New Power-Efficient FPGA Design Combining with Region-Constrained Placement and Multiple Power Domains

Ce Li, Yiping Dong and Takahiro Watanabe
The Graduate School of Information,Production and Systems,Waseda University
Fukuoka, 808-0135, Japan
Email: rice@fuji.waseda.jp

***Abstract*—Multiple power domain design architectures have been studied for the power-efficient FPGAs. But, most of these researches pay attention on the clustered logic block's fine-grain power gating which increases the FPGA size significantly. This paper presents a fast placement algorithm for coarse-grain FPGAs architecture, by which the circuit with multiple power domains is mapped into several regions for low power consumption. Each region uses one or several sleep transistors in order to conserve leakage energy. Using the CAD framework, we discuss the power efficiency of sleep region FPGA architecture by using the benchmarks assumed in multiple power domains. Simulation result shows that 9.1% power consumption of FPGA can be reduced on average by the proposed placement algorithm, compared to the traditional algorithm. Furthermore, when the dual power domains are individually power-on and -off, our proposed method can reduce the power more than 20%.**

I. Introduction

Power is a significant design constrain in the demand of battery-power devices. The power consumption limits Field-Programmable Gate Array (FPGA) in the portable digital applications. Many studies pay attention on the low power design methods of FPGA. Dual-VDD method is mentioned in [1], a field programmability of supply voltages for FPGA is studied in [2]. The area is induced after applied fine-grain power to FPGA obviously.

In Ref. [3], the power gating of logic fabrics was investigated and region-constrained placement was applied to reduce the leakage power of unused logic blocks on Xilinx FPGA. The placement algorithm places a designed circuit into contiguous regions by utilizing two different styles: horizontal and vertical placement. One of the limitations of this idea is that the parietal row cannot be used until the lower rows are fully filled in horizontal placement. So, the circuit for the IO PAD on the FPGA top edge may be placed in the bottom row when the FPGA size is large. It will increase the wire length and decrease the FPGA performance.

Ref. [4] presented dynamical power gating FPGA architecture that compose of regions, but they did not discuss the placement algorithm. In this paper, we propose a region-constrained placer based on this FPGA architecture by enhancing VPR [5]. VPR placer paid more attention on the wire length and critical path affected by the placement. But this kind of placement could not support the low power FPGA architecture with power domains (PD). CLBs which have the same power supply and the same power state are called in the same PD. Therefore, the placement algorithm for the circuit with multiple PDs (multi-PD) is explored. It is used not only to fully fill the regions with CLBs which belong to the same PD, but also to power off the unused region, because each region should support one power supply each time. We focus our effort on the circuit placement algorithm because it plays an essential role on the FPGA PD distribution.

The remainder of this paper is organized as follows. In the next section, low power design background is described. Placement algorithm based on regions and multi-PD is given in Section III. The CAD framework which supports this new FPGA architecture is shown in Section IV. Finally, experimental results and the conclusion are presented.

II. Background

A. Multiple Power Domains

The current FPGA architecture has the limitation in PD design. The basic units in a conventional FPGA [6] are Clustered Logic Block (CLB), Input Output Block (IOB), Connection Box (CB), Routing Channel(RC) and Switch Box (SB). FPGA implements the user's circuit by dividing it into small pieces which can be achieved by CLBs. Other units like CB, WC and SB are used as wires to connect the input and output among the CLBs. Multiplexers (MUX), buffers and transmission gates are also used to achieve the flexibility and reprogram ability of FPGA. FPGA users generally pay more attention on optimizing their circuits, but can do little improvement based on the fixed FPGA architecture. So, many multi-PD design methods based on their circuit could not be implemented on traditional FPGA chips.

Meanwhile, many low power design methods are studied, such as power gating and Dual-VDD. But double VDD supplies increase designs complexity. Not only do we need to add IO pins to supply the different power rails, but also need a more complex power grid and level shifters on signals running between PDs [7]. Usually, many ASIC designs are composed of modules, where the supply voltage in each module is fixed. Dual-VDD and level shifter should not affect the placement algorithm based on the module level design. For this reason, we simplify the FPGA model that all CLBs have the same

978-1-61284-137-3/11/$26.00 ©2011 IEEE

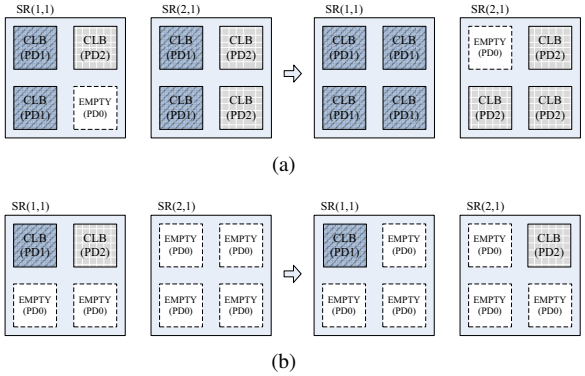


Fig. 1. CLB swapping results.

power voltage but could power off separately. If we need multiple VDDs, we can add those low power methods directly. Therefore, this paper focuses on the floor plan algorithm for multiple power domains with single voltage in this paper.

B. Sleep Region Architecture

A region based dynamical power gating architecture is proposed for island style FPGA in [4]. We call it Sleep Region (SR) in which several CLBs and RCs are power-gated by one PMOS sleep transistor. An SR is the minimum unit of the PD, and it is composed of CLBs in $k * k$ array. The SR size is defined by $N_{SR-size} = k * k$. To make the multi-PD idea clearly, simple examples of the CLB placements on FPGA with two SRs ($N_{SR-size}$=4) are shown in Fig.1(a) and Fig.1(b), where the left is the result after the initial placement and the right is after the CLBs swapping. After the initial placement, some SRs may include CLBs in different PDs; PD1 and PD2. After CLBs swapping, each SR uses only one PD, so that power-on or -off can be performed for each SR. So, a fast and effective placement algorithm should be explored.

III. Placement Algorithm

FPGA placement algorithm determines the CLB location of the circuit on chip. The goals are to minimize the total area, wire connection, the delay of critical path and so on. We focus our effort on finding a fast method which can separate the different power domain CLBs into different SRs.

A. Wire Cost and Timing Driven Cost

Based on the adaptive simulated annealing schedule, VPR [5] can get a better placement result in a short time. By using the linear congestion wire cost function $Cost_W$, VPR models the difficulty of routing connections in areas with different channel widths.

To get higher FPGA performance, the path timing driven cost, $Cost_T$, is also used in [8] based on the Elmore timing model. With the $Cost_T$, the CLBs connected by the critical path can be placed closely to reduce the path delay of the whole FPGA chip. So, the critical path delay becomes smaller.

B. Sleep Region Cost

The above-mentioned costs are not enough to get a good placement in our SR-architecture with multi-PD. We introduce a new cost, $Cost_{SR}$, to indicate the sleep region cost.

Two parameters are used for the PD information for each SR. One is the PD count in each SR, called $N_{pdc}(m,n)$, that is the number of different PDs used in the SR located at coordinates (m,n) on the FPGA chip. The other is the CLB count, $N_{CLB}(m,n,p)$,that is the number of CLBs with the p-th PD in the SR located at (m,n), where p is the index of PD and p=0,1,2, ..., N_{pdc}. p=0 means the current CLB slot is EMPTY, that is, no CLB is assigned to the slot. The placement goal for the SR architecture FPGA is that reducing $N_{pdc}(m,n)$ less than 2 and increasing $N_{CLB}(m,n,p)$ as large as possible for each SR.

A cost function, $Cost_{SR}(m,n,p)$, is introduced to indicate the PD cost of SR at the coordinate (m,n) of the FPGA. It is award of the swapping for the CLB count increasing in SR. The $Cost_{SR}(m,n,p)$ turns to zero when SR is fully filled with the CLBs in the same PD or without any CLBs. The cost function is shown as follow,

$$Cost_{SR}(m,n,p) = \begin{cases} 1-(\frac{N_{CLB}(m,n,p)}{N_{SR-size}})^2 & N_{CLB}(m,n,p) \neq 0 \\ 0 & N_{CLB}(m,n,p) = 0 \end{cases} \quad (1)$$

The total SR cost is the summation of all the $Cost_{SR}(m,n,p)$ on the FPGA shown in Eq.2. Power domain 0 is used for the "EMPTY" slot. Its cost does not need to be added into $Cost_{SR}$. But if we only do the addition of $Cost_{SR}(m,n,p)$ in each power domain, we cannot handle the case shown in Fig.1(b). The total cost of these two SRs will not be changed after swapping. So, we use $N_{pdc}(m,n)$ to penalize the case that one SR has the CLBs from different PDs.

$$Cost_{SR} = 1 + \sum_{m=1}^{Width} \sum_{n=1}^{Length} \sum_{p=1}^{N_{pdc}} Cost_{SR}(m,n,p) * N_{pdc}(m,n) \quad (2)$$

C. FPGA Total Cost

Similar to [8], we update the FPGA full chip cost function, $Cost_{WTSR}$, based on the cost of $Cost_W$, $Cost_T$ and $Cost_{SR}$. To do the tradeoff between each cost, factors γ and τ are introduced shown in the Eq. 3. Each cost at the old temperature (oldT) in SA process is added to balance the different cost. It makes the tradeoff factor more accurate.

$$Cost_{WTSR} = (1-\gamma)((1-\tau)\frac{Cost_W}{Cost_{W\_oldT}} + \tau\frac{Cost_T}{Cost_{T\_oldT}}) + \gamma\frac{Cost_{SR}}{Cost_{SR\_oldT}} \quad (3)$$

In the original VPR environment, τ is set to 0.5 to balance the wire cost and path timing driven cost [8]. Based on the

experiments, the γ is set to 0.1. It can make sure the circuit has a better performance based on the $Cost_T$ and $Cost_W$ when support the SR placement. Of course, it supports the fixed γ provided by the user to do the tradeoff between placement success rate and design performance.

Millions of potential block swaps will be executed in a typical placement even if a good annealing schedule is used. Therefore, the fast computation is indispensable. To reduce the CPU time during the placement, we do not re-compute the $Cost_{WTSR}$ during each swap. The swapped CLBs (or even swap to a empty slot) come from either the same PD or the different PDs. We just calculate the cost change based on the affected nets and SRs.

The changes of $Cost_W$ and $Cost_T$ due to the affected nets by the two swapped CLBs (or by moving a CLB to the empty slot) are calculated. The $Cost_{SR}$ is affected by the SRs whose $N_{CLB}(m, n, p)$ is changed. Based on Eq.1 and Eq.2, only swapping two CLBs in the different PDs between different SRs can affect the cost, but the swapping inside the same SR does not affect $Cost_{SR}$.

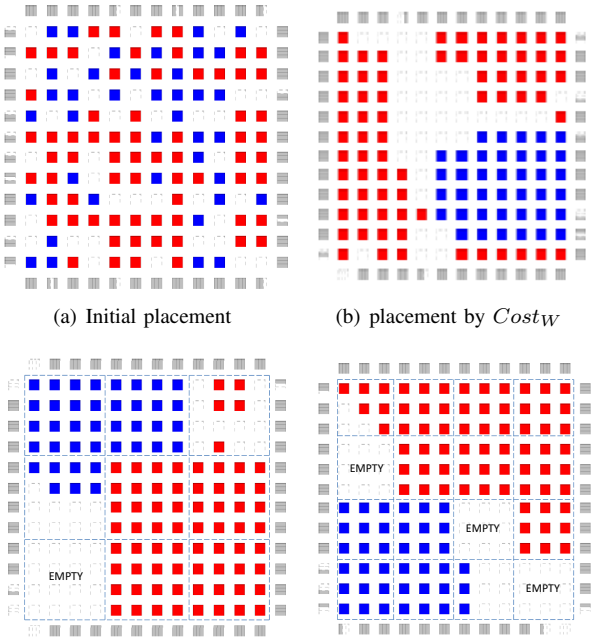
During each swap, we pick one CLB of the circuit randomly and get its information, such as coordinate in grid scale and PD. Then, we choose the destination within the specified swapping range of two CLBs. The $Cost_{SR}$ is calculated when the start point and the destination point are not in the same PD and SR.

The parameters used in our Simulated Annealing procedure such as the conditions to exit SA and temperature control are same as those used in the original VPR [5]. When the swapping range is small in the low temperature, the $Cost_{SR}$ is hardly changed. Only the $Cost_W$ and $Cost_T$ optimize the local placement to reduce the wire cost. Therefore, CLBs should be placed into the minimum count SRs in the early stage of placement without affecting the performance and area so much. By using the cost functions and adjusting parameters, the placer promises to cause the minimum count SRs.

D. Cost Function Comparison

Different cost functions and $N_{SR-size}$ are compared based on the same FPGA architecture in Fig.2. As far as we know, there is no benchmark to evaluate the multi-PD feature on FPGA. So, we use two MCNC benchmarks instead of two modules with different PDs. Although there are no interconnections between MCNC benchmark data, it is enough to experiment and evaluate our placement algorithm.

Fig.2(a) shows the initial placement after VPR is enhanced to get two circuits. The CLB in different color comes from different PDs. After the initial placement, the CLBs are placed randomly in the FPGA. The placement result by using $Cost_W$ is shown in Fig.2(b). Since there is no connection between two PDs, the CLBs in different PDs are placed separately. The blank grids located between the two PDs are disordered. Although the blank space in big enough for a 4*4 sleep region, but it is impossible to use power gating for the unfixed position. More efforts should be taken if we want to use some low power design methods. But, in Fig.2(c), the SR placer can



(a) Initial placement (b) placement by $Cost_W$

(c) placement by $Cost_{WTSR}$ (d) placement by $Cost_{WTSR}$

Fig. 2. Placement results by different cost

place CLBs by using the minimum number of SRs of 4*4 size. One region can be power off in this case. If we change the SR size to 3*3, three SRs can be always power off as shown in Fig.2(d).

IV. CAD Framework

A CAD framework should be explored to support our proposed SR architecture and to evaluate the power gating architecture. Fig.3 illustrates the CAD software flow supporting two PDs, which composes a CAD framework for placement, routing and power simulation for the un-commercial FPGA.

The Berkeley Logic Interchange Format (.blif) files are used to describe the MCNC benchmark circuits. The T-VPACK[5] program packs the LUTs and flip-flops into CLB which contains one or more LEs in .net format for each PD.

Power estimation framework based on VPR 5.0 is provided by [9]. To compare the power consumption, a power model that mentioned in [10] in terms of dynamic power, short-circuit power and leakage power is used in this framework. We embed our region-constrained placement algorithm based on this framework.

VPR could support kinds of FPGA architecture with different LE count, CLB count or channel width based on the wire and timing cost function. We enhanced the original VPR to support FPGA with multi-PD, called SR-VPR,

The inputs of SR-VPR are multiple circuits with different PDs. The SR setting parameters, such as $N_{SR-size}$ are also input. After the placement, the SR-VPR routes the internal connection. The outputs of SR-VPR are the area, critical path delay, circuit placement and routing information and power

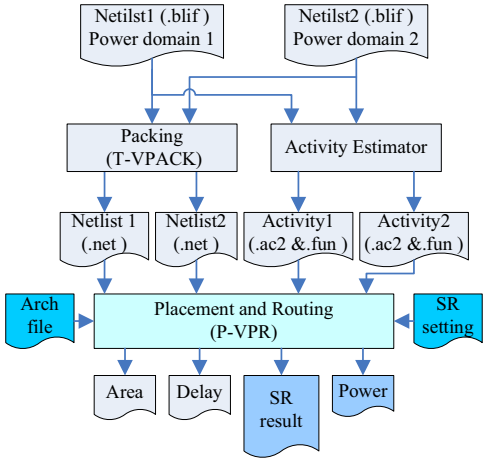


Fig. 3. Software flow for the SR based FPGA

TABLE I
THE FPGA POWER CONSUMPTION RESULT WITH PDS

		Power Consumption($10^{-2}W$)				
PD 1	PD 2	NORMAL	DESIGNON	PD1ON	PD2ON	PWROFF
alu4	alu4	2.39	2.12	1.73	1.72	0.90
apex2	alu4	2.26	2.04	1.69	1.69	0.97
apex4	alu4	2.27	1.88	1.60	1.69	1.06
bigkey	alu4	2.14	1.80	1.43	1.61	0.87
des	alu4	2.44	2.21	1.79	1.66	0.86
diffeq	alu4	1.74	1.75	1.31	1.57	0.81
dsip	alu4	2.27	1.89	1.49	1.69	0.95
ex5p	alu4	1.62	1.63	1.24	1.44	0.70
misex3	alu4	2.37	2.06	1.68	1.76	0.96
s298	alu4	1.41	1.46	1.13	1.38	0.70
seq	alu4	2.29	2.10	1.73	1.73	0.93
tseng	alu4	1.78	1.75	1.31	1.55	0.75
average		2.08	1.89	1.51	1.62	8.70
percent		100%	90.9%	72.7%	78.0%	41.8%

consumption. The SRAM value for the PMOS gating transistor can also be generated in the output file for each SR.

To import SR, not only the net delay, but also the PD should be considered. SR-VPR can also find a better solution to place and route the circuit to deal with different architectures. SR-VPR can automatically adjust the size of the FPGA based on the target circuits and the $N_{SR-size}$. In our experiments, both SR and FPGA are set to square.

V. EXPERIMENTS RESULT

Multi-PD design was experimented where twelve MCNC benchmark circuits are used as the first power domain PD1 and the circuit $alu4$ is used as the second power domain PD2. We use the .blif netlist whose CLBs consist of four Logic Elements (LEs). Each LE has one 4-LUT and one flip-flop. Large $N_{SR-size}$ will increase FPGA size for SR. In our experiment, $N_{SR-size}$ is 4*4. We use HSPICE to generate power information of 45 nm process technology. The routing parameters are set as Fc_{in}=0.25, Fc_{out}=0.1, Fs=3, and segment length L=4 is the same as those parameters used in the original VPR. Please refer [11] for the details of these parameters.

Table I shows the power estimation result based on multi-PD. The "NORMAL" VPR result is shown in the third column. The "DESIGNON" column means that all the unused SRs are power off. We could get 9.1% power reduction on this state. Different PDs are power on during "PD1ON" and "PD2ON" individually. The power consumption is reduced during these power states. We could find the merit of the multi-PD for FPGA. When all the modules are idle, the FPGA could enter a "PWROFF" state which could reduce the power consumption by 58.2%. The 41.8% power is cost by the CB, SB and SRAM.

VI. CONCLUSION

We have proposed a new FPGA placement algorithm for the circuit with multiple power domains, where a circuit is mapped into several disjoint regions. The CLBs in the different power domain could be placed in the minimum SRs separately and each SR can be powered-on or -off independently. The proposed method combining with multi-PD can reduce power consumption effectively. The further work will focus on the low power routing architecture to reduce the power of switch box, connect box and wire channels.

ACKNOWLEDGMENT

This work is partly supported by Regional Innovation Cluster Program (Global Type, 2nd Stage), MEXT and the Core Research for Evolutional Science and Technology (CREST) of Japan Science and Technology Agency (JST) in Japan.

REFERENCES

[1] G. L. Vijaykrishnan, A. Gayasen, K. Lee, N. Vijaykrishnan, M. J. Irwin, and T. Tuan, "A dual-vdd low power fpga architecture," in *In Proceedings of International Conference on Field Programmable Logic and Applications*, 2004, pp. 145–157.

[2] F. Li, Y. Lin, and L. He, "Field programmability of supply voltages for fpga power reduction," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, no. 4, pp. 752–764, April 2007.

[3] A. Gayasen, Y. Tsai, N. Vijaykrishnan, M. Kandemir, M. J. Irwin, and T. Tuan, "Reducing leakage energy in fpgas using region-constrained placement," in *in Proc. ACM Intl. Symp. Field-Programmable Gate Arrays*, 2004, pp. 51–58.

[4] A. Bsoul and S. Wilton, "An fpga architecture supporting dynamically controlled power gating," in *Field-Programmable Technology (FPT), 2010 International Conference on*, Dec. 2010, pp. 1–8.

[5] V. Betz, J. Rose, and A. Marquardt, *Architecture and CAD for Deep-Submicron FPGAs*. Norwell, MA, USA: Kluwer Academic Publishers, 1999.

[6] Altera, *Stratix IV Device Handbook*, Altera Co., February 2011, http://www.altera.com/literature/hb/stratix-iv/stratix4_handbook.pdf.

[7] M. Keating, D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low Power Methodology Manual: For System-on-Chip Design*. Springer Publishing Company, Incorporated, 2007.

[8] A. Marquardt, V. Betz, and J. Rose, "Timing-driven placement for fpgas," in *Proceedings of the 2000 ACM/SIGDA eighth international symposium on Field programmable gate arrays*. New York, NY, USA: ACM, 2000, pp. 203–213.

[9] P. A. Jamieson, http://www.users.muohio.edu/jamiespa/vpr_5_pow.html.

[10] K. K. W. Poon, S. J. E. Wilton, and A. Yan, "A detailed power model for field-programmable gate arrays," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 10, pp. 279–302, April 2005.

[11] *intelligent FPGA Architecture Repository*, http://www.eecg.utoronto.ca/vpr/architectures.