

Variation Aware Placement in FPGAs

Abstract—Technology scaling has adverse effects on the different process parameters in contemporary digital circuits. Such variations tend to significantly modulate the delay and the power characteristics of the logic blocks in FPGAs which impact both the total power consumed and the operating frequencies. One of the ways to counter such problems is appropriate tuning of process parameters statically to reduce the impact of variations. However, such techniques still leave the FPGA with blocks of variable leakage power consumption and delays, which continue to have an impact on the power consumption and operating frequencies. In this paper we demonstrate the impact of such variations in process parameters of FPGAs and propose two different placement techniques to nullify those effects. Our approach provides leakage benefits close to 14% on an average over different benchmark designs.

I. INTRODUCTION

Impact of variations in different process parameters like, gate length, threshold voltage, oxide thickness etc. have been discussed in different components of digital circuits extensively in recent times.

The various manufacturing effects on different process parameters have been demonstrated in [1] and [2]. Degradation in the operating frequencies and increase in leakage power consumption have been demonstrated in FPGAs in [3]. The paper demonstrated a 3x increase in the leakage power consumption and nearly 2X increase in the delay due to process variations. It proposed a static architecture and device tuning approach to adjust the nominal operating conditions to improve the leakage and timing yield of the device. Although static tuning of device parameters may counter such impacts to some extent, there are still variations that continue to impact the performance and power consumed by the chip. Consequently, such variations call for changes in the existing design tools and incorporation of variation awareness in those tools. In this paper, we propose a variation aware placement scheme in FPGAs and demonstrate the effectiveness of the scheme on Xilinx FPGAs and regular island style FPGAs.

II. PROCESS VARIATION IMPACTS ON FPGAS

Variations in gate length, effective channel widths and the resultant effects on the threshold can affect both delay and power consumption of the individual slices on FPGAs. To provide a comprehensive analysis we breakdown our analysis into the impact on memories and LUTs separately. Each of the components were laid out using micromagic and simulated using HSPICE, using the model files from BSIM4 for 65nm technology.

Since SRAM cells are not present in the critical path while estimating the frequency of operation of any design on FPGAs, their performance and latencies do not affect the frequency of operation of the FPGAs. Moreover, most FPGA technologies typically tend to use high threshold gates in designing such SRAM cells to minimize the leakage power consumed the impact of the leakage power consumed by the FPGAs is minimal as well. Such an observation directs all our focus towards analyzing and optimizing the impact of variations of process parameters on the LUT multiplexers as discussed in the next section.

Since the LUT multiplexers are in the critical path they may affect the frequency of operation of FPGAs. The delay varies significantly, in fact nearly by a factor of 1.3X, even with 20% variation in the threshold voltage from the nominal voltage of 0.2V. The leakage power consumed by the the LUT multiplexer is plotted with threshold in figure ???. Once again there is a variation of nearly 2X for 20% variation in threshold voltages of the transistors around the nominal threshold. It is however important to note that while the leakage power consumption increases by nearly 2X with 20% decrease in the threshold voltage, it does not change much with the increase in the threshold voltage.

To determine the impact on the delay and the power consumed by the blocks, we need the delay and leakage profiling of the SLICES statically, before the implementation of any design on them. Programming leakage sensors on FPGAs may not be feasible to employ due to their analog nature. A digital ring oscillator based delay sensor, which may be configured at different parts of the FPGA separately to obtain the delays of different blocks, may be used to get an estimate of the delay category in which the block falls into. To avoid any overhead such delay estimator could be used before loading the configuration of the design onto the FPGA, and removed thereafter.

III. PLACEMENT METHODOLOGY

We propose two different schemes for modifying the placement algorithm to incorporate variation tolerance. The first approach is a discarding scheme, where the placement algorithm discards the blocks, which are heavily affected by process variation. The second scheme is based on modification of the placement costs in any placement algorithm to accommodate the effect of variation in the logic blocks.

A. Block Discarding Policy (BDP)

We provide some hard constraints to the Xilinx placement algorithm, to skip certain blocks which may have adverse effects on the delay and the leakage power consumption of the design. Since the utilization of the logic blocks is not typically 100% in most designs, this leverages an opportunity to impose some placement constraints. The constraint to skip any block is imposed by adding the *PROHIBIT* constraint in the User Constraint File (UCF) file. Due to process variations in the LUTs of different blocks, the delay and the leakage power are conflicting in their trends, which makes it hard to decide on the blocks to be skipped. Consequently, the decision to skip a block should be based on both the leakage power consumption and the delay of any slice in the FPGA. We set a threshold value (L,D) for the leakage power consumption and delay of a SLICE respectively, above which the SLICE is considered for discarding. We vary this threshold based on the flexibility in the design, keeping in mind the utilization of the device since it affects the number of blocks to be skipped. Such a constraint file is generated as a text output from a perl program, which determines the blocks to be skipped, based on the leakage and delay thresholds.

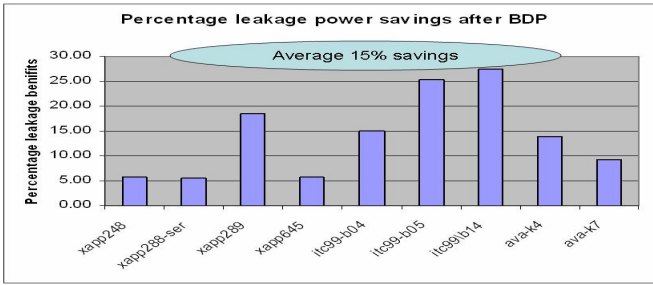


Fig. 1. Leakage power saving obtained using BDP

B. Variation Aware Placement

The block discard placement (BDP) scheme imposes a hard constraint and requires iterative determination of the number of blocks to be discarded based on the flexibility in the design. However, the variation aware placement scheme imposes a soft constraint that is incorporated by modifying the placement algorithm itself. We modified the placement algorithm in Versatile Place and Route tool (VPR) to incorporate the impact of new delays and leakage power consumed by the blocks selected for placing the design. At first, we associate the exact delay numbers with each of the blocks in VPR (termed as sub-blocks in VPR), to get a precise estimate of the critical path, which automatically ensures the incorporation of the variation in delay of each of the blocks in our algorithm. VPR's placement algorithm works on the principle of simulated annealing and therefore at each step, while jumping from one solution to another, it has a cost function to evaluate the current solution. Therefore, in order to incorporate the leakage power of the blocks, we explicitly add a normalized leakage cost factor to the total cost of each of the blocks as estimated by the existing placement algorithm.

IV. EXPERIMENTAL SETUP AND RESULTS

The leakage and delay numbers for different threshold voltages of the transistors were generated using simulations explained in II. To implement the BDP we generated the Gaussian distribution of threshold voltages using a library provided by GNU and obtained the leakage and delay of each of the SLICES from SPICE simulations. The final threshold values (L,D) mentioned in section III, are obtained when the design had the minimum possible leakage power consumption and placement and routing tools do not fail due to over constraining. The variation aware placement algorithm was implemented in VPR for island style FPGA devices as described in section III.

To demonstrate the effectiveness of the block discarding policy, we implemented various designs using the Xilinx ISE tool flow(ver 6.0), provided by Xilinx. There is an average increase in the total leakage power consumed by nearly 20% over different benchmarks due to process variations. Note that the leakage power consumption is always going to increase with any process variations, since the percentage increase in the leakage power of a single LUT is much higher than the decrease for the same deviation from its nominal threshold. We picked up the benchmarks with device utilizations lower than 90%, the reason being the fact that our whole algorithm is based on the flexibility in discarding the bad blocks while placing the cells. After applying the BDP based placement appropriately, nearly 14% leakage savings were obtained as demonstrated in figure 1.

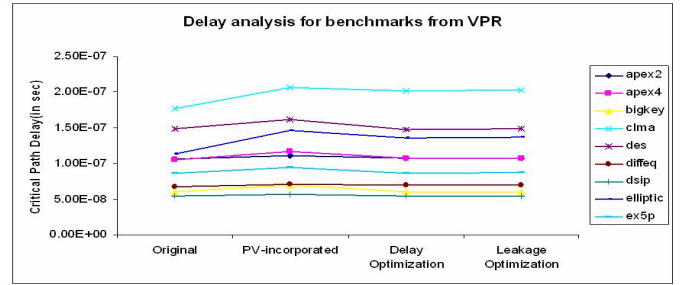


Fig. 2. Impact on clock frequencies of different implementations placed using VPR

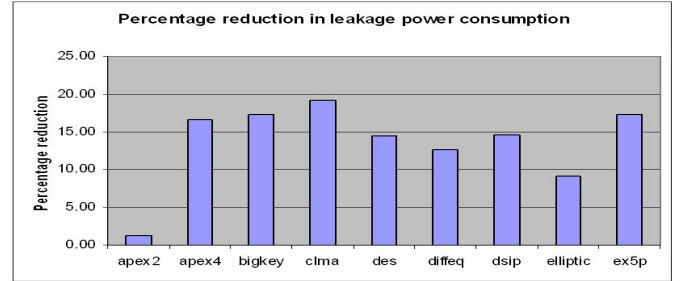


Fig. 3. Leakage savings using the variation aware placement algorithm

The variation awareness incorporated into the placement algorithm of VPR was tested over a set of blif benchmarks. We compare the operating frequencies of four different simulations for each benchmark in figure 2. The first label in the X-axis of the graph in figure 2 shows the original cycle time of the design after placement. The next simulation marked as pv-incorporated, is the frequency of operation of the design implemented on the FPGA, once the new delay values due to process variations are incorporated. In this simulation the delays are associated to the blocks after placement and therefore it provides an estimate on the impact of process variations on operating frequencies. Once again it is evident from the figure that process variations no longer allow the designs to operate at their original frequencies but at a lower frequency on an average by 12%. The third simulation shows the cycle time obtained from timing driven placement algorithm of VPR without incorporating the leakage optimization while the final simulation is for the variation aware placement scheme with the leakage power optimization incorporated. With the new delays incorporated into the blocks, the placement algorithm finds a better placement which allows to operate the design at a frequency on an average 10% higher than the increased frequency due to process variations. Finally, leakage benefits are obtained without compromising on the operating frequencies by choosing low leakage paths in the non critical nets. The leakage benefits are demonstrated in 3. It can be observed from figure 2 that the operating frequency is either not affected or very negligibly affected on applying the leakage optimization in the placement algorithm. This is mainly due to the fact that the placement still prioritizes the operating frequency and only operates for leakage optimizations in the non-critical nets. We observe an average leakage power savings of 14%, which is similar to the savings obtained in Xilinx implementation of BDP.

REFERENCES

- [1] S. Borkar, T. Karnik and V. De “Design and reliability challenges in nanometer technologies” *Proceedings of Design Automation Conference (DAC)*, 2004.
- [2] S. Nassif “Modeling and analysis of manufacturing variations” *Proceedings of IEEE Custom Integrated Circuits Conference*, 2001.
- [3] H. Yan, P. Wong, L. Cheng, Y. Lin and L. He “FPGA device and architecture evaluation considering process variation ” *Proceedings of International Conference on Computer Aided Design*, 2005.