Structured ASIC: Methodology and Comparison

Sam M. H. Ho [#1], Steve C.L. Yuen [#1], Hiu Ching Poon [#1], Thomas C.P. Chau [*2], Yan-Qing Ai [#1],

Philip H.W. Leong [+3], Oliver C.S. Choy [#1], Kong-Pang Pun [#1],

[#] *Department of Electronic Engineering*
The Chinese University of Hong Kong
Hong Kong

[1] {mhho, clyuen, hcpoon, yqai, cschoy, kppun}@ee.cuhk.edu.hk

[*] *Department of Computer Science and Engineering*
The Chinese University of Hong Kong
Hong Kong

[2] {cpchau}@cse.cuhk.edu.hk

[+] *School of Electrical and Information Engineering*
University of Sydney
Australia

[3] {phwl}@ee.usyd.edu.au

Abstract—As fabrication process technology continues to advance, mask set costs have become prohibitively expensive. Structured ASICs can offer price and performance between ASICs and FPGAs. They are attractive for mid-volume production and offer good intellectual property security. In this paper, a structured ASIC methodology, where 2 metal- and 1 via-mask are customised, is described. The CAD tools are fully compatible with conventional ASIC design flows and a comparison of area and delay performance with ASICs and FPGAs is given. A prototype structured ASIC implementing an LED-backlit LCD controller was fabricated in a 0.13μm CMOS process. It was verified and power consumption compared with an ASIC design.

I. INTRODUCTION

As feature sizes in semiconductor process technology continue to decrease, the cost of a full set of lithography masks has risen from over $1.5M for 90nm to $2M for 65nm [1] [3]. Complexity in large designs also increases the number of design re-spins needed before volume production [2]. Taken together, these issues considerably raise the non-recurring engineering (NRE) costs for standard cell-based ASIC designs which are becoming too expensive for low to medium volume design production. FPGAs serve as a possible solution, but these devices require large overhead compared with ASICs. This gap presents an opportunity for structured-ASIC devices. Structured ASICs consist of a repeating pattern of regular logic fabric. One or more metal/via-masks are modified to interconnect the logic to implement a given circuit. The advantage of this organisation is that different designs can share most masks, reducing NRE cost and turn-around-time.

Different vendors have diverse approaches towards structured ASICs [1], the design space spanning across cell granularity, routing constructs, metal/via mask programmability and EDA tool compatibility. On the academic side, several groups [4-6] have proposed via-configurable logic blocks and routing fabrics for use in Via Patterned Gate Arrays (VPGAs). They were compared to both standard cell-based ASICs and other via-configured cell implementations. Tong et. al. [7] compared different via-configurable lookup-tables (LUTs) circuits, while Patel et. al. [8] studied the effect of LUT sizing on VPGAs. Gopalani et. al. [9] proposed a design for manufacturability (DFM) aware structured ASIC using 2-input NAND arrays. Hsu et. al. [10] studied the buffer insertion issues for LUT-based structured ASICs containing hardwired routing fabric. Ahmed et. al. quantified the cost advantages of metal-programmable structured ASICs, and evaluated their area, delay and power trends [11], quantifying the impact of the number of programmable metal-masks.

In this paper, a 2 metal- and 1 via-mask customised structured ASIC is proposed. The main contributions of this paper are:

- Quantitative comparison of area, power and delay overheads compared with ASICs and FPGAs,
- Demonstrate a practical structured ASIC implementation with consideration on issues such as antenna design rules, etc., with testing on a fabricated chip performed,
- A fully ASIC compatible CAD flow proposed.

The rest of the paper is organised as follows. Section II details the architecture and CAD flow of our structured ASIC. Section III discusses the benchmarks and comparison results of our structured ASIC compared with ASICs and FPGAs. The silicon verification and power analysis of a real world sample design is given in Section IV. Finally, conclusions are presented in section V of this paper.

II. FABRIC ARCHITECTURE & CAD FLOW

Since a structured ASIC fabric is a regular array of interconnected logic cells, an obvious solution is to build a configurable lookup-table (LUT) cell as is used in FPGAs, with static RAM (SRAM) based configuration being replaced by configuration using vias or metal. Tong et. al. studied different logic cells [7], and reported that the transmission gate (TG) style provides good power and delay performance, with the cell having an added advantage of small area. Figure 1 shows the schematic of the TG style LUT chosen for our structured ASIC. Each cell contains 3 input inverters which generate complementary signals for the TGs. Complementary inputs were not considered since they double the routing requirements. The function generator is a 4-to-1 MUX, which can be considered a 2-input LUT if we tie individual inputs

n1-n4 to 0/1s. A technique of source-driving is applied here, where the 3rd input C and its complement CX can optionally be connected to any of these 4 inputs, effectively turning the cell into a 3-input LUT [7]. This reduces the number of transistors required, from 28 of a native 3-LUT (8-to-1 MUX) to 14. Although a LUT size of 4 is shown optimal in Patel et. al.'s work [8], sizing up the LUT with source-driving deteriorates performance since the signal needs to pass through one more layer of TG, and so we stayed with a 3-LUT configuration.
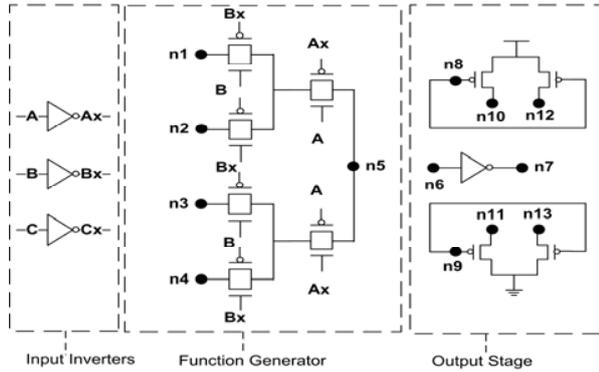

Fig. 1 3-LUT TG cell style

The output stage of the cell contains a configurable buffer. Together with the function generator, 3 driving strengths (1×, 2× and 3×) are available. We built a digital cell library for every 3-input function to provide a Synopsys standard cell compatible synthesis flow. Standard cell libraries usually offer logic cells in 4 driving strengths: 1× to 4×, but 3 different strengths lead to smaller TG cell area and our experiments showed they are adequate for smaller designs or designs with looser timing constraints. Other than logic functions, special cells such as tri-state buffers, tie cells, filler cells, D-latches and clock-gating cells can be implemented using TG cells. To avoid issues with custom flip-flops, we chose to use a heterogeneous architecture employing LUTs for combinatorial circuits and dedicated flip-flops as storage elements. A full-featured flip-flop from the standard cell library is used. Each of these is accompanied with tie high/low cells so that unused inputs are connected. Diodes are placed next to each flip-flop, so that a net can be rerouted through the diode to address antenna design rules [15].

In the layout of the LUT cell, metal-3, via-3 and metal-4 are used for (1) configuring individual logic cells, and (2) routing between LUTs. In each LUT cell, 50% of metal-3 and 100% of metal-4 is available for routing. Optional layers for routing above metal-4 are possible, and may be necessary for designs with have high routing demand. The digital cell library is built with the Cadence Encounter Library Characterizer (ELC) tool, which outputs in Liberty library format. All synthesis results reported in this paper for ASICs and structured ASICs were obtained using Synopsys Design Compiler.

The backend is a modified ASIC flow. In figure 2, the left flow is used for building a fabric: predefine legal sites for flip-flops and macros, and do power planning. Based on experiments a LUT to FF ratio of 8:1 was found to be sufficient. These steps are separated in an individual flow so that the resulting fabric can be reused for different designs.

The flow in the middle and right hand columns of the figure show how specific designs are implemented on a fabric. Steps in dashed boxes are unique to the structured ASIC while the remaining steps are standard ASIC steps.
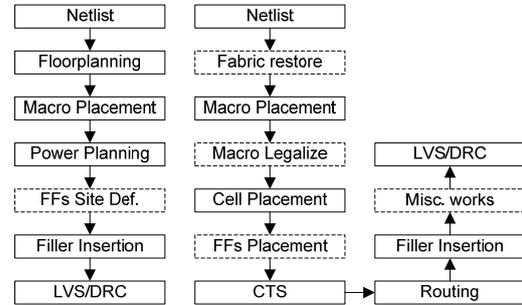

Fig. 2 Backend flows for the structured ASIC

Logic cell placement is similar to ASICs, but with predefined sites for structured elements blocked against logic cells. After that, placement of the structured elements is performed using custom scripts. Here we apply a greedy algorithm, which moves the flip-flop under consideration to the nearest legal site. After placement of flip-flops, optimization steps are still capable of moving cells in-front and behind the flip-flop on its timing path, thereby compensating the effect of predefining FF legal sites. After these processes, clock tree synthesis and global detail routing is performed, on metal-3 and metal-4. In the last "Misc works" step, inputs of all unused flip-flops are tied high.

The flow just described is highly compatible with a typical ASICs backend design flow, and can be operated at the users' site. It also eliminates the need for dedicated structured ASIC CAD tools. Moreover, compatibility allows users to do further verification after physical design is completed.

III. COMPARISON METRICS & BENCHMARKS

Our structured ASIC was evaluated over a number of benchmarks. The circuits chosen were the larger ones from the IWLS 2005 benchmarks [13].

For both ASIC and structured ASIC synthesis, we adapted the method of Kuon et. al. [14] to compare FPGA and ASIC performance. The desired clock rate is set to an unattainable 2GHz during a 1st round synthesis, and the resulting frequency obtained used during a 2nd round of compilation from which a maximum clock frequency was recorded. The Faraday standard cell library for UMC 0.13μ High Speed process [15] was used. Typical case libraries were used during synthesis for both the ASIC and structured ASIC designs. For designs with memories, they contribute 0.042 mm^2 of the area in the "Ethernet", and to 0.096 mm^2 of area in the "vga_lcd".

For FPGAs, the Xilinx ISE 10.1 tools were used and no clock speed constraints were provided since the default

settings already gives good results. On chip block memory was instantiated as appropriate.

For placement and routing, an initial utilization of 0.75 is set for floorplanning all the benchmark designs in ASIC flow such that all designs can finish without design rule checker (DRC) errors. All metal layers were set to be usable in the ASIC flow. Worst case timing libraries were used.

A comparison of area utilization between structured ASICs and FPGAs is not straightforward. For FPGAs, the flow targeted a Xilinx Virtex-II XC2V3000 device, speed grade -6, which is fabricated in a 0.12μm transistor/0.15μm 8-metal-layer process. Since the area of a CLB on a Virtex-II device is reported to be $50000\mu m^2$ in reference [12], we multiply this value by the number of CLBs used for individual designs, giving an estimate of the die area on the FPGA. For the structured ASIC, we build a smallest fabric for each specific design, and then extract the total cell area after place and route. The clock period is obtained from Synopsys PrimeTime with typical case libraries used.

Table I shows the area and clock period results obtained for ASIC, structured ASIC and FPGA approaches over the benchmark set. The column titled S/A presents the structured ASIC to ASIC ratio, while the column F/A is the ratio of FPGA to ASIC. The last row gives the geometric mean of the ratios over all designs. The results show that our structured ASIC on average has 2.7× larger area and 2.7× greater delay compared with an ASIC. In comparison, the FPGA has 68× larger area and 4.4× larger delay compared with the ASIC. We would consider the area estimate of the FPGA to be on the pessimistic side and in other studies, the ratio was measured to be 35 on average, albeit on a different benchmark set [14].

The structured ASIC has a 25× area and 1.6× performance advantage over the FPGA. For designs that require more routing resources or are relatively small in area, e.g. des_area, the area overhead for the structured ASIC is larger. Also, it can be seen from the table that structured ASIC has a lower delay than the FPGA for all but one design, "des_perf".

Structured ASIC has an area delay product ranging mostly below 10×, with a geometric mean of 7.4× to ASIC. The FPGA's spectrum spends a wider range from below 100× to around 600×, with a geometric mean of 298.4× to ASIC. Combined together, the structured ASIC is 40.3× better than FPGA in area delay product. This shows that our structured

TABLE II COMPARISON OF DIFFERENT STRUCTURED ASICs

Researchs	Area / delay v.s. ASIC	Custom layers	Non-commercial CAD flow
Jayakumar et al. [17]	4.96×/2.89×	Above M2	Synthesis
Gulati et al. [18]	6.08×/2.01×	7, (M1-M4)	Synthesis
Ran et al. [4]	2.16×/1.33×	5, (Via12, above M2)	SIS, Capo, custom maze router
Li et al. [5]	3×/2.7×	4, (Via12, M3 & up)	Logic packer, placement legalizer
Gopalani et al. [9]	1.12×/1.4×	M1 & up	SIS
Tong et al. [7]	N/A	N/A	Logic packer
Patel et al. [8]	N/A	Via23 & up	SIS, FlowMap, T-Vpack, VPR
Ours	2.7×/2.7×	3, (M3-M4)	Placement legalizer

ASIC is able to achieve the goal of filling the gap between ASICs and FPGAs.

Table II shows a comparison of our structured ASIC to previous approaches. The logic structure in [17] is PLA style, while that in [18] is a pass-transistor style if-then-else (ITE) cell. These approaches required custom synthesis tool support. The logic structure used in the work of [4] is a CMOS style via-configurable functional cell (ViaCC), and a configurable inverter array that also implements a 2-to-1 multiplexer. The work of [5] used a via-programmable CLB (VCLB) logic structure that can be used to implement both CMOS and pass transistor logics. The work of [9] used was solely a single 2-input NAND gate, where buffers and inverters are constructed by connecting the NAND cells in parallel. The approach shares only the masks up to the poly-layer, significantly reducing its appeal as a structured ASIC compared with ASICs. The approaches in [4] [9] used the SIS tool for synthesis. The approach in [5] is mostly compatible with commercial tools, but requires a custom logic packer and a placement legalizer. The work of reference [7] used Synopsys Design Compiler for all design mappings to different structured ASIC styles. However, custom tools for netlist compaction and packing of logics into CLBs are needed. In the work of [8] which compared the LUT size for VPGA cells, SIS and FlowMap were used for logic mapping, while T-Vpack is used for packing logic and Versatile Place and Route tool (VPR) used for place and route.

IV. SAMPLE APPLICATION & POWER ANALYSIS

We mapped and fabricated a real world design example onto our structured ASIC to verify its functionality. The

TABLE I DELAY AND AREA OF CIRCUITS ON ASIC, FPGA AND OUR STRUCTURED ASIC.
S/A IS THE RATIO OF STRUCTURED ASIC TO ASIC AND F/A IS THE RATIO OF FPGA TO ASIC.

bench	area (mm²)					period (ns)				
	ASIC	sASIC	FPGA	S/A	F/A	ASIC	sASIC	FPGA	S/A	F/A
s35932	0.263	0.844	22.975	3.21	87.50	0.960	3.310	4.151	3.45	4.32
s38417	0.267	0.743	21.663	2.79	81.28	1.460	4.480	9.503	3.07	6.51
s38584	0.246	0.627	20.400	2.55	82.98	1.030	3.130	6.472	3.04	6.28
b14_1	0.198	0.422	16.463	2.14	83.34	2.530	5.940	17.492	2.35	6.91
b15_1	0.191	0.448	18.925	2.35	99.31	2.140	5.300	11.398	2.48	5.33
des_area	0.115	0.414	8.063	3.60	70.18	2.040	5.950	6.240	2.92	3.06
des_perf	2.070	4.918	93.850	2.38	45.33	3.400	6.230	5.373	1.83	1.58
vga_lcd	0.429	1.047	10.775	2.44	25.12	1.670	4.930	6.400	2.95	3.83
ethernet	0.354	1.246	30.038	3.52	84.81	1.880	4.900	8.826	2.61	4.69
AVG				2.73	68.48				2.70	4.36

TABLE III AVERAGE POWER ANALYSIS OF SAMPLE DESIGN
ON ASIC AND STRUCTURED ASIC

	Switch Pwr (mW)	Int Pwr (mW)	Leak Pwr (mW)	Total Pwr (mW)
ASIC	0.921	0.205	0.032	1.160
sASIC	0.433	0.650	0.380	1.460

design was part of a controller circuit used to control individual LED brightness for an 18x10 LED backlit for LCD panel. The number of LEDs used is dependent on the specific backlit panel used. This makes it a good application for a structured ASIC for three reasons: (1) it targets a consumer device and hence will be produced in high volume (2) it is cost sensitive (3) different integrated circuits are required for different LCD families.

A clock constraint of 100 MHz was used for synthesis according to the application's requirements. PrimeTime PX is used for average power analysis. A sample picture is transformed into a simulation vector and used as a typical input of the circuit. Synopsys TetraMAX was used for automatic test pattern generation (ATPG), and both scan and functional test is performed after place and route.

Reports from static timing analysis after place and route show that the ASIC is capable of running at a 9.87ns period and the period of the structured ASIC was 12ns. Compared in this way, the structured ASIC is 25.6% slower than the ASIC. This shows that in real application with modest clock frequency requirements, the actual delay overhead of structured ASIC is less than that indicated in the previous section.

Table III shows an average power analysis of the design for both ASIC and structured ASIC, obtained using PrimeTime PX. The structured ASIC uses about half the switch power of ASIC, but consume more than three times the internal power and twelve times the leakage power, attributed to the LUT-based architecture. Each 3-LUT is accomplishing more work than single gates in ASIC, where 2-input gates are mainly used. Less wiring is needed and so less switching power is consumed. Since the LUTs are transmission gates, they have higher internal and leakage power. Overall, the sample design consumes about 26% more power in structured ASIC than in ASIC, for this typical sample input vector.

The fabricated chip is verified using both scan and functional tests on a HILEVEL Griffin tester. Results of other testing circuits included on the chip can be found in another work [16]. After verification, the chip is integrated into a running demonstration system. The chip is mounted to a custom PCB to connect a FPGA board through its GPIO for PC interfacing. The FPGA used was a Xilinx Virtex-5 in the ML555 development kit, installed on a PCI-E slot in a Linux PC. The remaining parts of the application also reside on the FPGA. Jungo WinDriver was used for the PCI-E interface.

Conclusion

A structured ASIC methodology that addresses practical implementation issues is proposed. Metal-3, via-3 and metal-4 are used as the programmable layers for both logic configuring and routing. The methodology is highly compatible with a conventional ASIC design flow and avoids EDA tools support issues. Benchmarks show that designs has a 2.7× area- and 2.7× delay-overhead compared with an ASIC, while being 25× smaller and 1.6× faster than an FPGA. In our LED backlight controller design, the structured ASIC consumes approximately 26% more power than ASIC. The proposed structured ASIC is fully verified on silicon with both scan and functional testing performed. A fabricated chip is integrated into system with a FPGA board as an interface. This work demonstrates the viability of the structured ASIC as a platform for integrating hard-IPs into a SoC in mass production as FPGA replacement.

References

[1] D. D. Sherlekar, "Design Considerations for Regular Fabrics," in Proc. ISPD'04, pp.97-102

[2] A. Sangiovanni-Vincentelli, "The tides of EDA," Design & Test of Computers, IEEE , vol.20, no.6, pp. 59- 75, Nov.-Dec. 2003

[3] C.M. Weber, C.N. Berglund, P. Gabella, "Mask Cost and Profitability in Photomask Manufacturing: An Empirical Analysis," Semiconductor Manufacturing, IEEE Transactions on , vol.19, no.4, pp.465-474, Nov. 2006

[4] Y. Ran, M. Marek-Sadowska, "Designing Via-Configurable Logic Blocks for Regular Fabric," IEEE Trans. on VLSI Systems, Vol. 14, No.1, Jan. 2006

[5] M.C. Li, H.H. Tung, C.C. Lai, R.B. Lin, "Standard Cell Like Via-Configurable Logic Block for Structured ASICs," ISVLSI'08, pp.381-386

[6] T.C.P. Chau, P.H.W. Leong, S.M.H. Ho, B.P.W. Chan, S.C.L. Yuen, K.P. Pun, O.C.S. Choy, X. Wang, "A Comparison of Via-programmable Gate Array Logic Cell Circuits," FPGA'09, pp.53-61

[7] K.Y. Tong, V. Kheterpal, V. Rovner, L. Pileggi, H. Schmit, R. Puri, "Regular Logic Fabrics for a Via Patterned Gate Array(VPGA)," Proc. of Custom Integrated Circuits Conference, Sept. 2003, pp.54-56

[8] C. Patel, A. Cozzie, H. Schmit, L. Pileggi, "An Architectural Exploration of Via Patterned Gate Arrays," ISPD'03, pp. 184-189

[9] S. Gopalani, S.P. Khatri, R. Garg, M. Cheng, "A Lithography-friendly Structured ASIC Design Approach," GLSVLSI'08, pp. 315-320

[10] P.Y. Hsu, S.T. Lee, F.W. Chen, Y.Y. Liu, "Buffer Design and Optimization for LUT-based Structured ASIC Design Styles," GLSVLSI'09, pp. 377-380

[11] U. Ahmed, G.G.F. Lemieux, S.J.E. Wilton, "Area, Delay, Power, and Cost Trends for Metal-Programmable Structured ASICs (MPSAs)," FPT 2009, pp. 278-284

[12] C.H. Ho, P.H.W. Leong, W. Luk, S.J.E. Wilton, S. Lopez-Buedo, "Virtual Embedded Blocks: A Methodology for Evaluating Embedded Elements in FPGAs," FCCM'06, pp. 35-44

[13] http://www.iwls.org/iwls2005/benchmarks.html

[14] I. Kuon, J. Rose, "Measuring the Gap Between FPGAs and ASICs," IEEE Trans. On Computer-Aided Design of Integrated Circuits and Systems, Vol. 26, No. 2, pp.203-215, Feb. 2007

[15] Faraday Cell Library FSC0H_D databook, pp.9, 2004

[16] T.C.P. Chan, D.W.L. Wu, Y.Q. Ai, B.P.W. Chan, S.M.H. Ho, O.K.L. Lau, S.C.L. Yuen, K.P. Pun, O.C.S. Choy, P.H.W. Leong, "Design of a Single Layer Programmable Structured ASIC Library," DDECS'10, pp. 32-35

[17] N. Jayakumar and S.P. Khatri, "A metal and via maskset programmable VLSI design methodology using PLAs," in Proceedings, IEEE/ACM International Conference on Computer-Aided Design, pp. 590-594, Nov 2004.

[18] K. Gulati, N. Jayakumar, and S.P. Khatri, "A structured ASIC design approach using pass transistor logic," in ISCAS pp. 1787-1790, IEEE, 2007.