Early Stage Power Management for 3D FPGAs Considering Hierarchical Routing Resources

Krishna Chaitanya Nunna  
Department of Advanced Informatics  
Kyushu University  
3-8-33-309 Momochi, Sawara-ku, Fukuoka 814-0001, Japan  
krishna@soc.ait.kyushu-u.ac.jp

Farhad Mehdipour  
E-JUST Center  
Kyushu University  
3-8-33-305 Momochi, Sawara-ku, Fukuoka 814-0001, Japan  
farhad@ejust.kyushu-u.ac.jp

Kazuaki Murakami  
Department of Advanced Informatics  
Kyushu University  
744 Motooka, Nishi-ku Fukuoka 819-0395, Japan  
murakami@ait.kyushu-u.ac.jp

ABSTRACT
Designing for low power consumption demands power-efficient devices and good design practices to leverage the architectural features without compromising performance. Power estimation at an early stage of electronic design automation (EDA) flow is essential in order to handle the design issues much earlier. In this paper, we are proposing a methodology for evaluating the power in three-dimensional field-programmable gate arrays (3D FPGAs) at an early stage of the design cycle namely at the partitioning step and making it a power-aware stage. As a part of the work, we also estimate the routing resources needed for the power evaluation. Our estimated power values are compared against the values obtained from a 3D place and route tool, TPR along with the added power calculations, which is demonstrating acceptable accuracy. Our results show that there is a scope for achieving desired distribution of power among the layers well before the placement with reasonable deviation in estimation and proves that our methodology is providing opportunity for power management at earlier stages of the design flow.

Categories and Subject Descriptors
B.7.1 [Integrated Circuits]: Types and Design Styles — Gate arrays; J.6 [Computer Applications]: Computer-Aided Engineering — Computer-aided design (CAD); I.6.5 [Simulation and Modeling]: Model Development.

Keywords
3D FPGA, power management, power-aware partitioning, early estimation, hierarchical routing resources.

1. INTRODUCTION
Three-dimensional field-programmable gate array (3D FPGA) is one of the promising innovations which can provide benefits like increasing transistor density, reduced form factor, heterogeneous architectures and improvement in delay by significantly reducing the wire lengths of integrated circuits [1]. Compared to application-specific integrated circuit (ASIC), power consumption in FPGA is most critical due to its large number of resources. Each of the available resources (logic, routing, clock etc.) will have significant contribution to the overall power consumption. In spite of advantages over two-dimensional (2D), thermal effects are expected to be significantly higher in 3D chips due to higher power density and this can cause greater degradation in device performance and chip reliability [2]. The main focus of commercial vendors and also the majority of published research on FPGA architecture and electronic design automation (EDA) are improving area efficiency and performance. Though area and speed have been the main research focus to date, power is likely to be a key consideration in the design of future FPGAs especially in 3D platform. Fast growing industry requirements result in rapid increase of circuit complexity. This is putting challenges to EDA tool developers for reducing the design time that can help make design decisions early in the process. Concentration on power at the early stage of design cycle will reduce the burden on later stages which are already having bigger problem size. Especially for 3D platform, as the design sizes increase rapidly, better handling of the critical parameters like power and temperature at the partitioning step will provide enough opportunities to the EDA developers to have more options in solving interdependent issues related to the later stages. Through our work, we are proposing a methodology for evaluating the power before placement and a power-aware partitioning algorithm for 3D FPGA considering hierarchical routing resources. As a part of the work, we study the variation in routing and logic resources between partitioning routine iterations and their effect on total capacitance and switching activity for each layer in order to evaluate the dynamic power consumption. Our methodology can be helpful in distributing the power based on the demand such as considering the location of heat sink for efficient thermal management. And also it is flexible for interfacing different types of partitioning routines available in the research domain.

The past and recent work related to this area is encouraging. In [3] authors introduce a TSV-aware partitioning algorithm for efficient utilization of the inter-layer connectivity. Ababei et al. [4] introduced 3D place and route tool for FPGAs in which the partitioning is done explicitly. The work proposed by [5] introduces a multi-level multi-layer partitioning for 3D integrated circuits (ICs). For both [4] and [5], the objective for partitioning is minimizing the number of cuts between the partitions targeting for better performance. A 3D IC partitioning routine is proposed by [6] using integer linear programs (ILPs) for minimizing the usage of vertical interconnects subject to the footprint and power consumption constraint. Work presented in [7] is performing the partitioning implicitly within placement. But as the circuit size is growing, explicit partitioning can provide opportunities for handling the critical issues at early stage of the design cycle. The
works presented above are trying to show the benefits of 3D integration in terms of reduced wire-length. But none of them dealt directly with the critical issues related to the power consumption at the early stage of design cycle. To the best of our knowledge, our work on early estimation of power consumption namely at the partitioning stage for 3D FPGAs is first of its kind. In the following, section 2 covers background on the preliminary experiments for power consumption in FPGA, 3D FPGA architecture specification and hierarchical routing resources in FPGA. Our methodology for power evaluation and routing estimation is explained in section 3. Section 4 covers the power-aware partitioning. Experiment results are discussed in Section 5 and the paper concludes along with the future work in section 6.

2. BACKGROUND

Power in FPGA comprises of dynamic and static components in which dynamic power depends on switching activities of the logic and static power is due to the leakage pertaining to the transistor structure. Earlier work in [8] and [9] shows that the dynamic power consumption in FPGA due to routing resources is much higher than others because of their large share in chip area. To get the knowledge on power distribution among FPGA resources, we have conducted preliminary experiments and results were shown in our work in [10]. We have observed that the power consumed by routing resources is much higher than the other resources and it is growing with the increase in circuit size. On the other hand, power due to logic is low and almost unchanged. From these observations it is evident that the role of routing fabric in FPGA power is momentous.

The target architecture for our work is similar to Xilinx Virtex-II device [11], which is a commercial 2D FPGA, for each layer and such identical device layers are aligned vertically using through-silicon-vias (TSVs) as shown in Fig. 1. Each layer consists of programmable resources like logic block (LB) array, connection box (CB), switch box (SB) and routing channels in X and Y directions. Each TSV is having a channel width set by the architecture definition. The hierarchical routing resources in FPGA provide a greater flexibility for routing the input circuit. In commercial FPGAs, for example, a Xilinx Virtex II device has fully buffered programmable interconnections, with a number of resources counted between any two adjacent SBs at which each vertical and horizontal routing channel interacts. The widely used academic tool VPR has similar resources with a combination of single-length (SL), double-length (DL), hex-length (HL) segments and long length (LL) wires. An example for the hierarchical routing segments in a 2D FPGA is shown in Fig. 2. Here a segment is defined as a metal wire that can span over a specific number of LBs and terminate at SBs. SL segments intersect at each SB to form a grid of interconnections. An example connection between a source LB and a sink LB routed by SL segments is shown in Fig. 3(a). DL segments are twice as long as SL ones and run past two LBs before entering a SB. Similarly HL segments run past six LBs. And example connection using DL and HL segments is shown in Fig. 3(b). LL wires bypass all SBs and form a grid of connections that runs vertically and horizontally the entire length and width of FPGA array.

3. POWER EVALUATION AND ROUTING ESTIMATION

Due to remarkable complexity of recent designs, it seems necessary to estimate performance metrics in early design phases. The availability of accurate early estimates can reduce the required number of iterations through the entire design flow, which allows design trade-offs to be evaluated at an abstract level resulting in reduction in the design time and cost. And there is an opportunity of refining the outcome of the early stage through incremental procedures. As a part of the power-aware partitioning (shown in Fig. 4(a)) we are proposing a power evaluation methodology (zoom-out) as well. As the configuration of partitions changes during the partitioning process, the amount of logic and routing resources also changes. Within the process, the variation in dynamic power consumption is measured accordingly. As the routing power dominates the other components, our work is mainly based on estimating the dynamic power consumed by the interconnections. However the power consumed by LBs is added because of variations in the number of nodes in each partition between the iterations of the partitioning routine. The dynamic power depends on the switching activity ($\alpha$), switching capacitance ($C$), frequency of operation ($f$) and supply voltage ($V$) and are related as:

$$P_{\text{dyn}} = \alpha * f * C * V^2$$  \hspace{1cm} (1)

Here $f$ and $V$ are constants for a given architecture. On the other hand, $\alpha$ and $C$ are the critical parameters in which we need to update $\alpha$ for each layer and measure $C$ time to time. For $\alpha$, probabilistic estimation is typically faster than simulation based approach since it involves a one-time calculation for each net in a circuit and it doesn’t change for a single net during the partitioning routine. Hence in our work, $\alpha$ is measured using a probabilistic approach similar to ACE-2.0 [12]. For power estimations, we need capacitance $C$ of the net as well as the associated switching elements (e.g. pass transistors). Capacitance of each net depends on its length while the exact length of each net is known only after the routing phase. At the partitioning stage, since no geometrical information is available, there should be
proper estimation criteria for capacitance. Already existing routing estimation techniques in FPGA research community provide only the probability of each net having a specific length. For more accurate power calculations, it is observed that the wirelength probability is not enough due to existing various segment types available in FPGA and various possibilities for routing a net using those hierarchical routing segments. Along with the routing segments, their associated switches also contribute to the total power consumption and hence it is necessary to know the used segments as well as used switches. In order to have such kind of routing estimation, we have already proposed a methodology for early estimation of hierarchical routing resources in 3D FPGA [10] as shown in Fig. 4(a). As each segment on FPGA plane terminates at SBs as shown in Fig. 3(a), it is straightforward to find the number of switches associated with each segment. After performing the multi-layer recursive bi-partitioning routine [13] on the input netlist, our routing estimation methodology [10] separates the obtained nets into 2D and 3D nets. As shown in Fig. 4(b), we get the probability of each net to have a specific length (wirelength distribution) for both 2D and 3D nets separately using Stroobandt’s model [14]. Depending on these values, segment estimation model calculates the number of segments used by the given circuit. Here the actual geometry of the nets is predicted depending on the possible combinations of routing the nets in different directions (X, Y and Z) separately for 2D and 3D nets. For more details on the implementation of this methodology, it is recommended to the reader to see our work in [10].

After obtaining the number of segments used by the given circuit, the distributed switching capacitance $C_{SWT, 2D}$ for 2D nets is calculated as:

$$C_{D, 2D} = \sum_{type} \frac{N_{type}}{\sum_{type} N_{type}} (C_{type} + C_{SWT}) + 2C_{SWT}$$  \hspace{1cm} (2)

In Eq. 2, $N_{type}$ is the number of segments of specific type used by the circuit, $C_{type}$ is the capacitance of each type of segment and $C_{SWT}$ is the associated switch capacitance. Both $C_{type}$ and $C_{SWT}$ are typically obtained based on the technology node. The term

$\frac{N_{type}}{\sum_{type} N_{type}}$ is the probability of using a specific type of segment out of the total segment types. The term $2C_{SWT}$ is the capacitance associated with the switches at the source’s output and sink’s input terminals as shown in Fig. 5(a). Since the typical capacitance of a TSV in 3D devices is several times of capacitance of a wire on horizontal plane, $C_{DSWT}$ for 3D nets need to be obtained separately by considering capacitance of TSV, $C_{TSV}$ (Fig. 5(b)) as given below:

$$C_{D, 3D} = \sum_{type} \frac{N_{type}}{\sum_{type} N_{type}} (C_{type} + C_{TSV} + 2C_{SWT}) + 2C_{SWT}$$  \hspace{1cm} (3)

Since we are having the number of TSVs in the form of cutsize from the partitioning routine, an extra switch $C_{SWT}$ is introduced separately for TSV (can be seen in Fig. 5(b)). Now these distributed capacitances for 2D and 3D nets and switching activities are supplied to the dynamic power calculation in Eq. 1.

The power associated with the LBs is also calculated by tracking the nodes in each partition.

4. POWER-AWARE PARTITIONING

Our power-aware partitioning algorithm is explained in Fig. 6. The routine starts with a random selection and moving the node between the initial partitions such that the number of nodes in each partition is less than the available nodes (LBs). At every movement, the partition number for each node is updated. After the current movement, we will apply our power evaluation methodology given in Fig. 4(a) (zoom-out) for dynamic power calculations.

While moving the nodes, the switching activity associated with each net is captured instantaneously. An example movement of a node between partitions is shown in Fig. 7. As shown, when node 4 is moving from partition $P_1$ to $P_2$, even though the cutsize remains same, contributing switching activities and capacitances for each partition will be affected due to the alterations in the nets’ configurations. As shown in Fig. 7(a), for example, before the movement of node 4 from $P_1$ to $P_2$, net $h$ is contributing to the power of $P_1$ due to its source node 4. As shown in Fig. 7(b), after the movement, net $h$ is contributing to the power of $P_2$ due to its source node 4 position in $P_2$. At the same time, even though nets $c$ and $d$ crossed the partitions, they contribute to the power of $P_1$ only as the source nodes of these nets are located in $P_1$. Similar way is followed for assigning $a$ and $C$ to a particular partition. After each node movement, the dynamic power for each partition is calculated. As we are targeting for desired distribution of power...
Algorithm: Power-aware partitioning

Terms: $N_{nodes}$: Number of nodes in each partition after movement, $P_{DD}$: total dynamic power, $P_{net}$: net power, $P_{logic}$: logic power, $P_{age}$: percentage power difference, $\Delta R^D$: power difference between desired distribution of layers, $P_{tol}$: power tolerance, $N_{cut}$: total cutsize, $N_{cut,MIN}$: minimum cutsize, $G$: maximum allowed cut percentage, $N_{TSV}$: available number of TSVs.

Objective: Partitioning the circuit in to $N$ parts based on power

Inputs: Netlist

Outputs: power-aware partitions

Constraints: $N_{cut} < N_{TSV}$; $N_{cut} < N_{cut,MIN} \times (1 + G)$

Function power aware partitioning (Nodes, Hyperedges)

While (1)

1. Select a random node;
2. $\{N_{nodes,partition}\}$: given array size for each FPGA layer
3. Move the node between partitions
4. Update the partition number for the node;
5. Separate the nets into 2D and 3D for each partition (Fig. 5(b));
6. Calculate $CT$ for each routing segment (Eq. 2 and Eq. 3); for each partition
   - for each net $i$ \{ Capture $\alpha$; Calculate $P_{age}$; \}
   - Calculate $P_{tot}$ for each node;
   - $P_{age} = \sum_{i \in nets} P_{net} + \sum_{lbs} P_{logic}$;

   Calculate $P_{age}$ between partitions and $P_{diff}$;
   - if ($P_{age} > previous\ P_{age}$) and ($N_{cut} > N_{cut,MIN} \times (1 + G)$) and
     - $\{ N_{cut} > N_{TSV} $\}
     - Update the partition number for the node;
   - if $((\Delta R^D - P_{tol}) < P_{age} < (\Delta R^D + P_{tol}))$ then terminate the procedure;

Figure 6. Power-aware partitioning algorithm.

among layers, the decision on fixing the nodes after movement depends on the desired power deviation $P_{DD}$. The final objective is to minimize $P_{DD}$ subject to certain constraints. It is given as,

\[ P_{DD} = \frac{\sum_{i=1}^{N} (R_i - R_{D})^2}{N} \]  

(4)

In Eq. 4, $R_i$ represents the ratio of power for each layer $P_i$ to the total power among $N$ layers. $R_{D}$ represents the desired distribution set by the user for each layer. This $P_{DD}$ is compared with the $P_{DD}$ of the previous movement. The current movement will be canceled under the conditions: $P_{DD}$ is greater than previous $P_{DD}$; cutsize is greater than $N_{cut,MIN} \times (1 + G)$, where minimum cutsize $N_{cut,MIN}$ is obtained by traditional partitioning routine with objective function as minimizing the cutsize, and $G$ is the additional cut percent allowed which is determined through empirical analysis; total cutsize is greater than the available number of TSVs, $N_{TSV}$ in the given architecture. The routine will stop once the condition $(\Delta R^D - P_{tol}) < P_{age} < (\Delta R^D + P_{tol})$ satisfies. Here $\Delta R^D$ is the difference between the desired distributions of layers, $P_{tol}$ is the tolerance of power that can be expected for a given circuit and $P_{age}$ is the percentage power difference among layers. This parameter can be varied according to minimum power difference that we can achieve for each and individual circuit.

5. EXPERIMENT RESULTS

Our proposed power-aware partitioning algorithm, power evaluation methodology and routing estimation methodology are implemented using C++ on Ubuntu Linux machine. We have considered a 4-layer 3D FPGA with an array size of 50 x 50 LBs and a routing channel width of 65 wiring tracks for each layer such that all the selected 14 MCNC benchmark circuits [15] will be successfully placed and routed. The share of each type of segment in the routing channel is similar to Xilinx Virtex II device. In order to observe the variation of power during the partitioning we have conducted experiments on sensitivity analysis prior to the actual work. We ran all the circuits on traditional partitioning routine with the objective of minimizing the cutsize and used our power evaluation methodology. As shown in Fig. 8, over a number of iterations, the variation of total power due to the movement of nodes and nets is significantly higher for a particular partition. And we have also observed that this variation is increasing with respect to the increase in the circuit size.

In the next phase of experiments, we have compared our power values estimated at the partitioning stage against the values obtained by running TPR [4] on 14 MCNC benchmark circuits for the validation purpose. TPR is a 3D place and route tool targeted for academic research on 3D FPGAs. It is a 3D extension to the highly successful VPR [15]. In TPR, explicit partitioning is done by hMeTIS [16]. We appended our power calculation technique to TPR and then performed placement and routing and obtained the actual length of the nets followed by our power calculations. Power values and percentage error for the individual layers of all 14 circuits are shown in Table 1. As shown, we have achieved absolute error of 12.25%, 13.47%, 14.41% and 13.43% on average for layer 1, layer 2, layer 3 and layer 4 respectively. Here we showed the absolute errors for individual layers separately instead of total array because we are targeting for power distribution among layers. It should be noted that the power is under-estimated for some circuits and over-estimated for others. This comes from the mean error difference obtained from the routing segments estimation methodology. From our observations, we believe that these under- and over-estimations can be mitigated by characterizing the placement and routing algorithms at the early stage of the design cycle which would be a part of our future work.

Figure 7. Assigning switching activities during the movement of nodes.

Figure 8. Variation of power and number of 3D nets over the partitioning iterations.
Table 1. Percentage estimation error validated against TPR

<table>
<thead>
<tr>
<th>Circuit</th>
<th>LBs</th>
<th>Nets including multi-terminal nets</th>
<th>Power at TPR output (mW)</th>
<th>Power from our estimations obtained at partitioning stage (mW)</th>
<th>Percentage Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>alu</td>
<td>1522</td>
<td>5408</td>
<td>p1: 14.2</td>
<td>240.2</td>
<td>10.8</td>
</tr>
<tr>
<td>pdc</td>
<td>1878</td>
<td>6902</td>
<td>p2: 12.3</td>
<td>129.3</td>
<td>4.5</td>
</tr>
<tr>
<td>clma</td>
<td>1707</td>
<td>6557</td>
<td>p3: 25.1</td>
<td>13.5</td>
<td>14.2</td>
</tr>
<tr>
<td>des</td>
<td>1591</td>
<td>6110</td>
<td>p4: 58.1</td>
<td>19.6</td>
<td>20.4</td>
</tr>
<tr>
<td>diffag</td>
<td>1497</td>
<td>5673</td>
<td>p1: 6.3</td>
<td>9.9</td>
<td>7.2</td>
</tr>
<tr>
<td>ellipta</td>
<td>3604</td>
<td>13756</td>
<td>p2: 15.8</td>
<td>29.5</td>
<td>12.8</td>
</tr>
<tr>
<td>frisc</td>
<td>3556</td>
<td>13658</td>
<td>p3: 61.0</td>
<td>9.5</td>
<td>35.3</td>
</tr>
<tr>
<td>pdc</td>
<td>4575</td>
<td>17193</td>
<td>p4: 12.2</td>
<td>289.6</td>
<td>13.2</td>
</tr>
<tr>
<td>s29h</td>
<td>1951</td>
<td>6959</td>
<td>s1: 62.5</td>
<td>152.2</td>
<td>74.6</td>
</tr>
<tr>
<td>x841n</td>
<td>6406</td>
<td>22807</td>
<td>s2: 234.1</td>
<td>198.7</td>
<td>22.1</td>
</tr>
<tr>
<td>x858fd</td>
<td>6447</td>
<td>22100</td>
<td>s3: 36.2</td>
<td>23.7</td>
<td>31.1</td>
</tr>
<tr>
<td>seq</td>
<td>1750</td>
<td>6193</td>
<td>s4: 98.7</td>
<td>104.1</td>
<td>46.0</td>
</tr>
<tr>
<td>spla</td>
<td>3690</td>
<td>13888</td>
<td>t1: 239.0</td>
<td>12.9</td>
<td>8.1</td>
</tr>
</tbody>
</table>

Absolute Average: 12.25 13.47 14.41 13.43

Figure 9. Non-uniform power distribution obtained from TPR with conventional partitioning.

Figure 10. Uniform power distribution obtained from (a) our proposed power-aware partitioning and (b) TPR running with power-aware partitions as input.

To investigate the performance of our power-aware partitioning algorithm, at first we ran five MCNC benchmark circuits, having different circuit sizes, using TPR with conventional partitioning routine which is targeting cutsize minimization objective. As shown in Fig. 9, non-uniform distribution of power is obtained with a variation of power from 20% (for example, between layer 2 and layer 3 for clma) to 90% (for example, between layer 1 and layer 2 for clma). These are the values obtained after successful routing. It is observable that conventional partitioning which is targeting cutsize minimization objective leads to a non-uniform power distribution among the layers. This kind of variation may result in heat problems on 3D platform. Note that in clma, the big difference for layer 1 compared to the others is coming from the high fan-out pins incurring large number of nets.

After that, we ran the same circuits on our proposed power-aware partitioning routine. The goal of this experiment is to achieve uniform/desired power distribution among four layers in considered 3D FPGA at the partitioning stage. This can be targeted to avoid high peak temperatures for thermal management at the early stage of design flow. As shown in Fig. 10(a) we have achieved around 12% power difference between layers for the proposed power-aware partitioning algorithm targeting uniform power distribution. Our results are compared against TPR shown in Fig. 10(b). These values are obtained by feeding the power-aware partitions, generated using our proposed power-aware partitioning algorithm, to TPR and running the placement and routing steps. As shown in the figure, percentage variation in power between layers is around 16% for TPR case where more accurate power evaluation is performed on the placed and routed circuit. This rather small difference shows that our power-aware partitioning is able to distribute the power among the layers such that the later stages can have less burden on resolving the issues related to power/thermal management.

Our power-aware partitioning routine is able to distribute the power among the layers according to the user requirement (e.g. when a non-uniform power density distribution is demanded depending on the location of the heat sink for the sake of thermal management). This distribution can be changed by setting the parameter $R_D$ in Eq. 4 as per the user requirement. Fig. 11(a) shows the desired power distribution with $R_D$ as 20%. It is showing that around 16% to 22% difference between the any layers is obtained from our power-aware partitioning routine. For example pdc, layer 1 is having the highest power consumption and layer 2 is having power consumption around 18% less than the power consumed by layer 1. Similarly power consumption in layer 3 is around 18% less than the power consumed by layer 2 and similarly for layer 4 also. This is an appropriate power distribution when the heat sinks are closer to the first layer. As shown in Fig. 11(b), our values are at comparable level with TPR results and our power-aware partitioning is able to distribute the power according
to the user requirement. Our results showing that the power distribution can be achieved at the early stage of the design cycle in order to consider the power/thermal issues well before the placement stage. Our methodology can be easily interfaced with TPR tool in which only performance based partitioning; placement and routing on the given design are done. We achieved the uniform/desired power distribution at a cost of cutsize increment. As shown in Fig. 6, one of the constraints given to the power-aware routine is \( N_{\text{cut}} < (N_{\text{cut,MIN}} (1+G)) \). That means, the total cutsize after each iteration must be less than \( 1+G \) times of the minimum cutsize that we can obtain from the traditional partitioning algorithm with the objective of cutsize minimization. The best suitable value for \( G \) is selected empirically such that the overall cutsize should be comparable to \( N_{\text{cut,MIN}} \).

6. CONCLUSION

We proposed a power evaluation methodology by considering the hierarchical routing resources at an early stage of the design cycle namely at the partitioning and a power-aware partitioning algorithm for 3D FPGAs. Our results show that the estimated power values at partitioning stage have acceptable deviation compared with the values obtained after doing actual placement and routing of the circuit using TPR. In addition, it is observed that the power can be distributed according to the demand produced by the user for the purpose of power/thermal management well before placement stage. We achieved uniform/desired power distribution among the layers at a cost of increased cutsize. We believe that the power-aware methodology at the early stage of design cycle can help the designers to handle multiple objectives more efficiently especially in 3D platform. The immediate future work would be extending the proposed power-aware technique to the early thermal-aware partitioning methodology for 3D FPGAs depending on the dynamic power as well as the static power consumption.

7. ACKNOWLEDGMENTS

The authors wish to thank Nobuaki Miyakawa (Honda Research Institute, Japan), Kai KJ, Ohashi Masahiro (Panasonic, Japan), Koji Inoue, Masayoshi Yoshimura and Takaki Hanada (Kyushu University, Japan), for their valuable comments on the work.

8. REFERENCES