A Circuit-Software Co-Design Approach for Improving EDP in Reconfigurable Frameworks

Somnath Paul†, Subho Chatterjee*, Saibal Mukhopadhyay* and Swarup Bhunia†
*Department of ECE, Georgia Institute of Technology, Atlanta, USA
†Department of EECS, Case Western Reserve University, Cleveland, USA

Abstract—**Use of two-dimensional memory array for lookup table (LUT) based reconfigurable computing frameworks has been proposed earlier for improvement in performance and energy-delay product (EDP). In this paper, we propose an integrated solution for achieving significantly higher EDP in these frameworks by leveraging on the read-dominant memory access pattern. First, we propose to employ an asymmetric memory cell design, which provides higher read performance (∼2X) and lower read power (∼1.6X) in order to improve the overall EDP during operation. Exploiting the fact that the proposed memory cell provides better read power/performance for cells storing logic '0', next we propose a content–aware application mapping approach, which tries to maximize the logic '0' content in the LUTs. We show that the joint circuit and application mapping level optimization approach provides significant improvement in system EDP for a set of benchmark circuits.**

Index Terms - **Low Power, Memory Design, Content–aware Mapping, Reconfigurable Frameworks**

I. Introduction

Since their introduction, Field Programmable Gate Arrays (FPGAs) have become a widely popular reconfigurable computing platform for implementing digital circuits. However, the downside of the reconfigurable nature is that the design mapped on a FPGA platform operates roughly 3 times slower, occupies 10 times more area and consumes almost 2 times the power compared to an ASIC implementation at the same technology [1]. The primary reason behind such a penalty is the elaborate programmable interconnect network connecting multiple Configurable Logic Blocks (CLBs). As the process shrinks, this interconnect delay does not scale in the same manner as the logic delay. Therefore the contribution of the interconnect delay is expected to increase in future generation of FPGAs fabricated in nanometer technologies [1]. In order to minimize the area dedicated to programmable interconnects, researchers have proposed mapping larger multi-input, multi-output partitions as lookup tables (LUT) to embedded memory blocks (EMBs) inside conventional FPGA frameworks [2-4]. This not only minimizes the area to implement the logic but also improves the performance by reducing the contribution from the interconnects. We refer to these frameworks as "EMB based Heterogenous FPGA". Contrary to the purely spatial computing model of EMB based FPGAs, a time-multiplexed hardware reconfigurable framework using 2-dimensional memory array has also been explored [5-7]. Since memory array is used as the underlying reconfigurable computing fabric, we refer to these alternate frameworks as Memory Based Computing (MBC) frameworks.

We note that optimization in the memory design and subsequent changes in the mapping flow can significantly improve the Energy Delay Product (EDP) in these MBC frameworks. We also note that the memory arrays used in reconfigurable computing [2-4, 5-6], have a read-dominant access pattern, while write occurs very infrequently during the reconfiguration process. In order to leverage on this observation, we present a memory cell design that offers high read performance and improved read power compared to the conventional 6-T SRAM cell. Moreover, we note that read power for the cell is significantly higher for a stored value of logic-1 compared to logic-0. Based on this observation, we propose a novel content-aware mapping algorithm to skew the LUT contents for an application so that the LUTs will contain more logic-0s rather than logic-1s. In particular, the paper makes the following major contributions. i) It presents a novel memory cell design for memory based reconfigurable computing frameworks which offers higher read performance and lower read power compared to a conventional 6-T SRAM cell. ii) For the proposed memory design, it proposes a content–aware application mapping scheme that minimizes the read power by skewing the LUT contents to contain more logic-0s than 1s.

II. Memory Cell Design

A. An Alternate 6-T Structure

We propose an alternate cell for the read heavy conditions which can benefit us in terms of increased SNM, more tolerance to process variability and reduced power consumption. However, writing to the cell is single-ended and requires word line boost up methods thereby consuming more write power than the conventional 6-T structure.

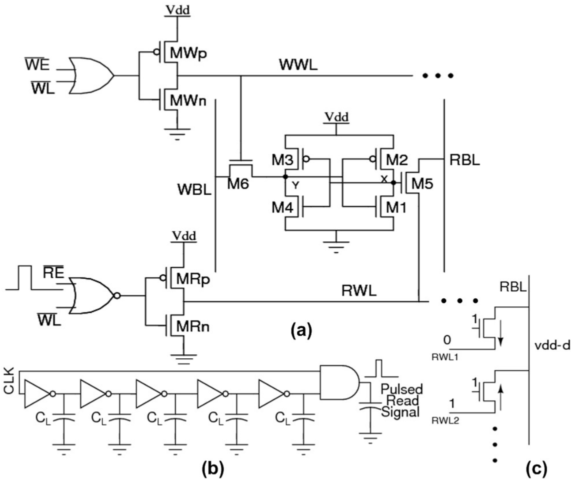

Fig. 1. a) The proposed 6-T SRAM cell; b) Pulse generation circuit for read operation; c) Pathological case showing the flow of static current from an unselected cell to the selected cell.

ICCAD'09, November 2–5, 2009, San Jose, California, USA.
Copyright 2009 ACM 978-1-60558-800-1/09/11...$10.00.

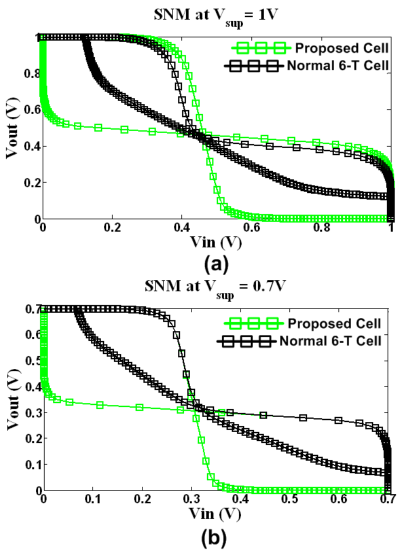

Fig. 2. a) The proposed cell offers a higher SNM compared to the conventional 6-T SRAM at nominal voltage; b) The improvement in SNM becomes more marked at lower supply voltages.

So the proposed structure's energy efficiency over the conventional case is a direct function of the read write ratio. The read and write mechanisms for this structure is quite different from the conventional 6-T structure. Following is an description of the read and write operations for such a structure:

Read: If a particular word line is to be read, the RWL is made low (RWL is derived from RE and WL as shown in Fig. 1(a)). In presence of read signal and word line selected, transistor M5 has its source set to '0'. If it stores a '0' at point X, the transistor does not conduct and the bitline does not discharge. If the transistor stores a '1' at point X, the bitline discharges through the transistor M5 and eventually though MRn. The greatest advantage with this structure is that the charge cannot enter the storage node during read discharge. Hence possibility of read disturb arising from flipping of data during read is nullified.

Write: The writes in this case are single ended. So to facilitate write we use techniques like word line boosting. This however comes at a price of higher energy consumption for writes and affects the overall energy benefit. Hence the cell is suited for cases with low percentage of writes for leveraging maximum performance.

B. Read Stability

The improved readability of the proposed structure is due to coupling of the read and writes such that the cell being read from is isolated from the bitline so that there is no issue of the cell flipping while reading. The SNM curves in Fig 2(a) illustrate the readability advantage of the cell at supply=1V. At lower supplies, the distinction with conventional 6-T SRAM is even more prominent as shown in Fig 2(b). This is due to the fact that even at smaller supply voltages charge at the node still remains decoupled for proposed cell structure.

C. Read Energy Analysis of the Cell

The word line switching energy of the proposed cell is lower as it sees the junction transistor of the read access transistors instead of the gate capacitance of two access transistors, leading to lower read energy of the proposed cell. However one has to confront a pathological case, which is illustrated in Fig. 1(c). When the RBL is pre-charged and let go, let us consider a case where row 1 is selected and let us also consider unselected cells along the same column storing '1' at the gate of the read transistor. Now as the selected cell discharges via RWL1 (shown by the arrow), the bit line voltage drops to $(V_{dd} - V_{th})$. The unselected cells start contributing bringing the RBL upto $(V_{dd} - V_{th})$. In order to minimize the static power dissipation for the unselected cells we perform a pulsed read. Fig. 1(b) shows the circuit to generate read pulse for the proposed memory cell. While choosing the pulse width we have to keep in mind the fact that it must allow the bit line to be discharged by the required amount for single ended detection (200mV or higher). Based on our objective of achieving reliable operation of the proposed cell and power savings over a conventional 6-T cell, pulse width of 150ps was selected.

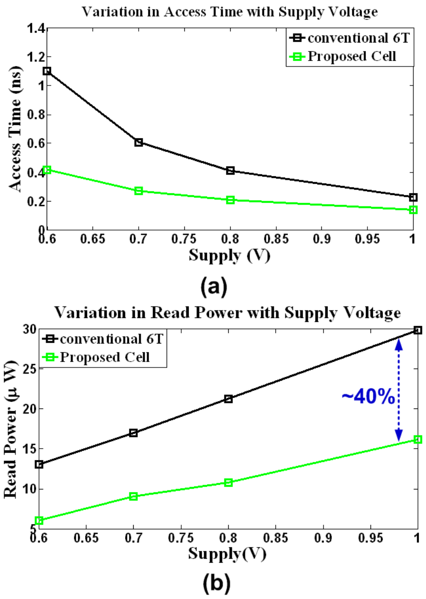

Fig. 3. a) The proposed memory cell has improved access time compared to the conventional 6-T SRAM cell. The improvement in access time is higher for lower supply voltages; b) The proposed cell also achieves a 40% reduction in the read power at nominal supply voltage.

D. Operation at Higher Frequency

The proposed cell is also a better choice for operation at higher frequencies. In the simulation, a 16X64 array of cells is considered with a 8 bit wordsize at 45nm predictive technology [8]. The proposed cell is taken so as to match the area for the conventional 6-T SRAM cell. Fig. 3 indicate that the proposed cell has 2X lower access time compared to conventional 6-T cell. It is to be noted here that reading a stored logic '0' does not require the bitline to be discharged. The power expended in reading logic '0' is therefore only the wordline power. The read power reported in Fig. 3(b) considers the worst case scenario of reading logic '1' from the proposed memory cell.

E. Writability

Write in the circuit has to be done by maintaining the selected cell at the cell supply voltage 0.6V whereas raising the voltage for the unselected cells to nominal V_{dd} value. This ensures unselected cells of selected columns have cell supply voltage but cells with the selected row have raised voltages to avoid flipping. Though a 6-T cell requires larger cell supply than this for reliable operation we consider a 6-T cell at 0.6 V for comparison purposes. The alternate cell requires additional wordline switching for the word line resulting in a write energy overhead of : $\Delta E/E = (V_{WL}/V_{cell})^2 - 1 = 1.78$. Considering that the target reconfigurable framework has a read dominated access pattern ($\sim 99.9\%$ of the operations are read), the total energy savings of this structure is given by: $\Delta E = 0.999 \times 0.4 - 0.001 \times 1.78 = 0.397$, where 0.4 denotes the 40% improvement in read energy from Fig. 3(b).

III. Content–aware Mapping for Power Reduction

A. Heuristic for Content-aware Mapping

We present a greedy heuristic for maximizing the percentage of logic ‘0’s stored in the LUTs mapped to the memory array. The partitions are first arranged in their topological order. Then beginning from partitions present in the first level, if the partition truthtable is found to contain more logic ‘1’s, then each LUT location is inverted. This is only done for LUTs which do not drive any primary output of the circuit or input to any internal state element. In order to preserve the correctness of the logic in the following levels, 0 and 1 locations for the LUTs in the fanout of the modified LUT are rearranged. This does not modify the total 0 or 1 count. We have validated the effectiveness of the proposed mapping approach using a set of standard circuits chosen from the ISCAS’85, ISCAS’89 and MCNC benchmark suites. For the benchmark circuits selected we note that the proposed mapping heuristic achieves almost *49%* increase in the percentage of logic ‘0’ count stored in the LUTs.

B. Integrated Design Flow

Fig. 4 shows an integrated design framework that combines the proposed memory cell design with the content-aware application mapping. This framework is used as our simulation platform. Input to the framework is a verilog netlist containing the hypergraph representation of the target circuit. The input netlist is partitioned into multi-input (M) multi-output (N) partitions which are mapped as LUTs in the memory arrays. We have implemented a partitioning routine based on the concept of Maximum Fanout Free Subgraph (MFFS) presented in [2]. The partitioning routine first maps the input netlist to a hypergraph representation containing M-input, 1-output LUTs using the depth optimal *flowmap* algorithm [12]. The partitions are then clustered based on maximum input cone sharing without violating partition output constraint (N). The next step in the design flow is the content–aware mapping heuristic. The final partitioned netlist is then distributed over multiple computing elements (referred to as MCBs) according to design specification of each MCB. By design specification we mean: i) The amount of memory (Mem) present in each MCB; ii) The maximum number of primary inputs (PI) to and primary outputs (PO) from each MCB in every cycle; iii) The number of temporary registers (Reg) present inside each MCB so as to hold the intermediate outputs from the partitions. The netlist of MCB elements are placed and routed using the VPR toolset [9]. Power and performance estimates of the new memory cell is then used to estimate the power and performance for the final netlist.

IV. Results

For the standard benchmark circuits considered in our simulations, $M = 12$, $N = 4$, $Mem = 2KB$, $PI = 32$, $PO = 32$ and $Reg = 24$ was found to offer significant performance benefit over a conventional FPGA framework. Power estimation for the routed netlists (both FPGA and MBC) are obtained using power-aware VPR tool [11]. Detailed spice simulations were carried out to estimate the cycle time and power for each MCB. Power and performance contribution of the memory array was estimated for both conventional 6-T SRAM and the new memory cell design proposed in this paper. Simulations were carried out for PTM 45nm models [8]. Fig. 5 shows the delay and power contributions from individual components of the MCB to the overall cycle time and and power consumed by each MCB. From Fig. 5, we note that for conventional 6-T SRAM based MBC framework, the memory array contributes *29.5%* to the MCB cycle time and *24.3%* to the MCB power per cycle. From Fig. 5,

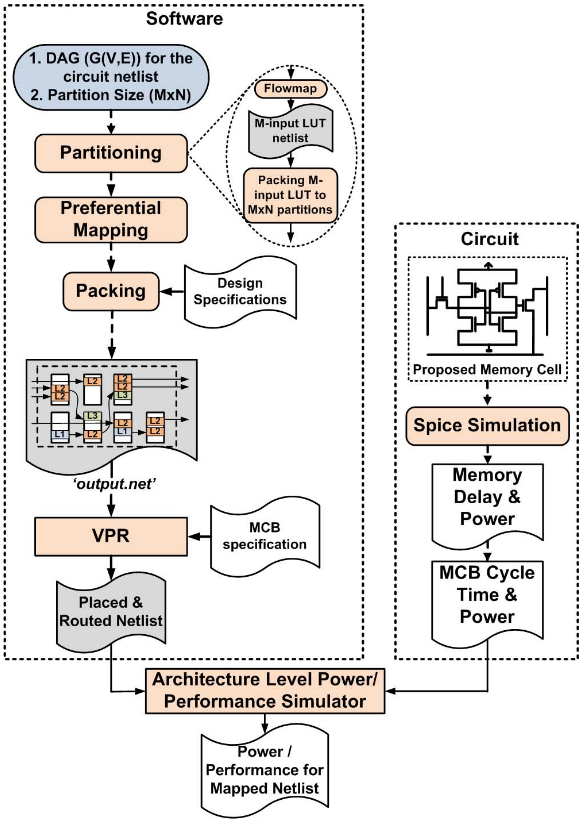

Fig. 4. Flowchart showing the essential steps of an integrated design/simulation framework.

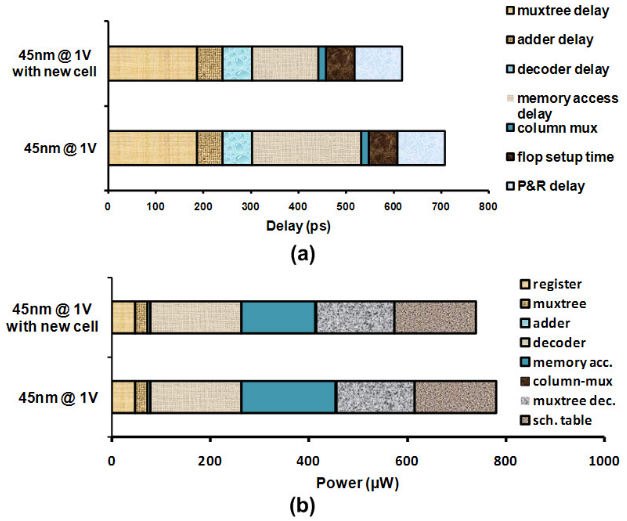

Fig. 5. Breakdown of a) MCB cycle time and b) Power per partition in each MCB for conventional 6-T SRAM and the proposed 6-T memory cell.

TABLE I
EDP IMPROVEMENT RESULTS IN THE MBC FRAMEWORK WITH NEW MEMORY CELL DESIGN AND CONTENT–AWARE MAPPING

	Delay (ns)			Energy (pJ)				EDP (pJ-ns)			
Ckts	FPGA	MBC with conv. 6T	MBC with alt. 6T	FPGA	MBC with conv. 6T	MBC with alt. 6T	MBC with alt. 6T and map	FPGA	MBC with conv. 6T	MBC with alt. 6T	MBC with alt. 6T and map
alu4	4.45	2.43	2.16	69.84	26.05	23.65	21.03	310.58	63.22	51.01	45.37
apex2	8.16	3.24	2.88	92.83	150.37	136.53	121.70	757.17	486.59	392.66	350.00
apex4	5.35	0.71	0.62	90.56	2.96	2.69	2.48	484.59	2.10	1.67	1.54
des	3.20	2.93	2.66	139.60	68.08	61.81	56.16	447.28	199.34	164.30	149.27
ex5p	2.41	0.71	0.62	8.39	9.47	8.60	7.60	20.25	6.73	5.33	4.71
misex3	5.00	2.43	2.16	73.89	52.10	47.30	41.89	369.52	126.44	102.03	90.36
seq	5.11	3.24	2.88	97.93	130.24	118.25	105.43	499.93	421.46	340.10	303.22
spla	6.86	9.46	9.01	235.40	120.77	109.65	97.93	1614.84	1142.47	987.98	882.31
pdc	7.93	3.24	2.88	232.10	142.67	129.54	115.31	1839.86	461.69	372.56	331.63
s38417	12.42	3.57	3.21	274.90	470.05	426.79	387.05	3413.16	1677.13	1369.15	1241.67
bigkey	6.68	0.71	0.62	64.48	94.72	86.00	77.56	430.86	67.25	53.32	48.09
elliptic	24.84	3.24	2.88	30.30	60.38	54.83	49.44	752.74	195.40	157.68	142.20

we note that using the proposed memory cell design gives *12.7%* improvement in MCB cycle time and *5.2%* improvement in MCB power respectively.

A. EDP Improvement Results

The total execution time for each benchmark circuit mapped to the MBC framework was obtained by: $T_{execution} = T_{cycle} \times \#of Partitions\ in\ Critical\ path$, where T_{cycle} denotes the cycle time for a single MCB. The total energy expended in the computation was computed by $Energy = Energy_{Partition} \times \#of Partitions$, where $Energy_{Partition}$ denotes the energy required to compute a single partition. The delay and energy for the programmable interconnects was obtained from VPR using the same specifications of the routing framework as given for a clustered FPGA model at the same technology node [10]. The baseline FPGA model considered for comparison consists of 7-input LUTs present inside a cluster of 10. Table I first demonstrates that MBC framework with the conventional 6-T SRAM achieves considerable improvement (*50.1%*) in EDP over a FPGA framework at the same technology node. The improvement in EDP is more pronounced (*59.5%*) with the use of the proposed memory cell due to higher performance at lower power dissipation. The EDP can be further improved (*63.5%*) with the skewing of the LUT contents to contain more logic '0' than logic '1' values.

B. Improvement in EDP for EMB based Heterogenous FPGAs

Since the proposed memory cell improves the performance and power for each memory access, an EMB based heterogenous FPGA [2-3] can greatly benefit from the use of the proposed memory design. Standard benchmark circuits were mapped to 12-input LUTs using a mapping algorithm similar to *Heteromap* [4]. Table II shows that the proposed circuit-software co-design approach achieves *36.21%* improvement in EDP in a EMB based heterogeneous FPGA framework. Since the routing energy remains the same for the two scenarios, we have only compared the logic energy with and without the proposed co-design approach.

TABLE II
EDP IMPROVEMENT IN EMB BASED HETEROGENOUS FPAGS USING THE PROPOSED CIRCUIT-SOFTWARE CO-DESIGN APPROACH

Ckts	Dly for conv. 6T	Dly for alt. 6T	Logic Energy for conv. 6T	Logic Energy for alt. 6T w map	EDP for conv. 6T	EDP for alt. 6T w map
alu4	1.0	0.8	66.0	55.4	69.2	44.4
apex2	1.5	1.1	381.0	320.1	581.8	359.5
apex4	0.3	0.2	7.5	6.4	2.2	1.3
des	0.8	0.7	172.5	146.3	136.6	99.3
ex5p	0.3	0.2	24.0	20.1	7.0	4.1
misex3	1.0	0.8	132.0	110.5	135.6	86.1
seq	1.5	1.1	330.0	277.3	482.1	298.6
spla	1.9	1.5	306.0	257.3	573.1	388.8
pdc	1.5	1.2	361.5	303.5	552.0	354.2

V. CONCLUSION

Memory access has considerable contribution towards performance and energy in reconfigurable frameworks which use large 2-D memory array for computation. Hence, optimizing the memory based on its read-dominant access pattern can significantly benefit such frameworks. A new memory cell has been proposed which offers higher read performance and lower read power. The impact on write performance during occasional reconfiguration is addressed using low-cost circuit technique. Furthermore, leveraging on the lower read power for logic-0, we have presented a content–aware application mapping algorithm that reduces EDP by skewing the LUT contents.

REFERENCES

[1] V. Betz, J. Rose and A. Marquardt, "Architecture and CAD for Deep-Submicron FPGAs", *Springer*, 1999.

[2] J. Cong and S. Xu, "Technology Mapping for FPGAs with Embedded Memory Blocks", *FPGA*, 1998.

[3] S.J.E. Wilton,"SMAP: Heterogeneous Technology Mapping for Area Reduction in FPGAs with Embedded Memory Arrays", *FPGA*, 1998.

[4] J. Cong and S. Xu,"Performance-Driven Technology Mapping for Heterogeneous FPGAs", *IEEE TCAD*, Vol. 19, No. 11, 2000.

[5] D. Jones and D.M. Lewis, "A time-multiplexed FPGA architecture for logic evaluation", *CICC*, 1995.

[6] S. Paul and S. Bhunia, "MBARC: A Scalable Memory Based Reconfigurable Computing Framework for Nanoscale Devices", *ASP-DAC*, 2008.

[7] S. Paul and S. Bhunia, "Reconfigurable Computing Using Content Addressable Memory for Improved Performance and Resource Usage", *DAC*, 2008.

[8] Predictive Tech. Model (PTM). [Online] http://www.eas.asu.edu/~ptm

[9] VPR & T-VPack: Versatile Packing, Placement and Routing for FPGAs v4.3. [Online] http://www.eecg.toronto.edu/~vaughn/vpr/vpr.html.

[10] Intelligent FPGA Architecture Repository. [Online] http://www.eecg.utoronto.ca/vpr/architectures/.

[11] VPR Package for FPGA Power Estimation. [Online] http://www.ece.ubc.ca/~stevew/powermodel.html.

[12] J. Cong and Y. Ding, "FlowMap: an optimal technology mapping algorithm for delay optimization in lookup-table based FPGA designs", *IEEE TCAD*, Vol. 13, pp. 1-12, Jan, 1994.