occur in space). Evolutionary algorithms are normally used to create circuit designs for these FPTAs, particularly in the case of FPTA2.

The Heidelberg FPTA [23] consists of an array of 16×16 programmable CMOS transistors. A programmable transistor cell (PT) comprises a matrix of 5×4 CMOS transistors with variable widths and lengths that share common source, gate, and drain connections, as illustrated in Fig. 4. By switching different subsets of the matrix on or off, the effective width and/or length of the PT and, therefore, its characteristics becomes adjustable. In addition, a PT can also be used to route a signal without connecting the transistor at all. PTs are connected to their four nearest neighbors.

In this respect, the Heidelberg FPTA represents the most fine-grained and general purpose architecture of the ones presented. A wide range of applications has been realized on this platform, including logic gates, analog filters, comparators, DACs, ADCs, and OPAMPs [10].

The proposed second version, the FPTA-2 [24], aims to reduce the number of switches to reduce the influence of parasitic effects while retaining the fine granularity of the Heidelberg FPTA. This is achieved via PTs that comprise programmable matched pairs of transistors and by shifting the architecture from a nearest neighbor connected mesh toward a crossbar architecture.

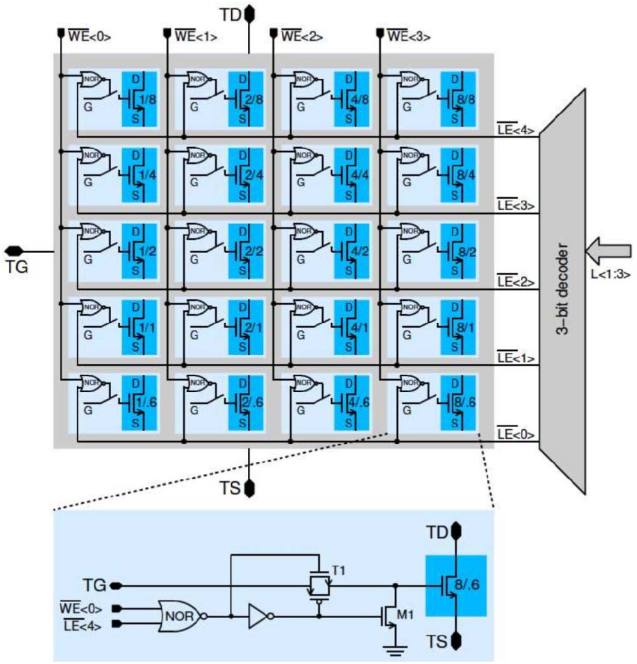


Fig. 4. The architecture of the PT from the Heidelberg FPTA [23].

To avoid overly constraining the circuit topologies that can be realized on both the JPL and Heidelberg FPTAs, a large amount of configuration circuitry is required. The additional configuration circuitry (and memory to store the configuration) introduces significant parasitic effects (i.e., capacitance, resistance), which have a major impact on the characteristics of the functional circuit. As a consequence, the ratio between design and configuration circuitry becomes too small, the operation speed decreases, and the distortion/noise increases.

However, there are other application areas where the presence of this additional configuration circuitry is not a drawback, i.e., when fault tolerance and adaptivity are required. The JPL FPTAs have been shown to withstand (or recover from) extreme temperatures or radiation impact. In these cases, transistor-level circuits may have advantages over digital logic circuits, due to their finer-grained nature; depending on where the error occurs it may not result in an entirely different result (e.g., bit error), but rather only increase, for instance, the noise margin of a circuit. In more severe cases, the finer granularity might also be advantageous, as the larger configuration space increases the likelihood of finding an alternative configuration with the same, or at least similar, functionality that can be realized using only undamaged resources. In addition, there is scope to use feedback as a mechanism to automatically adjust the circuit's behavior.

5 Combining Inspiration from FPTAs and Variation-Aware Design

It has been discussed in Section 3 how stochastic variability in CMOS transistors impedes designing reliable logic standard cells, which are the fundamental building blocks

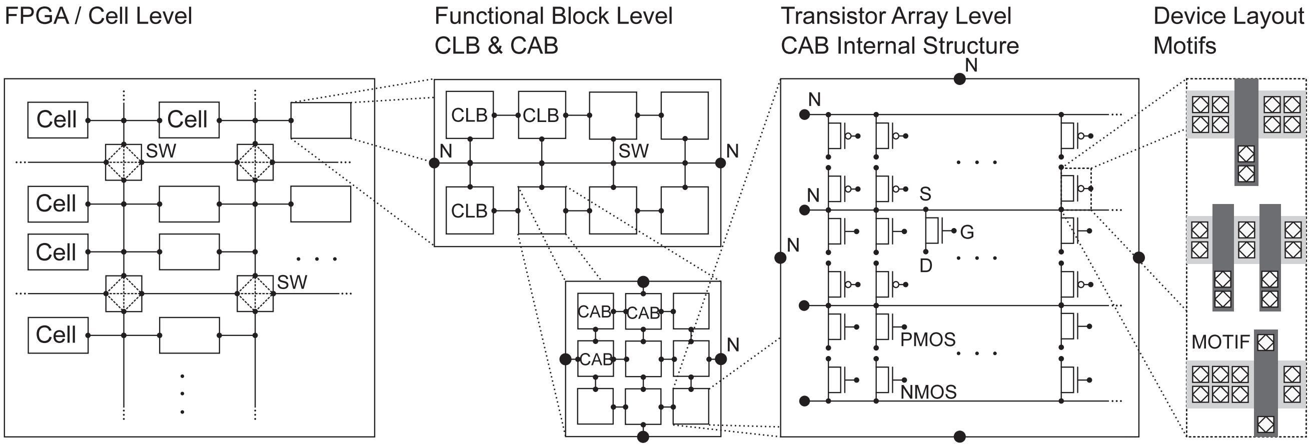


Fig. 5. The proposed conceptual overview of the PAnDA architecture. The Cell and CLB levels represent common groupings and components found in current commercial FPGAs. The Transistor Array (referred to as CTs) are a similar concept to the PTs in the Heidelberg FPTA. The CAB level is the interface layer between the FPGA and FPTA concepts and allows groups of CTs to be configured to form logic functions. The CAB and CT levels are unique to PAnDA and are not found in current FPGAs. While conventional FPGAs can only be reconfigured digital level (cell and CLB), PAnDA offers additional configuration options on the analog level (CAB and CT).

of FPGAs. Most severely affected designs are SRAMs and latches, which are fundamental elements of any current programmable logic architecture. The results in [5], [25], which have been obtained from statistical SPICE simulations, suggest that optimizing the widths of transistors in standard cells can improve their variability tolerance, speed, and power consumption. The previous section has also shown that it is possible to design and optimize analog CMOS circuits in hardware using FPTAs. Therefore, if FPTA-based reconfiguration mechanisms to alter device sizes according to [5] could be incorporated in hardware, it would be possible to optimize designs postfabrication in a dynamic fashion at runtime. This would not only have the advantage of being able to enhance variability tolerance and performance for a specific design, but could also account for variations between different devices. Due to the large numbers of statistical measurements necessary to analyze the effects of stochastic variability, it will be orders of magnitude faster to perform optimization directly in hardware, rather than in SPICE simulation. It would also be possible to optimize much larger circuits than are currently feasible in simulation. In addition, such a multifaceted reconfigurable device will provide significant resources to deal with both pre- and postfabrication faults.

The aim of the PAnDA project is to develop a novel FPGA architecture, which combines aspects of the digital reconfiguration layer of an FPGA (i.e., CLBs) with the analog reconfiguration layer of an FPTA (i.e., PTs), while trying to minimize the ratio of design to configuration circuitry of the device. The outcome will be the PAnDA architecture, onto which digital designs can be synthesized and then optimized on both the digital and analog layers on a device-by-device basis to overcome the challenges arising when shrinking device sizes to the nanoscale. Further optimization via dynamic reconfiguration of the PAnDA architecture at runtime will also provide increased reliability to overcome faults and device degradation during its lifetime and also design-specific power/performance tradeoffs depending on the operational state of the mapped design.

A proposed conceptual overview of the PAnDA hierarchy is illustrated in Fig. 5. As can be seen from the figure, the most important difference to a conventional FPGA is that the PAnDA architecture extends to a finer granularity of building blocks. The reason for introducing these additional levels is to both create and provide access to a lower design level, i.e., the transistor and function level, which enables changing the properties of the architecture on the analog level in addition to the more traditional digital level.

The envisioned PAnDA architecture comprises configurable transistors (CTs), configurable analog blocks (CABs), configurable logic blocks (CLBs), logic cells, and interconnect. CLBs, logic cells and interconnect are also present—and have similar structure—in current commercial FPGA architectures, whereas CTs and CABs are unique to PAnDA. The designs of the PAnDA CTs and CABs are introduced in this paper and are described in the following sections. The CLB and Cell levels will be investigated and designed in future work and included in future prototypes of the PAnDA chip.

6 The PAnDA Architecture

With the PAnDA we propose a novel FPGA architecture, which aims to overcome challenges arising when shrinking device sizes and provide significantly more resources for reconfiguration [26]. To achieve this, the PAnDA architecture features CTs, the effective device sizes of which can be altered by configuring them in various ways to compensate for intrinsic stochastic variability affecting the characteristics of the individual devices. As can be seen from Fig. 5, the architecture allows designs access to the transistor and function levels of the device. This enables designs not only to configure the logic levels, as would traditionally be the case, but also changing the underlying analog level is possible.

During the PAnDA project, we aim to design and fabricate a number of full custom prototype chips to test and analyze features of the proposed PAnDA architecture

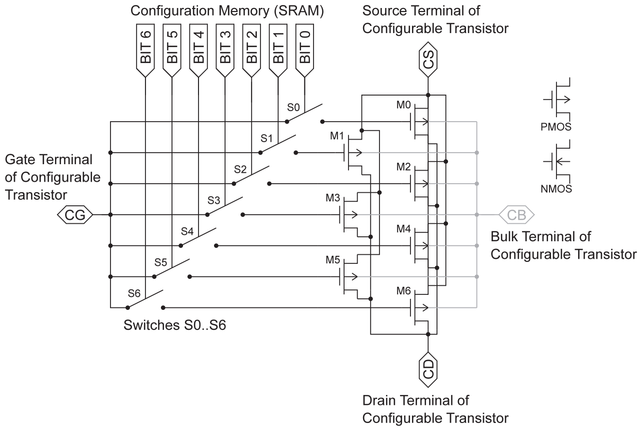

Configuration Memory (SRAM)
BIT 6
BIT 5
BIT 4
BIT 3
BIT 2
BIT 1
BIT 0
Source Terminal of Configurable Transistor
CS
S0
S1
S2
S3
S4
S5
S6
M0
M1
M2
M3
M4
M5
M6
PMOS
NMOS
Gate Terminal of Configurable Transistor
CG
CB
Bulk Terminal of Configurable Transistor
Switches S0..S6
CD
Drain Terminal of Configurable Transistor

Fig. 6. A PMOS CT of the PAnDA architecture. The transistor sizes used in the PAnDA CT are $L_{0\ldots6} = 40\ \mathrm{nm}$ and $W_{0\ldots6} = [120, 120, 140, 160, 180, 200, 220]\mathrm{nm}$, which allows for configuring the CT a range of 120 and $1{,}140\ \mathrm{nm}$.

in Fig. 5. The first of these prototype is currently being fabricated in a 40-nm technology and is expected early 2013. The aim of the first PAnDA prototype is to test the CT and CAB architectures, which are described in this section. Although the routing within a CT or CAB is fixed, the interconnect between CABs on the prototype chip uses a multiplexer-based switch matrix. However, due to the high computational demand required to perform simulations of the architecture, a simplified interconnect was used in the case studies presented Section 7, in which the CABs directly connect to each other. The higher level entities (i.e., CLB and Cell) shown in Fig. 5 have not yet been designed or implemented and are deliberately left for future work and the inclusion in a later prototype PAnDA chip.

6.1 Configurable Transistor

The CT is the smallest reconfigurable element of the PAnDA architecture. As can be seen from Fig. 6, the CT is a device formed from seven PMOS or seven NMOS (depending on whether it is a NMOS or PMOS CT) transistors $(M0 \ldots M6)$ that are connected in parallel. Each of the seven transistors can be individually turned on or off via opening or closing a switch $(S0 \ldots S6)$ that connects their gate to a common gate connection (CG). The states of the switches are controlled via configuration bits stored in a configuration static random access memory (SRAM).

Although the PAnDA CT is similar to the PT in the Heidelberg FPTA, it is much smaller in area, as it only has seven transistors instead of 20. This also reduces the amount of configuration switches required on the gate of each transistor. This reduction is achieved as all transistors in a CT are minimum length (as you would find in a digital standard cell library), whereas the PT allows devices of different lengths, which are required for analog design. The PAnDA CT also has a much finer uniform increment in transistor width than the PT, which has a quadratic spaced increment.

A PAnDA CT will consume at least 7 × more area than a conventional device. If the conventional device were fabricated larger, in terms of both length and width, then it would take up far less than 7 × the area and suffer less from the effects of intrinsic variability. However, such a device would also consume more power and would not benefit from the performance benefits of a smaller device, for which the fabrication process has been optimized. It would also be of a fixed size.

The PAnDA CT design exploits the fact that a number of CMOS transistors of the same gate length (L) can be connected in parallel to form a device that is equivalent to a single transistor of the same length and with the sum of the individual widths (W). For example, connecting two transistors in parallel, where the size of one is $\frac{W}{L} = \frac{120\ \mathrm{nm}}{40\ \mathrm{nm}}$ and the size of the other is $\frac{W}{L} = \frac{180\ \mathrm{nm}}{40\ \mathrm{nm}}$, would result in a device that is equivalent to a transistor with $\frac{W}{L} = \frac{300\ \mathrm{nm}}{40\ \mathrm{nm}}$. This allows the characteristics (i.e., the drive current) of the CT to be controlled to overcome the effects of intrinsic variability. Alternatively, the CT may be able to take advantage of the effects of intrinsic variability to implement a device with a characteristic outside of what would be classed as normal device behavior (i.e., lower power consumption).

The ability to control the CT characteristics provides many benefits that are not found using a fixed size single device, such as the ability to speed up/slow-down certain paths of a digital synchronous circuit to meet critical timing requirements and reduce power consumption. The overall clock speed of a synchronous circuit could also be improved if a faster variant of the devices that make up the critical path(s) were discovered. Alternatively, it would also be possible to change the mode of a circuit dynamically at runtime, to increase performance or reduce power consumption depending on the load or potentially the charge remaining in the battery (if it were in a portable device). Additionally, the CT is inherently fault-tolerant, so if one or more of the CT transistors develops a fault, another transistor or combination of transistors could be dynamically turned on to regain functionality of the design, albeit with a slightly different speed and/or power consumption. Such a fault tolerant capability would be much faster to regain circuit functionality with similar performance than re-configuring and rerouting part, or in the worst case scenario all, of the design.

The transistor sizes used in the PAnDA CT are $L_{0\ldots6} = 40\ \mathrm{nm}$ and $W_{0\ldots6} = [120, 120, 140, 160, 180, 200, 220]\mathrm{nm}$. Minimum width and length are constrained by the smallest device sizes that are allowed according to the design rules of the 40-nm technology in which the PAnDA prototype will be fabricated. The increment size is a tradeoff between the width precision required (a minimum of $5\ \mathrm{nm}$ is required by the design rules) and the number of transistors in a CT. An increment of $20\ \mathrm{nm}$ is chosen as half the minimum feature size of the technology used, which will provide a suitable value range when optimizing for effects of stochastic variability. The maximum size is determined by the effective width of combining the smallest two transistors in the CT minus the increment (i.e., $120\ \mathrm{nm} + 120\ \mathrm{nm} = 240\ \mathrm{nm} - 20\ \mathrm{nm} = 220\ \mathrm{nm}$). This allows the CT to have a continuous range of widths between 120 and $1{,}140\ \mathrm{nm}$ in increments of $20\ \mathrm{nm}$.

The $I - V$ curves that characterize all width configurations of an NMOS CT both with and without the effects of stochastic variability are shown in Fig. 7. The data for the $I - V$ curves were generated using Gold Standard Simulations Ltd's (GSS) RandomSpice statistical circuit simulator [27] with 100 stochastic variability runs to capture

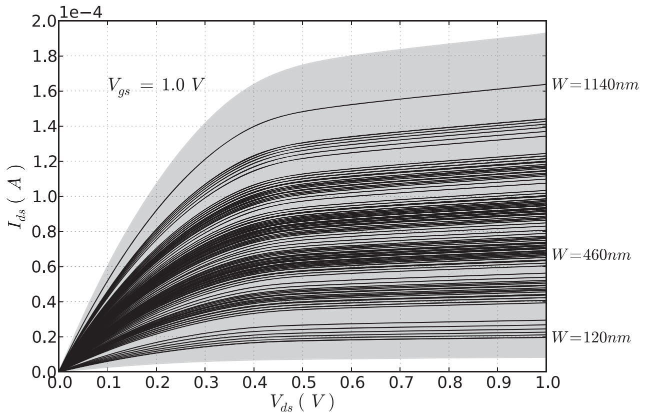


Fig. 7. $I-V$ characteristics for all 128 possible width configurations of a NMOS CT (lines). The drain-source voltage (V_{ds}) is plotted on the x-axis, the drain-source current (I_{ds}) is plotted on the y-axis, and the gate-source voltage (V_{gs}) is $1V$. The effective width of the CT corresponding to certain $I-V$ curves are shown on the right. When the effects of stochastic variability upon these 128 width configurations is considered, the behavior of each $I-V$ curve becomes less distinct and falls within the shaded region.

the statistical variations. RandomSpice uses a library of stochastic variability enhanced BSIM4 device models, which are derived and extracted by GSS from 3D atomistic TCAD simulations. From Fig. 7, the increase in current that the NMOS CT can produce as the width of the NMOS CT increases can be seen. However, when looking at the effect stochastic variability has upon the CT, it is no longer possible to see a clear correlation between CT width and output current. In fact, it is possible that a NMOS CT with a larger width may produce less current than a CT configured with a smaller width (or vice versa) due to the effect of stochastic variation upon the device.

The 128 different configuration options for the CT allow it to be configured with 47 unique widths between 120 and $1{,}140\ \mathrm{nm}$, and there is generally more than one possible combination of transistors to achieve a certain width of the CT, as can be seen in Fig. 8. This redundancy will help when optimizing for stochastic variability, because it is expected that different combinations for the same width of the CT will exhibit different behavior in terms of stochastic variability. In addition, such redundant behavior will prove invaluable when considering partial, dynamic reconfiguration, for instance, for fault recovery. This is demonstrated in Fig. 9, which shows the $I-V$ curves for the five different configuration options for a NMOS CT with width $460\ \mathrm{nm}$. Once again the data for the $I-V$ curves were generated using GSS's RandomSpice simulator [27] with 100 stochastic variability runs to capture the statistical variations.

Providing more reconfiguration capabilities, however, comes at the cost of adding additional configuration circuitry, which introduces parasitic effects (i.e., resistance and capacitance) that may degrade the performance of the CT. Hence, there is always a tradeoff to be made between lowering the performance and increasing variability and fault tolerance of the design. In this case, the ratio between transistors that process user signals and those used for configuration purposes is $1:11$. This number is calculated by comparing the seven "user" transistors with the number of transistors required for implementing switches (5T) and configuration SRAM (6T). Note that this number will slightly increase when also taking memory controllers and interconnect into account at the CAB level, which is discussed in Section 6.2.

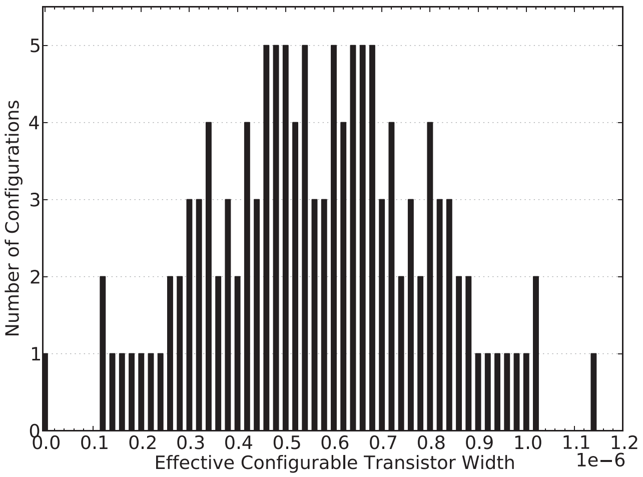
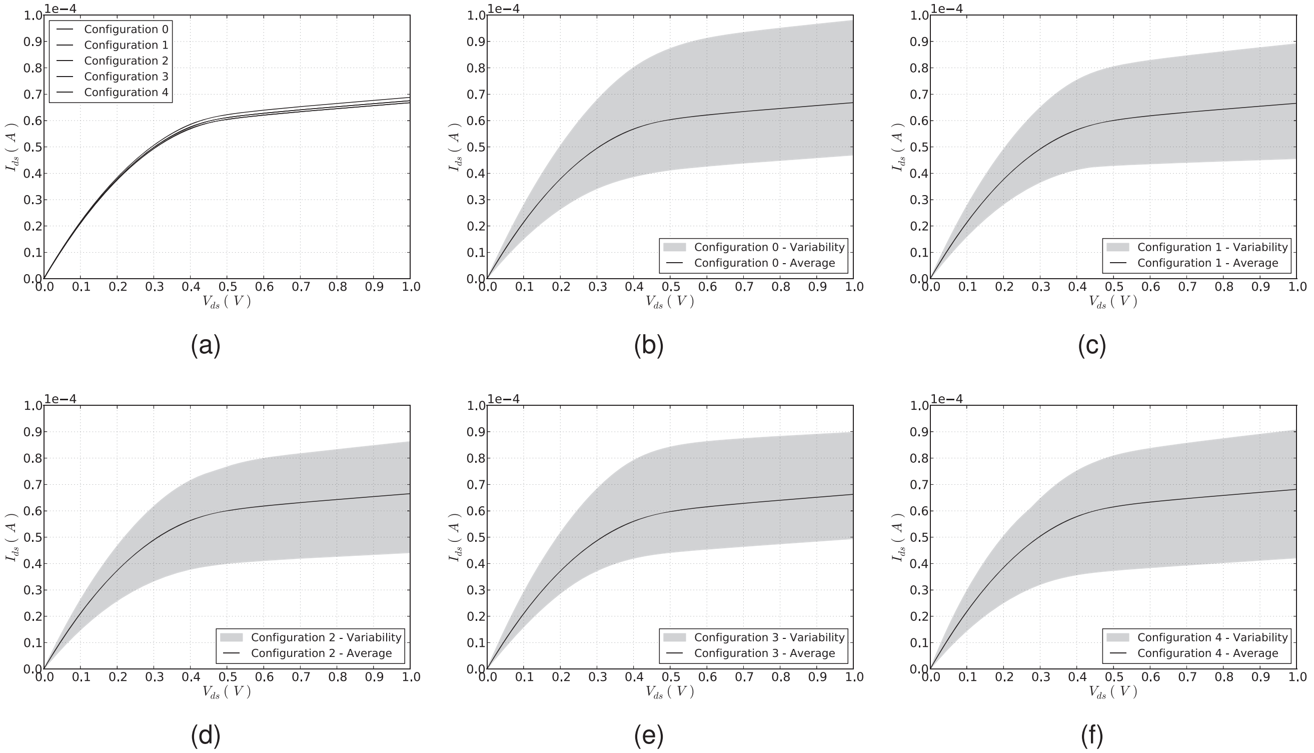


Fig. 8. The distribution of all 128 possible width configurations of a NMOS CT in the PAnDA architecture.

6.2 Combinational Configurable Analog Block (CCAB)

CCABs represent the next level of entities to the CTs, and is denoted as the CAB level in Fig. 5. The purpose of a CCAB within the PAnDA architecture is to represent basic logic blocks that can be implemented on the FPGA fabric. In this respect, a CCAB is the PAnDA equivalent of a LUT in a conventional FPGA. Due to the use of CTs and the native transistor implementation of the logic function (instead of addressing value stored in memory like a LUT), the CCAB requires significantly more area than a conventional LUT. However, the tradeoff for increased area is the ability to modify the speed, power consumption, and drive strength of the logic function that the CCAB implements, which is not possible with a conventional LUT in which all transistor sizes are fixed. This will allow the CCAB to compensate for the effects of intrinsic variability at the logic level by reconfiguring its CTs, in addition to providing tolerance to faults that occur in the CTs that make up the logic function.

With conventional CMOS design in mind, the PAnDA CCAB consists eight PMOS CTs and eight NMOS CTs arranged in a CMOS-like structure, where all source and drain terminals are directly connected and are nonconfigurable (see Fig. 10). The decision to remove the switches in the source-drain paths was made to reduce the ratio of design to configuration circuitry and also because previous work on FPTA architectures has shown that the parasitic effects (i.e., resistance and capacitance) that are introduced by inserting switches in the source-drain paths between MOSFETs can have a significant impact on the characteristics of the functioning circuit [9], [10]. The CCAB architecture could be implemented on the Heidelberg FPTA; however, all the routing switches between the source, gate, and drain connections of each PT would significantly degrade the overall performance of the design. The CCAB also contains a configurable interconnect block, which uses an analog multiplexor-based switch matrix to route signals from the CCAB inputs to the gate terminals of



(a) (b) (c)

(d) (e) (f)

Fig. 9. $I - V$ characteristics of the five different configurations for width $460\ \mathrm{nm}$ of a NMOS CT (a) and the affect of stochastic variability upon each of the five width configurations (b-f). The drain-source voltage (V_{ds}) is plotted on the x-axis, the drain-source current (I_{ds}) is plotted on the y-axis, and the gate-source voltage (V_{gs}) is $1\ V$. The figure illustrates that when variability is taken into account, the five different width combinations for the same effective width could each perform significantly different.

the CTs. A CCAB features three inputs and two outputs (one of which is an inversion of the other output) and is illustrated in Fig. 10.

The configurable interconnect maps the CCAB inputs to the gate terminals of the respective PMOS and NMOS CTs for the configured logic function and is controlled via configuration bits stored in a configuration SRAM. The configurable interconnect block has 14 outputs (labeled P1-P7, N1-N7 in Fig. 10), each of which connect to the correspondingly labeled gate of one of the CTs in the CCAB. Each output serves either to involve the corresponding CT in a logic function, to render the CT effectively transparent in terms of the role it plays in the logic function or to turn off the corresponding CT. The CCAB currently has eight different configuration options, providing 16 different 1-, 2-, and 3-input logic functions, shown in Table 1, using either the standard output or inverted output.

To keep the ratio between transistors that process user signals and those used for configuration purposes low also on the CCAB level, the configuration options of the CCAB have been constrained in such a way that the basic logic functions that can be realized are restricted to those in Table 1. Although this saves configuration circuitry and

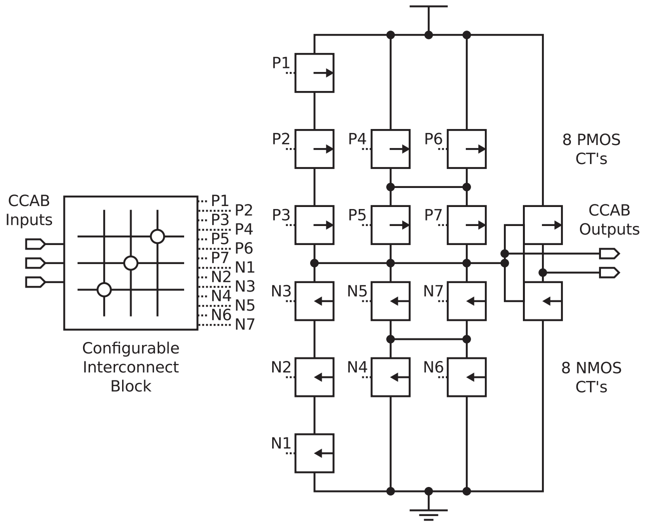


Fig. 10. CCAB of the PAnDA architecture. The configurable interconnect block maps the CCAB inputs to the gate connections of all PMOS (P1-P7) and NMOS (N1-N7) CTs (denoted by dashed lines), which determines the logic function of the CCAB.

TABLE 1
The 16 Configurable Functions of the PAnDA CCAB

Configuration	Function (Standard Output)	Function (Inverted Output)
0	Inverter	Buffer
1	NAND-2	AND-2
2	NOR-2	OR-2
3	NAND-3	AND-3
4	NOR-3	OR-3
5	AND-OR-INV-21	AND-OR-21
6	OR-AND-INV-21	OR-AND-21
7	PROG-AND-OR-INV-2	PROG-AND-OR-2

The PAnDA CCAB has two outputs (one is an inversion of the other), so can perform two functions for each of the eight configurations.

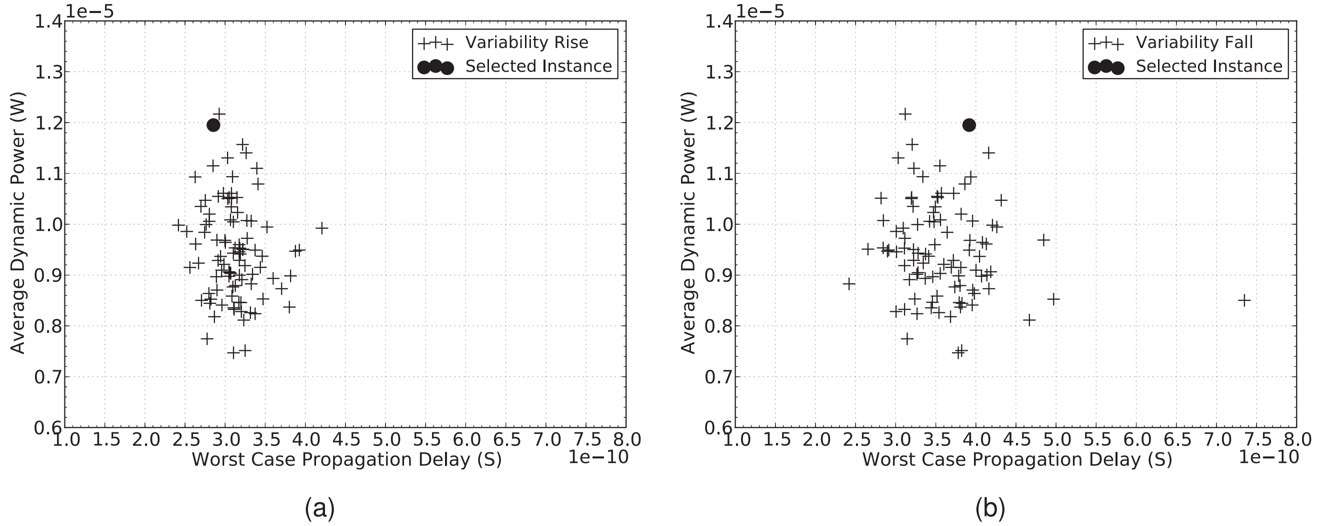


Fig. 11. The effect of stochastic variability upon 100 physical instances of the PAnDA architecture configured to perform the ISCAS C17 Benchmark. The results show both the worst-case rise (a) and fall (b) propagation delay with respect to average dynamic power consumption. The physical instance selected for optimization is highlighted.

improves the performance of the design, the ratio for the CCAB is $1:23$ (also taking CTs into account), which is slightly higher than in the case of the CT due to the additional configurable routing required to map the basic logic functions and inputs.

The fundamental ideas and architectural features have been described and a number of case studies will now be presented to demonstrate performance improvements that can be achieved using the PAnDA architecture.

7 Case Studies

The ISCAS C17 circuit and a 2-bit multiplier are used as benchmarks to demonstrate how the PAnDA architecture may improve the performance of circuits that are mapped onto it. The performance of the mapped circuits is assessed by calculating the propagation delay of the slowest transition, as this will limit the overall speed of the design and the average dynamic power consumption across all transitions. Both circuits are manually mapped to the PAnDA architecture and implemented by setting the appropriate configuration options in the netlist implementing the PAnDA CCABs including configuration circuitry. An automated design mapping methodology is planned for the future and will be discussed in Section 8.

This work represents an initial study of the PAnDA architecture in statistically enhanced SPICE simulation. To be able to accurately simulate the effects of stochastic variability, RandomSpice and a library of statistically enhanced compact models from GSS are used once again. These models make it possible to simulate the creation of "virtual physical instances of a chip," i.e., the results presented are obtained with specifically configured SPICE decks that simulate the random differences between dies after fabrication. This allows us to predict how the performance of mapped designs will vary on the PAnDA architecture when fabricated in modern deep submicron processes. A prototype chip of the PAnDA architecture is currently being fabricated in 40-nm CMOS technology.

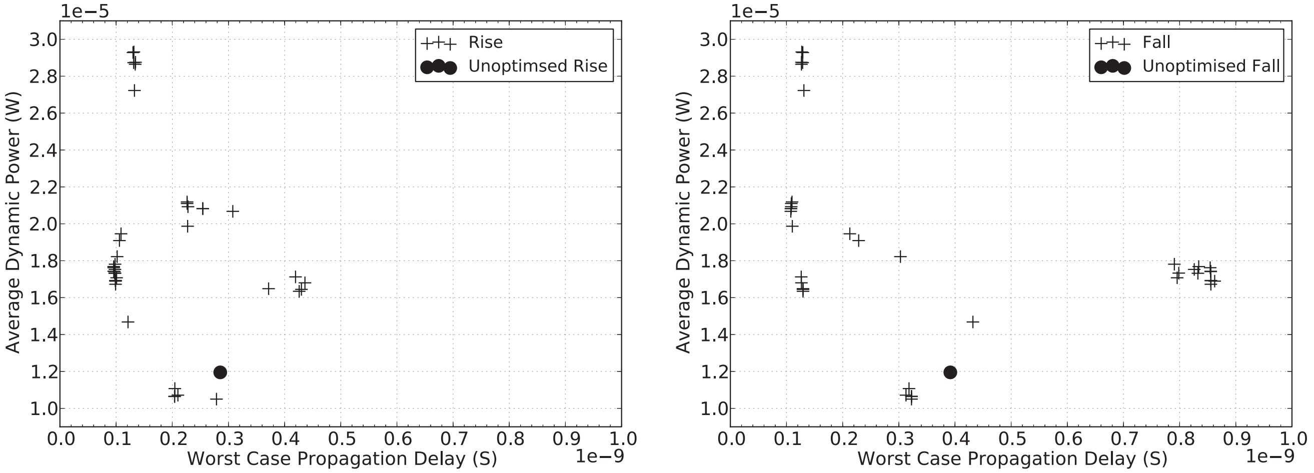
A "virtual physical instance" for each circuit, which is seen to be greatly affected by stochastic variability, is optimized for dynamic power and propagation delay by exploiting the transistor level configuration options of the PAnDA fabric, i.e., varying the widths of the CTs by altering the configuration. Optimization of the PAnDA fabric for each circuit is performed using ngenics' MOTIVATED technology, which combines multiobjective bioinspired algorithms with a massively parallel SPICE simulation engine, which is compatible with GSS' RandomSpice [28].

All SPICE simulations during the statistical variability analysis and physical instance optimization were performed on a 48-core Intel core 2 cluster running at 2.83 GHz with 96-GB RAM. The total runtime for the optimization of each physical instance was approximately 4-5 days.

7.1 ISCAS C17 Benchmark

The ISCAS C17 benchmark circuit comprises of six NAND gates, which are implemented via six CCABs in the PAnDA architecture, which are configured accordingly. The initial transistor sizing of the design corresponds to typical values found in the 40-nm standard cell library used for designing the PAnDA prototype chip to provide a fair and realistic comparison with the resulting circuits after optimization.

The results of the statistical simulations performed using RandomSpice of the ISCAS C17 benchmark implemented on the PAnDA architecture can be seen in Fig. 11. Each point in the figure represents a physical instance of the PAnDA architecture implementing the ISCAS C17 benchmark using an identical configuration. From the figure, it is possible to see how stochastic variability has effected the rise and fall propagation delay and dynamic power of 100 virtual instances of the PAnDA architecture. The spread of each performance measure is pushing the bounds of 3σ and in the case of the outlier in Fig. 11b, the worst case fall propagation delay is just beyond 6σ away from the mean. Such deviations from the mean can significantly impact the yield of such designs, especially if they are part of a timing or power critical circuit. This highlights the potential for the PAnDA architecture to alleviate such issue by reconfiguration of the CTs. To demonstrate this, a virtual instance that was seen to be significantly affected by stochastic variability



(a)

(b)

Fig. 12. The Pareto fronts for the optimized CT configurations of the selected physical instance for the ISCAS C17 Benchmark. The results are shown in terms of rise (a) and fall (b) worst-case propagation delay and average dynamic power consumption with respect to the unoptimized physical instance. Note the slight change in the scale of the axes compared with Fig. 11.

was chosen for optimization and is also highlighted in Fig. 11. MOTIVATED is then run on this virtual physical instance for 200 generations, optimizing the design for propagation delay and dynamic power consumption, and the results are shown in Fig. 12.

As can be seen from the figure, the resulting population contains a Pareto front of solutions that feature significantly faster propagation delay at the expense of dynamic power when compared with the selected virtual instance. There is generally a tradeoff between reducing power consumption and increasing speed when optimizing performance of a design. This is also highlighted by the shape of the Pareto fronts, shown in Fig. 12, resulting from the multiobjective optimization. However, solutions exist toward the tail of the Pareto front, where there is a more moderate improvement in propagation delay and a slight reduction in dynamic power compared with the selected virtual instance. Comparing one of these solutions to the unoptimized physical instance it can be seen that there is an improvement in rise and fall propagation delay by 29 and 18 percent, respectively, and an improvement in dynamic power by 11 percent. Considering the spread of propagation delay and dynamic power due to the effects of stochastic variability in Fig. 11, if this optimized configuration for the CT widths is used for this physical instance, then the position of the physical instance in the scatter cloud would move from the top of the cloud (as highlighted in Fig. 11) toward the center of the cloud. This implies that manipulating the widths of the CTs within the PAnDA fabric can help overcome the affects of stochastic variability postfabrication on an instance-by-instance basis and bring the chip back toward the target specification. Once the fabricated PAnDA chips have arrived, this investigation will be extended to verify these results and to look at the impact of optimization on a greater number of physical instances.

7.2 2-Bit Multiplier

The 2-bit multiplier benchmark circuit consists of six AND gates, two NAND gates, and two OR-AND gates; hence, 10 CCABs are used to implement this design. As in the ISCAS C17 case study, initial values of the transistor sizes of the logic gates corresponds to those found in the 40-nm standard cell library used for designing the PAnDA chip. Again, this provides a fair and realistic comparison between the selected physical instance (defined by the initial transistor sizing) and the population of optimized solutions. The results of the statistical simulations of the 2-bit multiplier benchmark using RandomSpice can be seen in Fig. 13, which shows how stochastic variability has affected the propagation delay and dynamic power. The virtual instance that was seen to be greatly affected by stochastic variability and was chosen for optimization is also highlighted. MOTIVATED is once again run on the virtual physical instance for 200 generations and the resulting Pareto fronts are shown in Fig. 14.

As in the previous section for the ISCASC17 benchmark, the figure shows the resulting population containing a Pareto front of solutions that feature significantly faster propagation delay at the expense of dynamic power when compared with the selected physical instance. However, unlike the previous case study, no solution was found after 200 generations that showed a reduction in dynamic power consumption when compared with the selected physical instance. This is most likely attributed to the fact that the initial CT width configuration of the physical instance is set at the lower end of the CT width range, which means that when the CT is reconfigured it is more likely to choose a width that it larger than that of the initial configuration and, therefore, consumes more power when the CTs are switching. The effect could also be exacerbated by the stochastic variations present in the transistors of the CTs in this physical instance that are not activated in the initial configuration. In the future, further optimization runs will be conducted to see if the results presented in this paper are conclusive.

Despite the increase in dynamic power, it is still possible to extract a configuration from the Pareto front with improved propagation delay, which would result in bringing the timing of this physical instance for the 2-bit multiplier

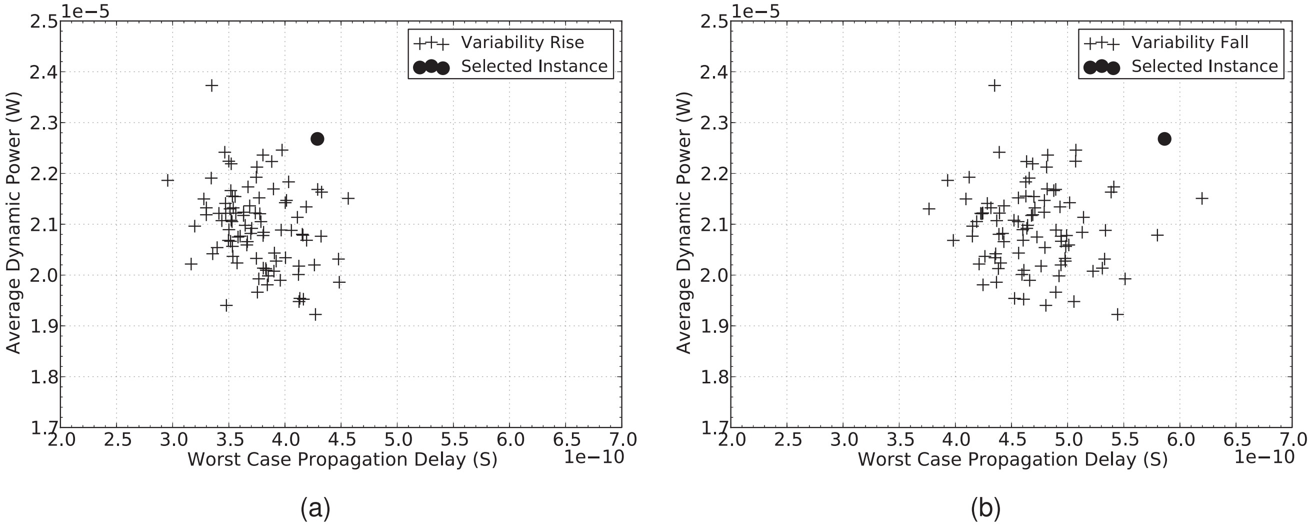


Fig. 13. The effect of stochastic variability upon 100 physical instances of the PAnDA architecture configured to perform the 2-bit multiplier Benchmark. The results show both the worst-case rise (a) and fall (b) propagation delay with respect to average dynamic power consumption. The physical instance selected for optimization is highlighted.

problem back toward the center of the stochastic variability scatter cloud (shown in Fig. 13) and once again enabling similar functionality to that of the target specification, albeit at higher power. Therefore, it is possible to say that once again this method of postfabrication optimization using the novel features of the PAnDA architecture could also improve the yield of a potentially broken device.

8 Conclusion and Future Work

This paper introduces a novel FPGA architecture, namely the PAnDA architecture. PAnDA is a hierarchical architecture, which consists of CTs, CABs, CLBs, and interconnect. At the highest CLB and interconnect level, PAnDA is similar to current commercial FPGA architectures. However, at lower levels it is the presence of CTs, CABs, and their programmability, which are unique to PAnDA. These additional levels of granularity provide access to a lower level of electronic design, i.e., the transistor (CT) and function (CAB) level, which enables the properties of the architecture at the analog level as well as at the digital level (CLBs) to be changed.

It is intended that the PAnDA architecture will close the gap between the analog design of standard cells and the design of reconfigurable digital systems based on standard cell libraries, by providing a design platform that is reconfigurable on both the analog and digital levels. The focus is to configure PAnDA with digital designs and optimize them in multiple stages at runtime, by manipulating their properties and improving the intrinsic variability of parts of the circuit by changing device sizes of the underlying analog layers. This is a novel approach to synthesizing and optimizing designs on programmable logic devices, which is not possible with any currently existing commercial FPGA.

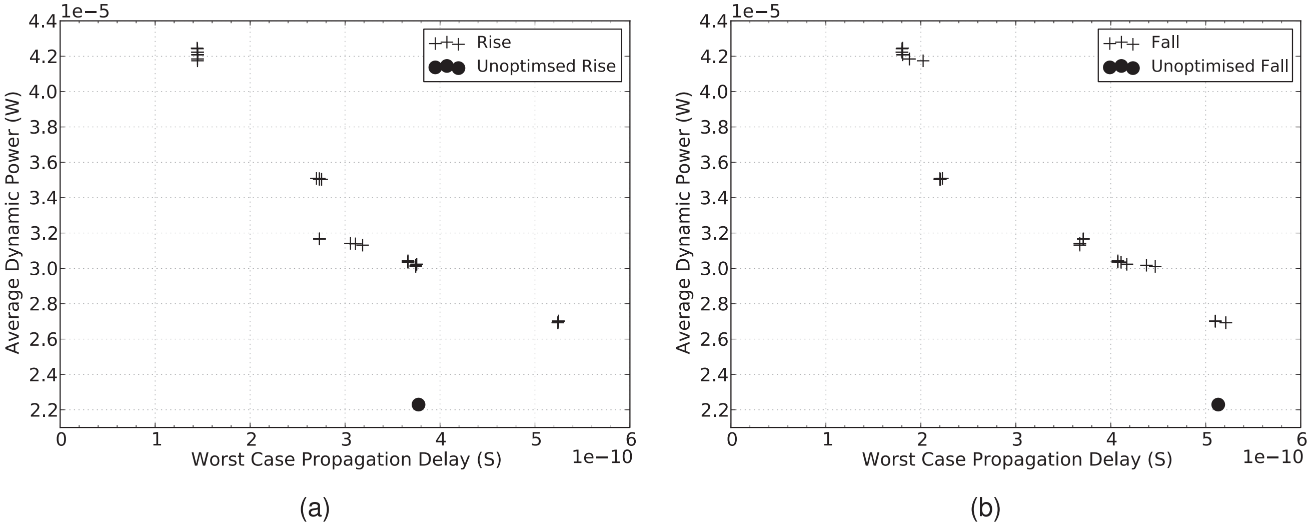


Fig. 14. The Pareto fronts for the optimized CT configurations of the selected physical instance for the 2-bit multiplier Benchmark. The results are shown in terms of rise (a) and fall (b) worst-case propagation delay and average dynamic power consumption with respect to the unoptimized physical instance. Note the slight change in the scale of the axes compared with Fig. 13.

As a result of stochastic variability both performance and reliability of electronic designs decreases, which leads to a lower production yield. It has been shown in this paper that the PAnDA architecture can successfully recover performance of a design with respect to low power consumption and even significantly improve performance with respect to speed (shorter propagation delay) when fabricated in deep submicron processes that suffer from the affects of intrinsic, stochastic variability. Similar affects caused by faults can likewise be addressed by the architecture.

This has been successfully tested in two real-world case studies, the ISCAS C17 benchmark circuit and a 2-bit multiplier using a simulation model of the PAnDA architecture, of which a prototype chip is currently being fabricated and is expected to be available in early 2013. In the case of the ISCAS C17 benchmark, speed could be improved and low power consumption could be recovered when compared with the unoptimized physical instance. Whereas only speed could be improved at the expense of power for the 2-bit multiplier. In both cases, this would correspond to improved yield figures in the case of fabrication with respect to meeting timing specification.

Furthermore, the compatibility with commercial FPGAs on the CLB level, together with the additional configuration features on CAB and CT level and the possibility of applying postmapping design optimization techniques on those levels indicates significant potential of PAnDA to be a next-generation FPGA architecture. The significantly accelerated execution of hardware, as opposed to statistical SPICE simulation, will enable us to investigate the optimization of large scale digital circuits on multiple layers of abstraction using novel bioinspired approaches in future work.

The PAnDA project has a number of strands of research that are being investigated currently or in the future. First, further prototype chips are planned for fabrication over the next two years to test and analyze designs for all entities in the PAnDA architecture, in addition to alternative transistor motifs and on-chip measurement circuits. Second, once the first prototype PAnDA architecture has returned from fabrication, we plan to investigate methods of measuring the intrinsic variability across chip and creating a "variability map" of each prototype chip. This map could then be used by the mapping and optimization tools to reduce the impact of intrinsic variability on a design and/or improve the designs performance, without the need for continuous on-chip measurements. Finally, a software framework is being implemented for the PAnDA architecture that will interface with other commercial tools to allow designs written in VHDL or verilog to be automatically synthesized, mapped, and optimized on the PAnDA chip.

Acknowledgments

This work was part of the PAnDA project that was funded by EPSRC (EP/I005838/1) and was the subject of a UK patent application (GB1119099.8). The authors would like to thank Gold Standard Simulations Ltd and ngenics Ltd for the use of RandomSpice and MOTIVATED.

References

[1] G.E. Moore, "Cramming More Components onto Integrated Circuits," *Electronics*, vol. 38, pp. 114-117, 1965.

[2] A. Asenov, "Variability in the Next Generation CMOS Technologies and Impact on Design," *Proc. First Int'l Conf. CMOS Variability*, 2007.

[3] G. Declerck, "A Look into the Future of Nanoelectronics," *Proc. Symp. VLSI Technology Digest of Technical Papers*, pp. 6-10, 2005.

[4] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter Variations and Impact on Circuits and Microarchitecture," *Proc. 40th Ann. Design Automation Conf. (DAC)*, pp. 338-342, 2003.

[5] J.A. Walker, J.A. Hilder, D. Reid, A. Asenov, S. Roy, C. Millar, and A.M. Tyrrell, "The Evolution of Standard Cell Libraries for Future Technology Nodes," *Genetic Programming and Evolvable Machines*, vol. 12, no. 3, pp. 235-256, Apr. 2011.

[6] A. Asenov, "Statistical Nano CMOS Variability and Its Impact on SRAM," *Extreme Statistics in Nanoscale Memory Design*, pp. 17-50, 2010.

[7] G. Gielen, E. Maricau, and P. De Wit, "Analog Circuit Reliability in Sub-32 Nanometer CMOS: Analysis and Mitigation," http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=5763239&contentType=Conference+Publications, 2013.

[8] M. Murakawa, T. Adachi, Y. Niino, Y. Kasai, E. Takahashi, K. Takasuka, and T. Higuchi, "An AI-Calibrated IF Filter: A Yield Enhancement Method with Area and Power Dissipation Reductions," *IEEE J. Solid-State Circuits*, vol. 38, no. 3, pp. 495-502, http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1183858, Mar. 2003.

[9] A. Stoica, R. Zebulum, and D. Keymeulen, "Progress and Challenges in Building Evolvable Devices," *Proc. Third NASA/DoD Workshop Evolvable Hardware*, p. 33, http://dl.acm.org/citation.cfm?id=517089.871975, July 2001.

[10] J. Langeheine, M. Trefzer, J. Schemmel, and K. Meier, "Intrinsic Evolution of Digital-to-Analog Converters Using a CMOS FPTA Chip," *Proc. NASA/DoD Conf. Evolvable Hardware*, pp. 18-25, June 2004.

[11] B. Nikolic and L. teck Pang, "Measurements and Analysis of Process Variability in 90nm CMOS," *Proc. Eighth Int'l Conf. Solid-State and Integrated Circuit Technology*, pp. 505-508, 2006.

[12] A. Papanikolaou, M. Miranda, H. Wang, F. Catthoor, M. Satyakiran, P. Marchal, B. Kaczer, C. Bruynseraede, and Z. Tokei, "Reliability Issues in Deep Deep Sub-Micron Technologies: Time-Dependent Variability and Its Impact on Embedded System Design," *Proc. IFIP Int'l Conf. Very Large Scale Integration*, pp. 342-347, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber= 4107654, Oct. 2006.

[13] T. Matsunawa, H. Nosato, H. Sakanashi, M. Murakawa, E. Takahashi, T. Terasawa, T. Tanaka, O. Suga, and T. Higuchi, "Adaptive Optical Proximity Correction Using an Optimization Method," *Proc. IEEE Seventh Int'l Conf. Computer and Information Technology (CIT)*, pp. 853-860, 2007.

[14] V. Kheterpal, V. Rovner, T.G. Hersan, D. Motiani, Y. Takegawa, A.J. Strojwas, and L. Pileggi, "Design Methodology for IC Manufacturability Based on Regular Logic-Bricks," *Proc. 42nd Ann. Design Automation Conf.*, pp. 353-358, 2005.

[15] S. Nassif, K. Bernstein, D.J. Frank, A. Gattiker, W. Haensch, B.L. Ji, E. Nowak, D. Pearson, and N.J. Rohrer, "High Performance CMOS Variability in the 65nm Regime and Beyond," *Proc. IEEE Int'l Electron Devices Meeting*, pp. 569-571, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4419002, 2007.

[16] K. Takeuchi, T. Fukai, T. Tsunomura, A.T. Putra, A. Nishida, S. Kamohara, and T. Hiramoto, "Understanding Random Threshold Voltage Fluctuation by Comparing Multiple Fabs and Technologies," *Proc. IEEE Int'l Electron Devices Meeting*, pp. 467-470, http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4418975, 2007.

[17] A. Asenov, S. Kaya, and J.H. Davies, "Intrinsic Threshold Voltage Fluctuations in Decanano MOSFETs due to Local Oxide Thickness Variations," *IEEE Trans. Electron Devices*, vol. 49, no. 1, pp. 112-119, Jan. 2002.

[18] J.W. Tschanz, J.T. Kao, S.G. Narendra, R. Nair, D.A. Antoniadis, A.P. Chandrakasan, and V. De, "Adaptive Body Bias for Reducing Impacts of Die-to-Die and within-Die Parameter Variations on Microprocessor Frequency and Leakage," *IEEE J. Solid-State Circuits*, vol. 37, no. 11, pp. 1396-1402, Nov. 2002.

[19] J.B. Bernstein, M. Gurfinkela, X. Lia, J. Waltersa, Y. Shapiraa, and M. Talmora, "Electronic Circuit Reliability Modeling," *Microelectronics Reliability,* vol. 46, no. 12, pp. 1957-1979, 2006.

[20] J.E. Rubio, M. Jaraiz, I. Martin-Bragado, R. Pinacho, P. Castrillo, and J. Barbolla, "Physically Based Modelling of Damage, Amorphization and Recrystallization for Predictive Device-Size Process Simulation," *Materials Science and Eng. B,* vol. 114-115, pp. 151-155, 2004.

[21] M. Mizuno and V. De, "Design for Variability in Logic, Memory and Microprocessor," *Proc. VLSI Circuits,* 2007.

[22] A. Stoica, D. Keymeulen, R.S. Zebulum, A. Thakoor, T. Daud, G. Klimeck, Y. Jin, R. Tawel, and V. Duong, "Evolution of Analog Circuits on Field Programmable Transistor Arrays," *Proc. Second NASA/DOD Workshop Evolvable Hardware,* pp. 99-108, July 2000.

[23] J. Langeheine, J. Becker, S. Fölling, K. Meier, and J. Schemmel, "A CMOS FPTA Chip for Intrinsic Hardware Evolution of Analog Electronic Circuits," *Proc. Third NASA/DOD Workshop Evolvable Hardware,* pp. 172-175, July 2001.

[24] M.A. Trefzer, "Evolution of Transistor Circuits," PhD dissertation, Rupertus Carola Univ. of Heidelberg, Seminarstrasse 2, 69120 Heidelberg, Dec. 2006.

[25] J.A. Walker, R. Sinnott, G. Stewart, J.A. Hilder, and A.M. Tyrrell, "Optimizing Electronic Standard Cell Libraries for Variability Tolerance through the Nano-CMOS Grid," *Philosophical Transactions, Series A, Math., Physical, and Eng. Sciences,* vol. 368, no. 1925, pp. 3967-3981, http://rsta.royalsocietypublishing.org/cgi/content/abstract/368/1925/3967, Aug. 2010.

[26] M.A. Trefzer, J.A. Walker, and A.M. Tyrrell, "A Programmable Analog and Digital Array for Bio-Inspired Electronic Design Optimization at Nano-Scale Silicon Technology Nodes," *Proc. IEEE Asilomar Conf. Signals, Systems, and Computers,* Nov. 2011.

[27] Gold Standard Simulations Ltd (GSS), "RandomSpice," http://www.goldstandardsimulations.com/services/circuit-simulation/random-spice/, 2010.

[28] ngenics Ltd, "MOTIVATED," http://www.ngenics.com/services, 2012.

James Alfred Walker (M'10) received the BSc degree in mathematics and computer science and the MSc degree in advanced computer science from the University of Birmingham in 2002 and 2003, respectively, and the PhD degree in electronic engineering from the University of York in 2007, for which he received the Kathleen Mary Stott Memorial Prize for Excellence in Scientific Research. He is a research fellow in the Department of Electronics at the University of York on the EPSRC funded "PAnDA: Programmable Analogue and Digital Array" project. He was previously a research associate on the EPSRC funded e-science pilot project "Meeting the Design Challenges of Next Generation Nano-CMOS Electronics" in the same department. His primary research interests include bioinspired algorithms, evolvable hardware and electronic design automation, focusing on the areas of analog and digital synthesis, design optimization, reconfigurable architectures, fault tolerance, and variation-aware design. He is an author of more than 30 peer-reviewed conference and journal publications and has attracted funds in excess of £1.4M. He is a member of the IEEE.

Martin A. Trefzer (M'10-SM'13) studied physics from the University of Heidelberg, Germany, and at the Technische Universität Berlin, Germany, from where he received a first class honors degree. He received the PhD degree in physics from the Kirchhoff-Institute for Physics, University of Heidelberg, Germany, where he was working on the evolution of transistor circuits using a custom designed CMOS FPTA. After the PhD degree, he started working on artificial developmental hardware systems as a research associate at the Department of Electronics, University of York, United Kingdom. He is currently a lecturer in the Department of Electronics, University of York, United Kingdom. His current research interests include nanoscale electronic design optimization using bioinspired approaches on reconfigurable architectures, with a special emphasis on considering the physical effects of intrinsic, stochastic variability when shrinking device sizes to atomistic scales. In recent years, he developed techniques aimed at analog and digital electronic design optimization using bioinspired approaches in conjunction with multiobjective optimization on a number of custom, reconfigurable architectures. He is also pursuing the development of novel architectures and techniques to tackle scalability, adaptivity, and robustness of electronic systems. He has published numerous papers in these areas, and has attracted funds in excess of £ 1.4 M. He is a senior member of the IEEE and a member of the DPG.

Simon J. Bale (S'05-M'12) received the MEng and PhD degrees from the University of York, United Kingdom in 2004 and 2012, respectively. He is currently employed as a research associate in the Department of Electronics at the University of York on the EPSRC funded PAnDA: Programmable Analogue and Digital Array project. In the evolutionary computing area, his research interests include: microelectronic design optimization using bioinspired methods and techniques. In the RF/microwave area, his research interests include: high Q microwave resonators, ultralow-phase noise oscillators, and low-noise measurement techniques. He is a member of the IEEE.

Andy M. Tyrrell (SM'96) received the first class honors degree in 1982 and the PhD degree in 1985 (Aston University), both in electrical and electronic engineering. He joined the Department of Electronics at the University of York in April 1990, he was promoted to the chair of digital electronics in 1998. Previous to that he was a senior lecturer at Coventry Polytechnic. Between August 1987 and August 1988, he was a visiting research fellow at Ecole Polytechnic Lausanne Switzerland, where he was researching into the evaluation and performance of multiprocessor systems. From September 1973 to September 1979, he worked for STC at Paignton Devon, on the design and development of high-frequency devices. His primary research interests are in the design of biologically inspired architectures, artificial immune systems, evolvable hardware, FPGA system design, parallel systems, fault tolerant design, and real-time systems. In particular, over the last 15 years his research group, at York, have concentrated on bioinspired systems. This work has included the creation of embryonic processing array, intrinsic evolvable hardware systems and the immunotronics hardware architecture. He is the head of the Intelligent Systems Research Group at York and was the head of the department between 2000 and 2007. He has published more than 260 papers in these areas, and has attracted funds in excess of £ 6.5M. He is a senior member of the IEEE and a fellow of the IET.