Low-Energy Embedded FPGA Structures

Eric Kusse
EECS Department
University of California at Berkeley
Berkeley, CA. USA
(now with Intel Corp, Hillsboro OR.)
ekusse@ichips.intel.com

Jan Rabaey
EECS Department
University of California at Berkeley
Berkeley, CA. USA
jan@eecs.berkeley.edu

1. ABSTRACT

This paper introduces an energy-efficient FPGA module, intended for embedded implementations. The main features of the proposed cell include a rich local-interconnect network, which drastically reduces the energy dissipated in the wiring, and a dual-voltage scheme that allows pass-transistor networks to operate at low-voltages yet maintain decent performance. Simulations on a benchmark set demonstrate that the proposed module succeeds in its goal of reducing energy consumption by an order of magnitude over existing implementations.

1.1 Keywords

FPGAs, Low Energy, Dual Voltage, Pass-transistors, Power, Embedded, Low Swing, Interconnect Network.

2. INTRODUCTION

With the trend towards integration of complete system functionality on a single die (*systems-on-a-chip*), the need has arisen for the combination of heterogeneous programmable architectures on a chip [5]. A variety of companies [3], [4] and academic research groups are currently striving to integrate traditional micro- and/or DSP processor with large macromodules of embedded PGA (programmable gate arrays). The intended function of these programmable logic modules (PLMs) varies from user-modifiable periphery to the implementation of high-performance signal-processing functions that are not efficiently implemented on traditional programmable architectures. While this approach can have a major impact on the performance and flexibility of these future systems-on-a-chip, it is severely hampered by the energy-inefficiency of today's FPGA implementations. Energy (and/or power) consumption has become a major concern in IC design and will become even more so with higher levels of integration. This is definitely the case in the systems-on-a-chip world where most applications are embedded such as cellular telephony. Thus, lowering energy dissipation is a priority.

PGAs, as can be expected from their general purpose architecture, pay a high price in energy consumption. This situation is further aggravated by the fact that today's designs have been solely optimized for performance and density, and energy efficiency has only been barely considered. This is illustrated in Table 1, which shows energy metrics for a variety of designs. A comparison between the XC4003A [7] and a static CMOS full adder implementation show a 100 x difference in energy consumption for an 8-bit adder. Entire processors consume less power than a design consisting of 50 XC4000XL logic functions. Clearly, power and energy consumption in PGAs must be examined and reduced if their utility is to be preserved in the system-on-a-chip scenario.

Design Example	Vdd	Energy
Xilinx XC4003A 8-bit Adder (measured)	5v	4.2mW/MHz
Xilinx XC4000XL Series	3.3v	92uW/MHz/ Logic Function Output
.8um Gate Array	unspecified	7.5uW/gate/ MHz
Static CMOS Full Adder	3.3v	5.5uW/MHz
54x54 Multiplier	2.5v	2.23mW/MHz
DSP Processor	1v	0.21mW/MHz
StrongArm Microprocessor	1.5v	2.1mW/MHz
Alpha Microprocessor	2v	60mW/MHz

Table 1: Energy Metrics for Various Designs

This paper examines the dominant sources of power dissipation in current PGA incarnations. Based on the obtained information, we propose an architecture and implementation of an energy-efficient FPGA cell and module. The paper is then concluded with an analysis of the effectiveness of the proposed cell (and module).

3. Energy Dissipation in FPGAs

When attempting to design energy-efficient PGA structures, an essential first step is to gain some insight into PGA power consumption by opening the black box. However, without the luxury of detailed schematics or layout of a complete

ISLPED98, Monterey, CA, USA

PGA, the only means to study an array was by physical lab measurements. A systematic procedure was developed to obtain a detailed picture of the internal capacitances of such a chip. These capacitance values were then combined with architectural knowledge to form some crucial insights about where power goes in PGAs.

A Xilinx XC4003A became the analysis target because of the existence of a test board and the necessary Xilinx design software. The XC4003A was fabricated in a 0.6 µm, 2-layer metal process. Although the XC4000 series has been improved upon in recent years, the underlying architectural design has remained consistent allowing useful conclusions still to be drawn. The measured results are shown in Table 2. Observe that each measurement was performed in a relative fashion. That is to say, one measurement was recorded without the circuit component toggling and one with the component toggling. Thus, all extraneous sources of current draw (power) are eliminated from the desired measurement. In addition, the accuracy of measurements was improved by enabling several of the same components at one time to give a larger current differential.

Component	Energy (mW/ MHz=nJ)	Estimated Cap. (pF)
CLB Function Generator	0.025	1.10
I/O Input Path (I1,I2)	0.062,0.140	2.5,5.6
CLB Input Interface	0.040-0.050	1.95-2.4
CLB Output Interface	0.041	1.64
10 CLB Longline (Horiz,Vert)	0.054,0.088	2.7,4.4
5 CLB Longline (Horiz,Vert)	0.022,0.040	1.1,2.2
Double Line	0.06-0.107	3-5.35
Single Line	0.048-0.088	2.4-4.4
Clock Connection	0.030	1.5
Clock Column Distribution Wire	0.128	6.4
Carry Chain	0.050	2.5

Table 2: Measured Component Energies and Capacitances

The most astounding fact about the numbers is their order of magnitude. In conventional VLSI CMOS designs, the capacitances are typically in the range of tens to hundreds of fFs. Only on long, high fanin/fanout busses would one be likely to encounter capacitive loads in the picofarad range. However, in the case of the PGA, a single wire spanning one CLB (combinational logic block[1]) pitch has a load of 2-4 pFs and the other types of interconnect fall in this range as well. One might have anticipated a high value for interconnect capacitances given that interconnect delays are fairly substantial in these devices, but picofarads is quite a surprise.

While this data is worthwhile, the only relevant question is the distribution of capacitance and energy for real mappings and data activities. Therefore, a power-prediction tool was developed. The tool combines the baseline capacitance data obtained from the detailed lab measurements and the existing Xilinx software flow. By forming a linear combination of all the characterized components used in a design, the total capacitance and dynamic power consumption could be calculated. A first order estimate of the activities of various nets was then included to achieve the proper weighting of terms. A set of Perl programs were written to perform these various tasks.

The combined results from running the analysis tools on a set of 36 mapped designs are displayed in the series of graphs displayed below. The 36 designs used for the data are composed of 27 Xilinx macros for common functions and 9 larger designs. The set of designs gave a good cross-section of the probable make-up of a PGA design. In all cases, the designs were analyzed without any outputs driving chip pins. This insured that the power contribution from driving board capacitances did not get factored into the power profile of the internal PGA circuitry.

The results of the experiments are shown in the figures below. Clearly apparent from these charts is the overwhelming dominance of interconnect to a design's power consumption. In all cases, at least 65% of a design's power is dissipated in the collection of interconnect resources and logic cell interface circuitry that a design utilizes. The category of interconnect resources encompasses single length, double length, carry chains, and longlines. Interface circuitry includes the input multiplexers used to select CLB inputs from the wiring tracks (an essential part of the interconnect structure) and the output buffers used to drive signals onto the interconnect fabric. Astoundingly, the logic circuitry that actually implements the desired functionality consumes a minimal amount of power. A good way of thinking about the situation is to treat

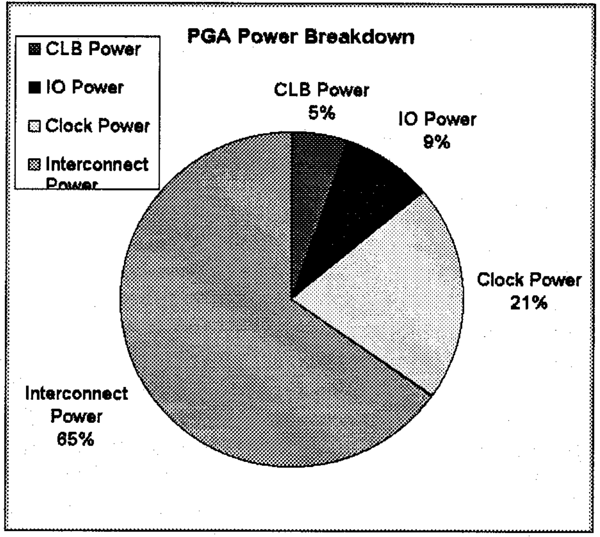

Figure 1. Power Breakdown for Xilinx XC4003A

[1] A combinational logic block for the Xilinx XC4000 series mainly consisted of two 4 input look-up tables (LUTs) and the associated interface circuitry; this represents one tile in the overall array.

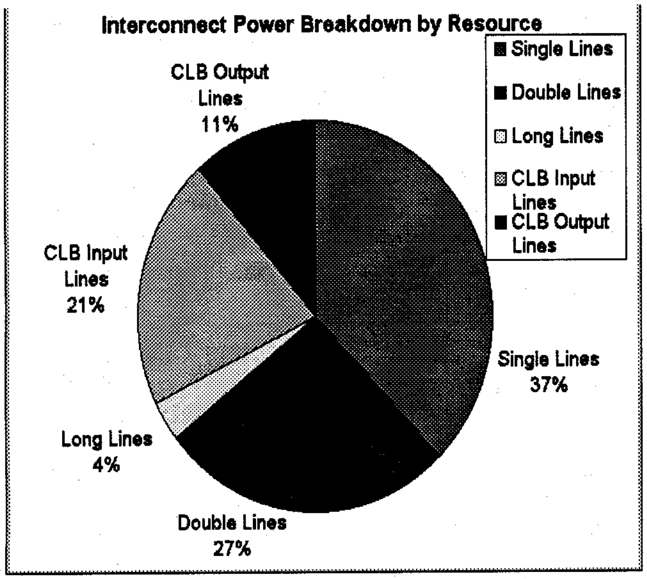

Figure 2. Interconnect Power Breakdown by Resource.

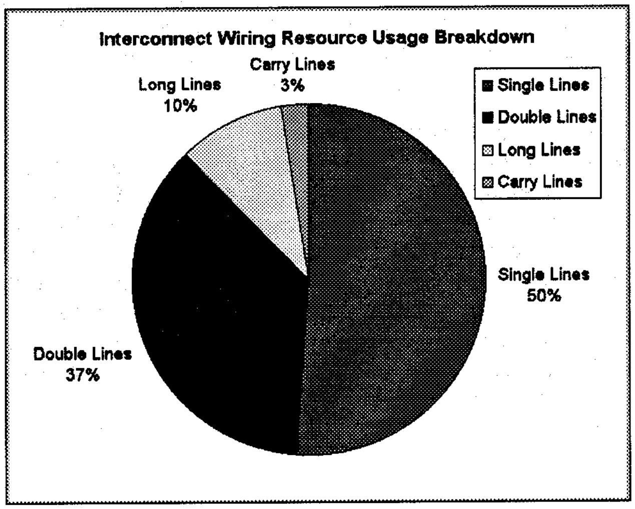

Figure 3. Interconnect Wiring Resource Usage Breakdown.

the CLBs as islands in a sea of very expensive interconnect resources. As soon as a signal leaves an island, it immediately drowns in the capacitance discontinuity caused by the general purpose routing network. Therefore, any attempt to reduce PGA power must focus on how to minimize interconnect power.

4. Low-Energy FPGA Modules

Attempting to design low-energy FPGA modules requires an effort at all levels of design abstraction. Two main targets of optimization are apparent: 1) reduce the impact of the interconnect wires; 2) reduce the operation voltage. Achieving either of these goals (while still maintaining decent performance) requires optimization at the architecture, logic, and circuit abstraction levels. In the coming paragraphs, we present a cell structure that addresses the interconnect problem. Circuit topologies that help reducing the supply voltage are discussed thereafter.

4.1 Module Architecture

The basic cell architecture defines a PGA's map-ability and its internal capacitances. Once the logic cell's functionality has been defined, the surrounding interconnect network can be constructed based on the mapping properties of the cell. Determination of the cell-interconnect interfacing resources and the amount of flexibility to support were developed from several mapping experiments and intuition built up from studying PGAs. Using the initial design as a template, a series of real-world design examples and common logic operations were hand mapped, the results of which were used to tune the architecture. Accumulators, counters, shift registers, and comparators were some of the logic building blocks that were analyzed. In addition, a correlator and the add-compare-select block of a Viterbi decoder were mapped to examine the architecture's viability given larger functional blocks.

The general motivation for the architecture focused on *preserving the locality and structure* present in most designs. By combining the connectivity properties of designs with an underlying PGA structure which facilitated those required mapping characteristics, low capacitance mappings could be achieved across a broad range of designs. The resulting structure aims at striking a balance between datapath style design requirements and random logic mappings. In doing so, highly regular, dense mappings can be produced yielding very efficient cell utilization. Furthermore, the trade-off between flexibility and resource capacitance is managed by providing a highly useful, **low capacitance, neighbor-to-neighbor network** combined with a **datapath-oriented local bus structure**. Finally, a hierarchy of stacked grids overlays the lower levels to allow long distance routing in a similar fashion as traditional over-the-cell routing in IC datapaths. Careful construction of these interconnect resources creates a gradual progression from low capacitance local wiring, to more flexible higher capacitance routing, and enables designs to exhibit much lower average net capacitances.

The resulting cell and module architecture is shown in Figure 4. In actuality, two logic cells are grouped together so that they share inputs although they can operate independently from each other through the direct path interconnect. Each logic cell is represented by a black box since the circuit details inside are not important. The only important information from an architectural standpoint is the ability of the logic cells to implement any function of 3 variables with complete permutability of the inputs A,B, and C. A 3-input look-up table or LUT was chosen as the basic building block as it effectively implements the fundamental operation of a BDD (Binary Decision Diagram), the methodology of choice in logic synthesis today. Surrounding the cells are the input interface multiplexers, which connect to the various interconnect resources.

The most important feature of our architecture is the level-0 interconnect layer, which serves as the primary resource for

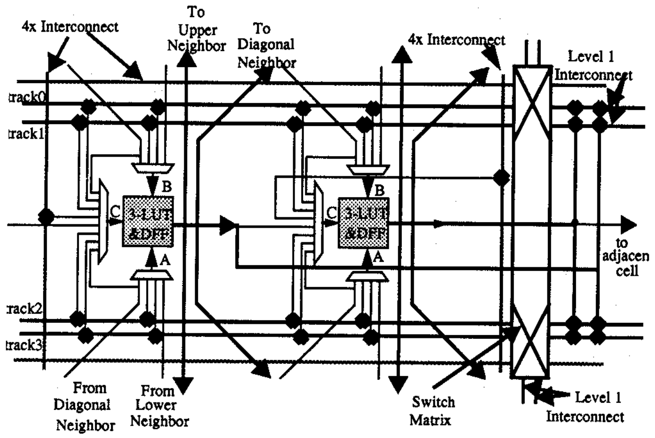

Figure 4. Basic Logic Cell-Pair Tile

localized communication. The segments are designed as point-to-point connections allowing them to have lower capacitance than the other types of interconnect. An abstract representation of the level-0 interconnect is shown in Figure 5. Using the direct connects, a logic cell can efficiently support fanouts as high as 5 to neighboring cells. Although designs commonly see fanouts of only 1 or 2, the architecture was made symmetrical about the horizontal to facilitate the mapping task.While other architectures have been proposed using extensive local interconnect (called cellular FPGAs [1] [6]), none has provided the richness and combined it with the hierarchical approach advocated here.

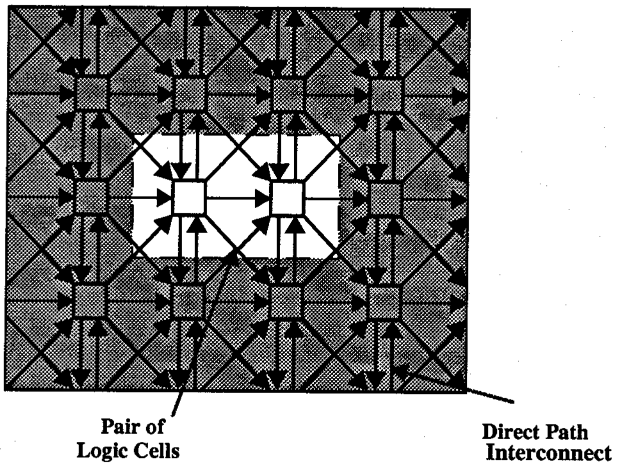

Figure 5. Abstract view on Level-0 interconnect.

One can see that most of the direct interconnect flows either vertically or diagonally. The reason for this is that the level 1 interconnect (Figure 6) is more aptly suited for horizontal routing along the dataflow direction. However, by including the vertical and diagonal connections, many paths avoid using the more general purpose interconnect layer allowing the general purpose layer of interconnect to be greatly simplified. In addition, many mapping targets (adders, comparators, etc.) require data to be passed across the width of a datapath from bit-slice to bit-slice, this task is extremely well suited to the level-0 interconnect. In summary, the paths offered by the direct connection network provide a valuable low-cost routing resource which should be leveraged for as much of the local wiring demands as possible.

On top of the level-0 and level-1 interconnect sits a further hierarchy of grids. Additional interconnect is necessary to provide further wiring capacity and to facilitate long-distance routing. The first layer is aligned along a 4x4 block of cells. In addition to the 4x4 grid, successive levels of hierarchy can be added as needed (e.g. 16x16). In order to minimize the capacitance impact of the hierarchy on the lower routing levels, connections to lower levels are only made at the edges of the grid. Thus, the long distance routes are reserved for signals which need to skip over some logic cells before reaching their destinations. As a result, the global wires do not see fanout capacitance due to logic cell inputs, only from interfacing to other interconnect layers.

Benchmark mappings (of a Viterbi coder and a CDMA correlator) demonstrated the effectiveness of the proposed architecture, and its potential for very high utilization. The mapping diagrams for these structures were too large to include a readable version in this paper. More information on these examples can be found in [2].

4.2 Circuit Design Issues

Reducing the supply voltage is one of the most effective techniques to minimize energy dissipation. However, this method presents a tough challenge in the design of FPGAs, since most of these structures make extensive use of pass-transistor logic. The problem of supply reduction is further exacerbated in processes that do not have low-threshold devices. In these processes, lowering the supply voltage below 2 Volts — 1.5 V is our intended operating voltage — results in a dramatic loss of performance and even causes some circuits to malfunction. Even though low threshold

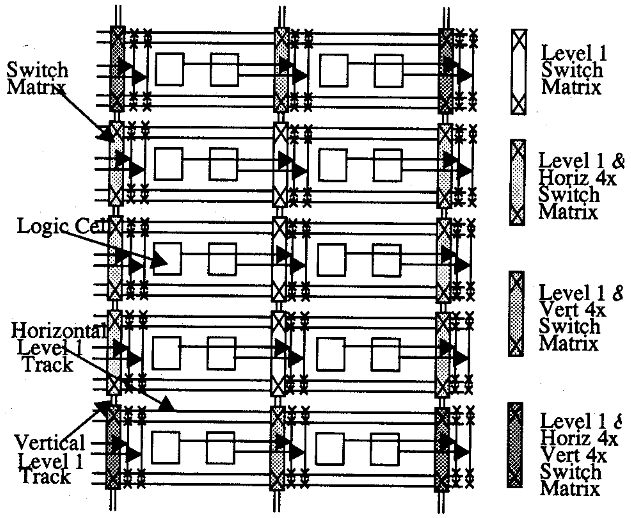
Switch Matrix
Logic Cell
Horizontal Level 1 Track
Vertical Level 1 Track
Level 1 Switch Matrix
Level 1 & Horiz 4x Switch Matrix
Level 1 & Vert 4x Switch Matrix
Level 1 & Horiz 4x Vert 4x Switch Matrix

Figure 6. Level-1 bus-oriented interconnect architecture.

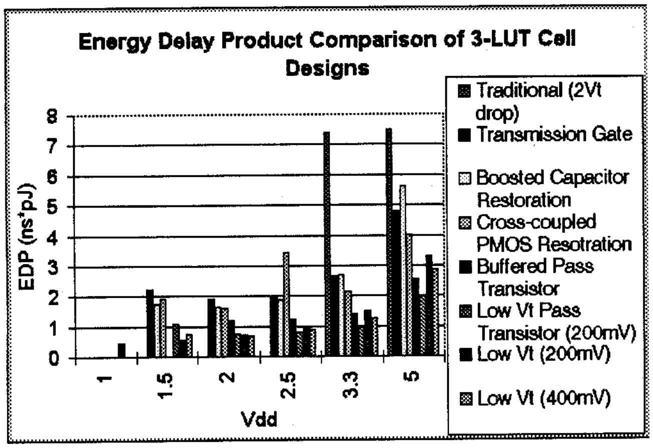

Figure 7. Simulated energy-delay product of 3-input LUT for different implementation styles.

transistors were not available in our standard 0.5 μm CMOS process, we still conducted an extensive study of all design options for the logic cell and programmable interconnect circuits. An overview of the simulated energy-delay products for a variety of cell implementations is given in Figure 7.

Based on these observations, we came to the conclusion that a 2-voltage scheme, combining signals at 2V and 1.5V was the only reasonable solution. FPGAs have the advantage that the programming signals only change at reconfiguration time, and hence have low activity. Placing these signals on a 2V supply has little impact on the energy dissipation. Yet, as these signals are often applied to the gates of the pass-transistors, increasing their voltage level has a positive impact on the performance. Figure 8 demonstrates how a judicious choice of the supply and signal levels can result in a high performance, yet low-energy pass-transistor logic in a high-threshold process.

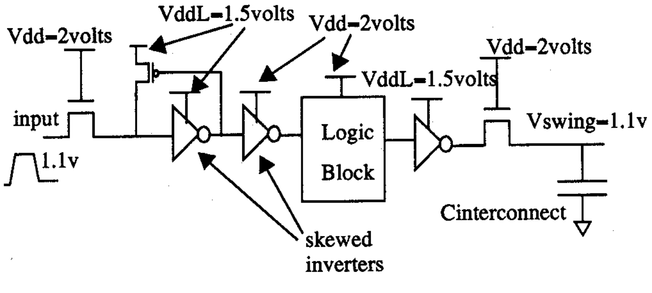

Figure 8. Typical path from interconnect through logic cell and back to interconnect (as implemented in FPGA module).

5. Results

The layout of a mini-array of 8 cells (4x2) is shown in Figure 9. From this design, the following table was extracted listing energy and extracted capacitance data for the paths in the PGA basic cell. When looking at the table, one should remember that the relationship between the energy and capacitance numbers for a given component cannot be described by a single scaling factor, instead the energies depend on the supply voltage and swing for that particular path. The number in parentheses that appears next to some values indicates the reduction that was achieved over the Xilinx XC4003 component.

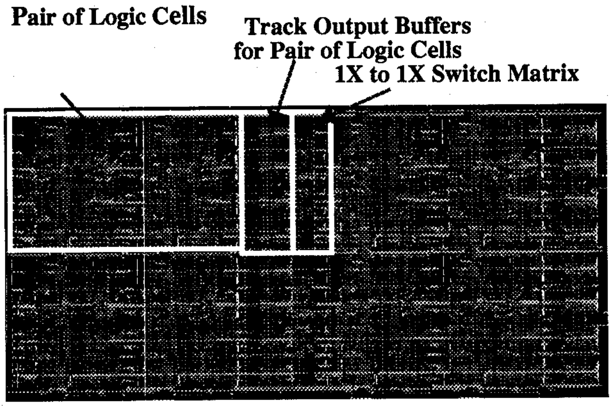

Figure 9. 2x4 Mini-Array of Logic Cells (100,000 μm^2 area).

Path	Energy (pJ)	Cap. (fF)
A or B Input	.275 (145x)	95 (20x)
C input	.34 (147x)	140 (17x)
LUT Tree (all paths toggling)	.265 (94x)	120 (9x)
LUT Intermediate Output Buffers and Fanout Node	.33 (124x)	180 (9x)
Vertical Direct Path	.115 (417x)	55 (43x)
Diagonal Direct Path	.135 (355x)	60 (40x)
Direct Horizontal Path	.14 (343x)	65 (37x)
Level 1 Horizontal Track 0 or 3 and Buffers	.42 (143x)	200 (15x)
Level 1 Horizontal Track 1 or 2 and Buffers	.435 (138x)	220 (13.6x)
Level 1 Vertical Track	.066	40
Clock Input/ Logic Cell	.13 (115x)	60 (12.5x)
Reset# Signal/ Logic Cell	.063	28
Program#/ Logic Cell	.13	32.5
Bit Line/ Logic Cell	.13	33
Word Line/ Logic Cell	.08	20

Table 3: Energy and Extracted Capacitance Data

From the data, one can see that the cell capacitances and energy have been greatly reduced. The average energy reduction for the various components was over two orders of magnitude. The substantial improvement in energy consumption comes from a combination of lower voltage and lower capacitances. The average capacitance of resources was lowered by 10x-15x. Much of the decrease in capacitance can be attributed to efficient sizing of drivers, architectural minimization of fanout, appropriate choice of switch size (1.8um) to reduce switch capacitances, and a compact layout strategy.The reduction in energy comes at a penalty: a factor of two in performance degradation (approximately) can be observed compared with respect to the equivalent, 5 V industrial design.

To study the effectiveness at the module level, the energy consumption for an 8-bit adder is analyzed. The Xilinx design consumes about 3.5nJ of energy excluding the I/O and clock components. By comparison, an 8-bit adder on the low power design is estimated to burn 50pJ of energy assuming a comparable amount of interconnect resource usage and an activity factor of 1 (An example of the cell mapping is shown in Figure 10). Thus, the low power design consumes 70 times less energy. If an average activity factor of 0.3 is used, the difference becomes greater than two orders of magnitude.

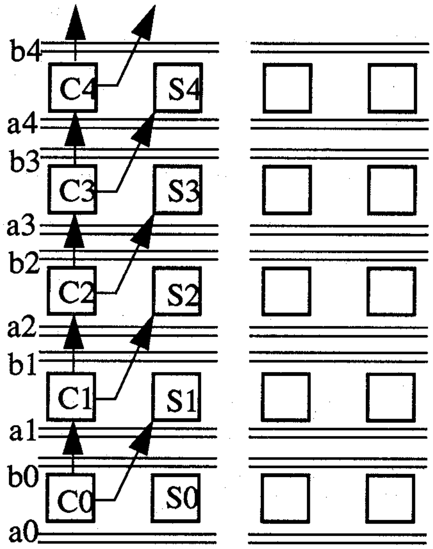

Figure 10. Adder Cell Mapping Example

Observe that several improvements have been made to the Xilinx family since the 4003A. Using some recent data [8], we can conclude that state-of-the-art designs (in a 0.35 µm CMOS technology) consume 92pJ/ logic cell or 46pJ/ logic function. By comparison, our low power design consumes 1.5pJ/ logic function without interconnect. Assuming that the average cell's interconnect needs require an equal amount of energy (an estimation based on mappings to **this** architecture), the total energy/logic cell is about 3pJ. Thus, this low power design still offers a substantial improvement in energy (15x).

6. Summary

In this paper, we presented an analysis of the sources of power dissipations in FPGA modules. We concluded that interconnect presents the large majority of energy dissipation. We have presented a logic module that addresses some of the problems posed by the interconnect, as well as a dual-voltage circuit design approach that helps to reduce the supply voltage while maintaining performance.

7. Acknowledgments

We would like to acknowledge DARPA's support for the Pleiades project (under the ACS program). Also, we want give credit to our colleagues in the project, who provided valuable inspiration and insight.We are grateful to Scott Hauck for making his designs available for study.

8. References

[1] Hauck, S.,et al., "Triptych: An FPGA Architecture with Integrated Logic and Routing", in *Advanced Research in VLSI and Parallel Systems: Proceedings of the 1992 Brown/MIT Conference*, (March 1992), 26-43.

[2] Kusse, E., "Analysis and Circuit Design for Low Power Programmable Logic Modules“, Masters Thesis UC Berkeley, http://infopad.EECS.Berkeley.EDU/research/reconfigurable/reports/ekusse/thesis.html, (December 1997).

[3] "Motorola chip to combine ColdFire, FPGA cores", http://techweb.cmp.com/eet/news/98/992news/motorola.html.

[4] National Semiconductor's Adaptive Systems on-a-Chip, http://www.national.com/appinfo/milaero/napa1000.

[5] Rabaey J.,et al., ""Heterogeneous Reconfigurable Systems", in *Proc. Sips 97*, Leicester, (Nov. 1997), 24-34

[6] Trimberger, S., "Field Programmable Gate Array Technology, Kluwer Academic Publishers, Boston Mass., 1994.

[7] Xilinx Corporation, "XC4000 Field Programmable Gate Arrays: Programmable Logic Databook", 1996.

[8] Xilinx Corporation, "Application Brief #14, A Simple Method of Estimating Power in XC4000 XL/EX/E FPGAs", 1997.