Parametric Yield Driven Resource Binding in Behavioral Synthesis with Multi-V_{th}/V_{dd} Library

Yibo Chen, Yuan Xie
Pennsylvania State University
University Park, PA 16802
{yxc236, yuanxie}@cse.psu.edu

Yu Wang
Tsinghua University
Beijing, China 100084
yu-wang@tsinghua.edu.cn

Andres Takach
Mentor Graphics Corporation
Wilsonville, OR 97070
andres_takach@mentor.com

***Abstract*—The ever-increasing chip power dissipation in SoCs has imposed great challenges on today's circuit design. It has been shown that multiple threshold and supply voltages assignment (multi-V_{th}/V_{dd}) is an effective way to reduce power dissipation. However, most of the prior multi-V_{th}/V_{dd} optimizations are performed under deterministic conditions. With the increasing process variability that has significant impact on both the power dissipation and performance of circuit designs, it is necessary to employ statistical approaches in analysis and optimizations for low power. This paper studies the impact of process variations on the multi-V_{th}/V_{dd} technique at the behavioral synthesis level. A multi-V_{th}/V_{dd} resource library is characterized for delay and power variations at different voltage combinations. A parametric yield-driven resource binding algorithm is then proposed, which uses the characterized power and delay distributions and efficiently maximizes power yield under a timing yield constraint. During the resource binding process, voltage level converters are inserted between resources when required. Experimental results show that significant power reduction can be achieved with the proposed variation-aware framework, compared with traditional worst-case based deterministic approaches.**

I. Introduction

Integrating billions of transistors on a single chip with nano-scale transistors has resulted in great challenges for chip designers. One of these challenges is that the pace of productivity gains has not kept up to address the increases in design complexity. Consequently, we have seen a recent trend of moving design abstraction to a higher level, with an emphasis on **Electronic System Level (ESL)** design methodologies. A very important component of ESL is raising the level of abstraction of hardware design. High-level synthesis (HLS) provides this component by providing automation to generate optimized hardware from a high-level description of the function or algorithm to be implemented in hardware. HLS generates a cycle-accurate specification at the register-transfer level (RTL) that is then used in existing ASIC or FPGA design methodologies. Commercial high-level synthesis tools [1] have recently gained a lot of attention as evidenced in recent conference HLS workshops (DATE2008, DAC2008, ASPDAC2009), conference panels and publications that track the industry.

Power consumption and process variability are among other critical design challenges as technology scales. While it is believed that tackling these issues at a higher level of the design hierarchy can lead to better design decisions, a lot of work has been done on low-power high-level synthesis [2]–[4] as well as process-variation aware high-level synthesis [5]–[8]. These techniques have been successfully implemented but most of the existing work focuses on one side of the issues in isolation. Recently, Srivastava et al. [9] explore the multi-$V_{th}/V_{dd}/T_{ox}$ design space with the consideration of process variations at the gate level. Nevertheless, variation-aware low power exploration for behavioral synthesis is still in its infancy.

Multiple threshold and supply voltages assignment (multi-V_{th}/V_{dd}) has been shown as an effective way to reduce circuit power dissipation [2], [3], [10], [11]. Existing approaches assign circuit components on critical paths to operate at a higher V_{dd} or lower V_{th}, and non-critical portions of the circuit are made to operate at lower V_{dd} or higher V_{th}, respectively. The total power consumption is thus reduced without degrading circuit performance. However, nowadays circuit performance is affected by process variations. If the variations are under-estimated, for example, using nominal delays of circuit components to guide the design, non-critical components may turn to critical ones due to the variations, and circuit timing constraints may be violated. On the other hand, in existing corner-based worst-case analysis, variations are over-estimated resulting in design specs that are hard to meet, and this consequently increases design effort and degrades circuit performance.

This paper presents a variation-aware power optimization framework in high-level synthesis using simultaneous multi-V_{th}/V_{dd} assignments. Firstly, the impact of parameter variations on the delay and power of circuit components, is explored at different operating points of threshold and supply voltages. A variation-characterized resource library containing the parameters of delay and power distributions at different voltage "corners", is built once for the given technology, so that is available for high-level synthesis to query the delay/power characteristics of resources. The concept of *Parametric Yield*, which is defined as the probability that the design meets specified constraints such as delay or power constraints, is then introduced to guide design space exploration. Statistical timing and power analysis on the data flow graph (DFG) is used to populate the delay and power distributions through the DFG, and to estimate the overall performance and power yield of the entire design. A variation-aware resource binding algorithm is then proposed to maximize power yield under a timing yield constraint, by iteratively searching for the operations that have the maximum potential of performance/power yield improvement, and replacing them with better candidates in the multi-V_{th}/V_{dd} resource library. During the resource binding process, voltage level converters are inserted for chaining of resource units having different V_{dd} supplies.

The contribution of this paper can be summarized as:

- First, this is the first work to apply multi-V_{th}/V_{dd} techniques during high-level synthesis under the context of both delay and power variations. A flow for variation-aware power optimization in multi-V_{th}/V_{dd} HLS is proposed. This flow includes library characterization, statistical timing and power analysis methodologies for HLS, and resource binding optimization with variation-characterized multi-V_{th}/V_{dd} library.
- Voltage level conversion is explored during the resource binding in high-level synthesis, enabling the full utilization of multi-V_{dd} components.

II. Related Work

Prior research work tightly related to this paper mainly falls into two categories: 1) Low power high-level synthesis using multi-V_{th}

This work was supported in part by NSF CAREER 0643902, NSF 0720659, NSFC 60870001, and a grant from SRC.

978-1-4244-5767-0/10/$26.00 2010 IEEE

or multi-V_{dd}; 2) Process variation aware high-level synthesis.

Shiue and Chakrabarti [2] proposed low-power scheduling schemes with multi-V_{dd} resources by maximizing the utilization of resources operating at reduced supply voltages. Khouri et al. [3] performed high-level synthesis using a dual-V_{th} library for leakage power reduction. Tang et al. [4] formulated the synthesis problem using dual-V_{th} as a maximum weight independent set (MWIS) problem, within which near-optimal leakage power reduction is achieved with greatly reduced run time. Very recently, Shin et al. explored optimal register allocation for for high-level synthesis using dual supply voltages [12]. However, all of these techniques were applied under deterministic conditions without taking process variation into consideration.

Process variation aware high-level synthesis has recently gained much attention. Jung et al. [6] proposed a timing yield aware HLS algorithm to improve resource sharing and reduce overall latency. Gregory et al. [8] integrated timing-driven floorplanning into the variation-aware high-level design. Mohanty et al.'s work [13] took into account the leakage power variations in low-power high-level synthesis, however, the major difference between [13] and our work is that, the delay variation of function units was not considered in [13], so the timing analysis during synthesis was still deterministic. Recently, Wang et al. [14] proposed a joint design-time optimization and post-silicon tuning framework that tackles both timing and power variations. Adaptive body biasing (ABB) was applied to function units to reduce leakage power and improve power yield.

III. Multi-V_{th}/V_{dd} Library Characterization Under Process Variations

This section presents the characterization of the variation-aware multi-V_{th}/V_{dd} resource library, including the delay and power characterization flow and the selection of dual threshold and supply voltages.

A. Variation-aware Library Characterization Flow

In order to fully explore the design space, a diverse library of functional units is built for high-level synthesis. Each of the units can be characterized as a $(delay, power)$ pair. Under the influence of process variations, the delay and power of function units are no longer fixed values, but spread to probabilistic distributions. Therefore, the characterization of function units with delay and power variations requires statistical analysis methodologies.

This paper uses a commercial gate-level statistical timing analysis tool, Synopsys PrimeTime *VX* [15] to perform the characterization. This variation-aware tool increases the accuracy of timing analysis by considering the statistical distribution of characterized parameters, such as channel length, metal-line width and oxide thickness.Validation against SPICE Monte Carlo statistical analysisshows that PrimeTime *VX* analysis holds the similar accuracy but reduces the running-time significantly.

The characterization flow takes as input the statistical distributions of process parameters (channel length, oxide thickness, etc.) and generates the statistical distributions of delay an power for each resource in the library. To characterize the delay of function units under the impact of process variations, the following steps are performed:

1) All the standard cells in a technology library are characterized using variation-aware analysis, and the results including parameters of cell delay distributions, are collected to build a variation-aware technology library. In this paper, we use NCSU FreePDK 45nm technology library [16] for all the characterization and experiments.
2) The function units used in HLS are then synthesized and linked to the variation-aware technology library.
3) Statistical timing analysis for the function units is performed using PrimeTime *VX*, and the parameters of delay distributions are reported.

Statistical power characterization for function units in the resource library can be done using Monte Carlo analysis in SPICE.The power consumption of function units consists of dynamic and leakage components. While dynamic power is relatively immune to process variation, leakage power is greatly affected and becomes dominant as technology continues scaling down [17]. Therefore, in this paper only leakage power is characterized using statistical analysis. However, this doesn't mean considerations for dynamic power can be omitted. In fact, dynamic power optimization in high-level synthesis has been a well explored topic [1]. Our variation-oriented work emphasizing leakage power optimization can be stacked on or integrated into existing power optimization approaches in high-level synthesis, to further reduce the total power consumption of circuits.

The power characterization flow is stated as follows. Process variations are set in the MOS model files, and 1000 runs of Monte Carlo iterations are performed for each library cell. After the characterization, the parameters of the leakage power distributions of library cells are reported.

B. Multi-V_{th}/V_{dd} Library Characterization

According to Berkeley short-channel BSIM4 model [18], higher threshold voltages can lead to exponential reduction in leakage power. Meanwhile, lower supply voltages will result in a quadratic reduction in switching power and roughly a cubic reduction in leakage power [19]. Previous implementations using multiple threshold and supply voltages in conjunction have shown a very effective reduction in both dynamic and leakage power [11]. Therefore, our approach considers the combination of dual threshold and dual supply voltages, and characterizations are performed at the four "corners" of voltage settings, namely $(V_{th}^L, V_{dd}^H), (V_{th}^H, V_{dd}^H), (V_{th}^L, V_{dd}^L), (V_{th}^H, V_{dd}^L)$, where (V_{th}^L, V_{dd}^H) is the nominal case and the other three are low-power settings. Note that although only 4 voltage settings are discussed in this paper, it is natural to extend the approach presented here to deal with more voltage settings. To reduce the process technology cost, in this paper, the multi-V_{th}/V_{dd} techniques are applied at the granularity of function units. That means, all the gates inside a function unit operate at the same threshold and supply voltages. Voltages only differ from function units to function units.

The selection of appropriate values of threshold and supply voltages for power minimization has been discussed under deterministic conditions [11]. While the empirical models in [11] are validated on actual circuit benchmarks [19], they may not be accurate under the impact of process variations. A refined model taking into account the process variations is presented in [9], and this guides the optimal value selection in this work.

The characterization results (which will be further discussed in Section VI) show that, power reduction is always achieved at the cost of delay penalties. Moreover, larger delay variations are observed for slower units operating at high-V_{th} or low-V_{dd}, which means larger probability of timing violations when they are placed on the near-critical paths. This further demonstrates the necessity of statistical analysis and parametric yield-driven optimization approaches.

IV. Yield Analysis in Statistical High-Level Synthesis

In this section, a parametric yield analysis framework for statistical HLS is presented. We first show the necessity of statistical analysis by a simple motivational example, and then demonstrate the statistical

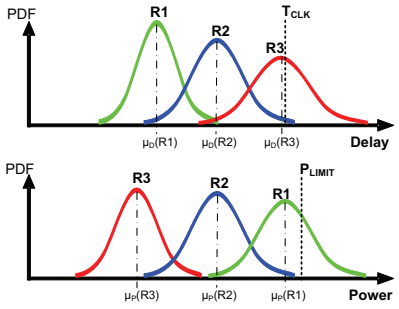

Fig. 1. Motivating example of yield-driven synthesis.

timing and power analysis for HLS, as well as the modeling and integration of level converters for multi-V_{dd} HLS.

A. Parametric Yield

To bring the process-variation awareness to the high-level synthesis flow, we first introduce a new metric called *Parametric Yield*. The parametric yield is defined as the probability of the synthesized hardware meeting a specified constraint $Yield = P(Y \leq Y_{max})$, where Y can be delay or power.

Fig. 1 shows a motivational example of yield-aware analysis. Three resource units $R1$, $R2$ and $R3$ have the same circuit implementation but operate at different supply or threshold voltages. Fig. 1 shows the delay an power distributions for $R1$, $R2$, $R3$. In this case the mean power follows up $\mu_P(R3) < \mu_P(R2) < \mu_P(R1)$ and the mean delay follows $\mu_D(R1) < \mu_D(R2) < \mu_D(R3)$ which is as expected since power reduction usually comes at the cost of increased delay. The clock cycle time T_{CLK} and the power consumption constraint P_{LMT} (e.g., the TDP (thermal design power) of most modern microprocessors) are also shown on the figure. If the variation is disregarded and nominal-case analysis is used, any of the resource units can be chosen since they all meet timing. In this case $R3$ would be chosen as it has the lowest power consumption. However, under a statistical point of view, $R3$ has a low timing yield (approximately 50%) and is very likely to cause timing violations. In contrast, with corner-based worst-case analysis only $R1$ can be chosen under the clock cycle time constraint (the worst-case delay of $R2$ slightly violates the limit), whereas $R1$ has a poor power yield. In fact, if we set a timing yield constraint instead of enforcing the worst-case delay limitation, $R2$ can be chosen with a slight timing yield loss but a well balanced delay and power tradeoff. Therefore, a yield-driven statistical approach is needed for exploring the design space to maximize one parametric yield under other parametric yield constraints.

B. Statistical Timing and Power Analysis for HLS

High-level synthesis (HLS) is the process of transforming a behavioral description into register level structure description. Operations such as additions and multiplications in the DFG are scheduled into control steps. During the resource allocation and binding stages, operations are bound to corresponding function units in the resource library meeting type and latency requirements.

Given the clock cycle time T_{CLK}, the timing yield of the entire DFG, $Yield_T$ is defined as:

$$Yield_T = P(T_1 \leq T_{CLK}, T_2 \leq T_{CLK}, \ldots, T_n \leq T_{CLK}) \quad (1)$$

where $P()$ is the probability function, $T_1, T_2, \ldots, T_n$ are the arrival time distributions at control step $1, 2, \ldots, n$, respectively.

The arriving time distribution of each clock cycle can be computed from the delay distributions of function units bound at that cycle. Two operations, sum and max, are used to compute the distributions:

- sum operation is used when two function units are chained in cascade within a clock cycle, as shown in $CC1$ and $CC2$ of Fig. 2. The total delay can be computed as the "sum" of their delay distributions (normal distribution assumed);
- max operation is used when the outputs of two or more units are fed to another function unit at the same clock cycle, as shown in $CC1$ of Fig. 2. The "maximum" delay distribution can be computed out of the contributing distributions using tightness probability and moment matching [14].

With these two operations, the arriving time distribution of each clock cycle is computed, and the overall timing yield of the DFG is obtained using Equation (1).

The total power consumption of a DFG can be computed as the sum of the power consumptions of all the function units used in the DFG. Given a power limitation P_{LMT}, the power yield of the DFG $Yield_P$ is computed as the probability that total power P_{DFG} is less than the requirement, as expressed in Equation (2).

$$Yield_P = P(P_{DFG} \leq P_{LMT}) \quad (2)$$

Since dynamic power is relatively immune to process variations, it is regarded as a constant portion which only affects the mean value of the total power consumption. Therefore, the total power is still normally distributed, although statistical analysis is only applied to the leakage power. As aforementioned in Section III, our proposed yield-driven statistical framework can be stacked on existing approaches for dynamic power optimization, to further reduce the total power consumption of circuits.

C. Voltage Level Conversion in HLS

In designs using multi-V_{dd} resource units, voltage level convertors are required when a low-voltage resource unit is driving a high-voltage resource unit. Level conversion can be performed either synchronously or asynchronously. Synchronous level conversion is usually embedded in flip-flops and occurs at the active clock edge, while asynchronous level converters can be inserted anywhere within the combinational logic block.

When process variations are considered, asynchronous level converters are even more favorable, because they are not bounded by clock edges, and timing slacks can be passed through the converters. Therefore, time borrowing can happen between low-voltage and high-voltage resource units. As slow function units (due to variations) may get more time to finish execution, the timing yield can be improved and the impact of process variations is consequently reduced.

While many fast and low-power level conversion circuits have been proposed recently, this paper uses the multi-V_{th} level converter presented in [20], taking the advantage that there is no extra process technology overhead for multi-V_{th} level converters, since multi-V_{th} is already deployed for function units. The proposed level converter is composed of two dual-V_{th} cascaded inverters. Its delay and power are then characterized in HSPICE using the listed parameters [20].

The delay penalty of a level converter can be accounted by summing its delay with the delay of the function unit it is associated to. The power penalty can be addressed by counting the level converters used in the DFG and adding the corresponding power to the total power consumption.

V. Yield-Driven Power Optimization Algorithm

In this section we propose our yield-driven power optimization framework based on the aforementioned statistical timing and power yield analysis. During the high-level synthesis design loop, resource binding selects the optimal resource instances in the resource library and binds them to the scheduled operations at each control step.

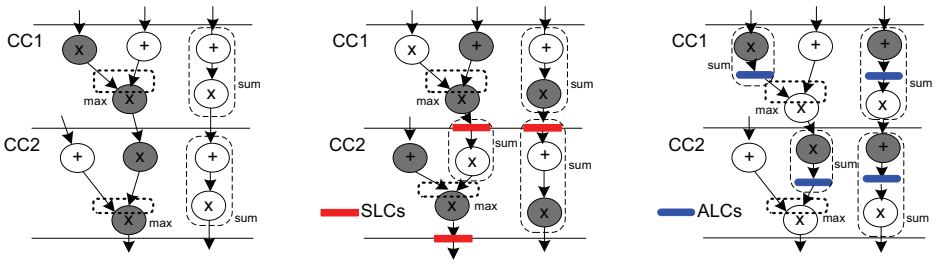

Fig. 2. Timing yield computation in multi-V_{dd} high-level synthesis with different level conversions. Shadowed operations are bound to function units with low supply voltages, and the bars indicate the insertion of level converters.

A variation-aware resource binding algorithm is then proposed to maximize power yield under a preset timing yield constraint, by iteratively searching for the operations with the maximum potential of timing/power yield improvement, and replacing them with better candidates in the multi-V_{th}/V_{dd} resource library.

A. Variation-aware Resource Binding Algorithm Overview

Our variation-aware resource binding algorithm takes a search strategy called *variable depth search* [14], [21], [22] to iteratively improve the power yield under performance constraints. The outline of the algorithm is shown in Fig. 3, where a DFG is initially scheduled and bound to resource library with nominal voltages (V_{th}^L, V_{dd}^H). A lower-bound constraint on the timing yield is set, so that the probability of the design can operate at a given clock frequency, will be larger than or equal to a preset threshold (e.g., 95%). In the algorithm, a move is defined as a local and incremental change on the resource bindings. As shown in the sub-routine GENMOVE in Fig. 3, the algorithm generates a set of moves, and finds out a sequence of moves that maximizes the accumulated $gain$, which is defined as $\alpha * \Delta Yield_D + \Delta Yield_P$, where α is a weighting factor to balance the weights of timing and power yield improvements. The optimal sequence of moves is then applied to the DFG, and the timing and power yields of the DFG are updated before the next iteration. The iterative search ends when there is no yield improvement or the timing yield constraint is violated.

VABINDING($DFG, ResLib, Constraints, LCStrategy$)
▷ Initialization
1 Scheduling using ASAP strategy
2 Initial Binding to (V_{th}^L, V_{dd}^H) resources
▷ Variation-aware resource binding
3 **while** $\Delta Yield_P > 0$ AND $Yield_D \geq Constraint$
4 **do for** $i \leftarrow 1$ to $MAXMOVES$
5 **do** $Gain_i \leftarrow$ GENMOVE($DFG, ResLib, LCStrategy$)
6 Append $Gain_i$ to $Gain\_List$;
7 Find subsequence $Gain_1, \ldots, Gain_k$ in $Gain\_List$ so that $G = \sum_{i=1}^{k} Gain_i$ is maximized
8 **if** $G > 0$
9 **do** Accept moves $1 \ldots k$
10 Evaluate $\Delta Yield_P$ and $Yield_D$

GENMOVE($DEG, ResLib, LCStrategy$)
1 MOVE: Choose a move using steepest descent heuristic [21]
2 Check whether and where level conversion is needed
3 **if** $LCStrategy = Avoidance$ AND $NeedConversion$
4 **do goto** MOVE
5 **if** $LCStrategy = Synchronous$ AND $NeedConversion$
▷ Check whether conversion is synchronous or not
6 **do if** Conversion is inside operation chaining
7 **do goto** MOVE
8 Count the overhead of level conversion
9 Evaluate the $Gain$ of this move
10 Return $Gain$

Fig. 3. Outline of the variation-aware resource binding algorithm

B. Voltage Level Conversion Strategies

Moves during the iterative search may result in low-voltage resource units driving high-voltage resource units. Therefore, level conversion is needed during resource binding. However, if resources are selected and bound so that low-voltage resource units never drive high-voltage ones, level conversion will not be necessary and the delay and power overheads brought by level converters can be avoided. This reduces the flexibility of resource binding for multi-voltage module combinations, and may consequently decrease the attainable yield improvement. The trade-off in this conversion-avoidance strategy, can be explored and evaluated within our proposed power optimization algorithm.

We also incorporate other two strategies of level conversions in the power optimization algorithm for comparison. All the three strategies are listed as follows:

- **Level Conversion Avoidance**. Resource binding is performed with the objective that low-voltage resources never drive high-voltage ones. As shown in Fig 2(a), no dark-to-light transition between operations is allowed (while dark operations are bound to low-V_{dd} units), so that level conversion is avoided. This is the most conservative strategy.
- **Synchronous Level Conversion**. Voltage level conversion is done synchronously in the level-converting flip-flops (SLCs). As shown in Fig. 2(b), the dark-to-light transition only happens at the beginning of each clock cycles. The flip-flop structure proposed in [23] is claimed to have smaller delay than the combination of an asynchronous converter and a conventional flip-flop. However, as discussed previously, synchronous level conversion may reduce the flexibility of resource binding as well as the possibility of timing borrowing. The effectiveness of this strategy is to be explored by the optimization algorithm.
- **Asynchronous Level Conversion**. Asynchronous level converters (ALCs) are inserted wherever level conversion is needed, as dark-to-light transition can happen anywhere in Fig. 2. This aggressive strategy provides the maximum flexibility for resource binding and timing borrowing. Although it brings in delay and power overhead, it still has great potential for timing yield improvement.

C. Moves Used in the Iterative Search

In order to fully explore the design space, three types of moves are used in the iterative search for resource binding:

- **Resource Rebinding**. In this move, an operation is assigned to a different function unit in the library with different timing and power characteristics. The key benefit of the multi-V_{th}/V_{dd} techniques is that it provides an enlarged design space for exploration, and optimal improvements are more likely to be obtained.

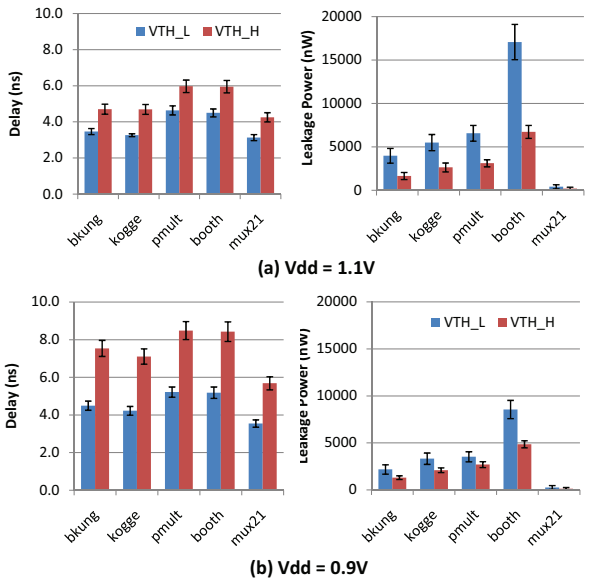

Fig. 4. Delay and leakage power characterization of function units with multi-V_{th}/V_{dd} and variation awareness.

- **Resource Sharing**. In this move, two function units that are originally bound to different function units, are now merged to share the same function unit. The type of move reduces the resource usage and consequently improves the power yield.
- **Resource Splitting**. In this move, the operation that originally shared function unit with other operations, is split from the shared function unit. This type of move might lead to other moves such as resource rebinding and resource sharing.

After each move, the algorithm checks where the low supply-voltage function units are used, and decides whether to insert or remove the level converters, according to the predefined level conversion strategy. If a move is against the strategy, it is revoked and new moves are generated until a qualifying move is found.

D. Algorithm Analysis

It has to be noted that, in the procedure GENMOVE shown in Fig. 3, even though the returned $Gain$ might be negative, it still could be accepted. Since the sequence of a cumulative positive gain is considered, the negative gains help the algorithm escape from local minima through hill-climbing.

As for the computational complexity, it is generally not possible to give non-trivial upper bounds of run time for local search algorithms [21]. However, for variable depth search in general graph partitioning, Fiduccia and Mattheyses [21] found a near-optimal growth rate of run time to be $O(n \log n)$, where n is the number of nodes in the graph. In our proposed algorithm, the timing and power yield evaluation, as well as the level converter insertion, are performed at each move. Since the yield can be updated using a gradient computation approach [14], the run time for each move is at most $O(n)$. Therefore, the overall run time for the proposed resource binding algorithm is $O(n^2 \log n)$.

VI. EXPERIMENTAL RESULTS

In this section, we present the experimental results of our variation-aware power optimization framework for high-level synthesis. The results show that our method can effectively improve the overall power yield of given designs and reduce the impact of process variations.

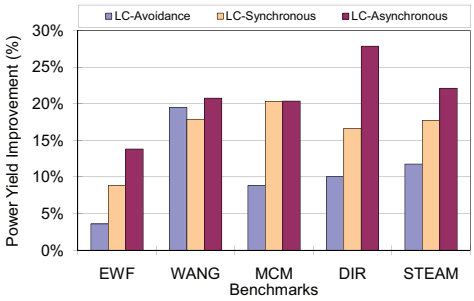

Fig. 5. Power yield improvement against deterministic worst-case approach with different level conversion strategies and timing yield constraint of 95%.

We first show the variation-aware delay and power characterization of function units. The characterization is based on NCSU FreePDK 45nm technology library [16]. Variations on two device parameters, *channel length* and *oxide thickness*, are set with relative deviations (σ/mu) to be 5%, respectively. The voltage corners for the characterization are set as: $V_{th}^L = 0.37V$, $V_{th}^H = 0.56V$, $V_{dd}^L = 0.9V$, $V_{dd}^H = 1.1V$. The characterization results for five function units, including two 16-bit adders *bkung* and *kogge*, two 8-bit×8-bit multipliers *pmult* and *booth*, and one 16-bit multiplexer *mux21*, are depicted in Fig. 4. In the figures, the color bars show the nominal case values while the error-bars show the deviations. It is clearly shown that with lower V_{dd} and/or higher V_{th}, significant power reductions are achieved at the cost of delay penalty.

With the variation-aware multi-V_{th}/V_{dd} resource library characterized, our proposed resource binding algorithm is applied on a set of industrial high-level synthesis benchmarks. A total power limitation P_{LMT} is set for each benchmark to evaluate the power yield improvement. The dynamic power consumption of function units is estimated by Synopsys *Design Compiler* with multi-V_{th}/V_{dd} technology libraries generated by *Liberty NCX*. In this work with FreePDK 45nm technology, the dynamic power is about 2 times of the mean leakage power. The power yield before and after the improvement, is then computed using Equation (2) in Section IV-B. The proposed resource binding algorithm is implemented in C++ and experiments are conducted on a Linux workstation with Intel Xeon 3.2GHz processor and 2GB RAM. All the experiments run in less than 60s of CPU time.

We compare our variation-aware resource binding algorithm against the traditional deterministic approach, which uses the worst-case (μ+3σ) delay values of function units in the multi-V_{th}/V_{dd} library to guide the resource binding. Worst-cased based approach will naturally lead to 100% timing yield, however, the power yield is poor as shown in the motivational example in Fig. 1. In contrast, our yield-aware statistical optimization algorithm takes the delay and power distributions as inputs, explores the design space with the guidance of *YieldGain*, and iteratively improves the power yield under a slight timing yield loss. The comparison results are shown in Fig. 5-7, respectively.

Fig. 5 shows the power yield improvement against worst-case delay based approach, with different level conversion strategies. A fixed timing yield constraint of 95% is set for the proposed variation-aware algorithm.The usage of function units and level converters under the three listed conversion strategies (conversion avoidance, synchronous conversion and asynchronous conversion) is listed in Table I, in which "# Vdd-H FUs" and "# Vdd-L FUs" show the numbers of function units with high/low supply voltages, respectively, and "# LCs" counts the number of converters used in the design. The last column counts the total power overhead of the asynchronous level converters. The average power yield improvements for the three

TABLE I
THE USAGE OF FUNCTION UNITS AND LEVEL CONVERTERS WITH DIFFERENT LEVEL CONVERSION STRATEGIES

Bench Name	LC-Avoidance		LC-Synchronous			LC-Asynchronous			
	# Vdd-H FUs	# Vdd-L FUs	# Vdd-H FUs	# Vdd-L FUs	# LCs	# Vdd-H FUs	# Vdd-L FUs	# LCs	LCs Overhead
EWF	6	1	4	3	1	3	4	2	3.7%
WANG	9	2	8	3	2	5	6	3	3.4%
MCM	20	4	16	8	4	15	9	5	2.4%
DIR	28	4	20	12	8	14	18	12	4.8%
STEAM	34	8	25	19	11	21	23	13	5.0%

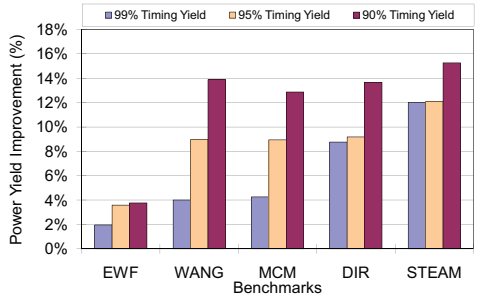

Fig. 6. Power yield improvement against deterministic worst-case approach with multi-V_{th} only and different timing yield constraints.

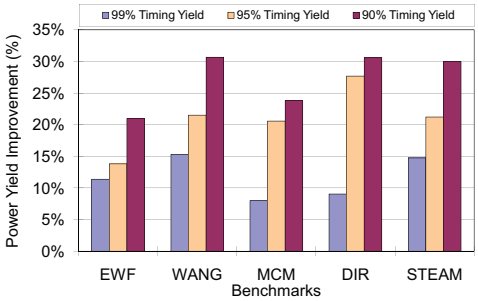

Fig. 7. Power yield improvement against deterministic worst-case approach with asynchronous level conversion and different timing yield constraints.

strategies are 10.3%, 16.7% and 21.5%, respectively. From Fig. 5 and Table I we can see that larger power yield improvements can be achieved when more low-Vdd function units are used in the design. The results also validate our claims in Section IV-C and Section V-B that, asynchronous level conversion is more favorable in statistical optimization, because it enables timing borrowing between function units, and leads to the timing yield improvement that can compensate the overhead of the converters. Therefore, compared to the synchronous case, more asynchronous converters are used while yielding better results.

Fig. 6 shows power yield improvement with multi-V_{th} technique only, which means only the resource units with nominal supply voltage V_{dd}^H can be selected. In this case, no level conversion is needed so there is no overhead for level converters. The average power yield improvements against worst-case delay based approach, under timing yield constraints 99%, 95% and 90% are 6.6%, 8.5% and 11.0%, respectively. At timing yield 95%, the average power yield improvement (8.5%) is smaller than the LC-Avoidance case (10.3%) in Fig. 5, which shows that using multi-V_{dd} resource units can further improve the power yield.

Fig. 7 shows the power yield improvement against worst-case delay based approach, under different timing yield constraints. Asynchronous level conversion is chosen in this series of experiments. The average power yield improvements under timing yield constraints 99%, 95% and 90% are 11.7%, 21.0% and 27.2%, respectively. It is clearly shown that, the power yield improvement largely depends on how much timing yield loss is affordable for the design. This will further push forward the design space exploration for a well balanced timing and power trade-off.

VII. CONCLUSIONS

In this paper, we investigate the impact of process variations on multi-V_{th}/V_{dd} techniques for low power high-level synthesis. We characterize delay and power variations of function units under different threshold and supply voltages, and feed the variation-characterized resource library to the HLS design loop. Statistical timing and power analysis for high-level synthesis is then introduced, to help our proposed resource binding algorithm explore the design space, and maximize the power yield of designs under given timing yield constraints. Experimental results show that significant power reduction can be achieved with the proposed variation-aware framework, compared with traditional worst-case based deterministic approaches.

REFERENCES

[1] P. Coussy and A. Morawiec (Editors). *High-Level Synthesis: From Algorithm to Digital Circuit*. Springer, 2009.
[2] W. Shiue. High level synthesis for peak power minimization using ILP. In *ASAP*, 2000.
[3] K. Khouri and N. Jha. Leakage power analysis and reduction during behavioral synthesis. In *IEEE Transaction on VLSI*, 2002.
[4] X. Tang, H. Zhou, and P. Banerjee. Leakage power optimization with dual-Vth libraray in high-level synthesis. In *DAC*, 2005.
[5] W.L. Hung., X. Wu, and Y. Xie. Guaranteeing performance yield in high-level synthesis. In *ICCAD*, 2006.
[6] J. Jung and T. Kim. Timing variation-aware high-level synthesis. In *ICCAD*, 2007.
[7] F. Wang, G. Sun, and Y. Xie. A variation aware high level synthesis framework. In *DATE*, 2008.
[8] G. Lucas, S. Cromar, and D. Chen. FastYield: Variation-aware, layout-driven simultaneous binding and module selection for performance yield optimization. In *ASPDAC*, 2009.
[9] A. Srivastava, T. Kachru, and D. Sylvester. Low-power-design space exploration considering process variation using robust optimization. In *IEEE TCAD*, 2007.
[10] K. Usami and M. Igarashi. Low-power design methodology and applications utilizing dual supply voltages. In *ASPDAC*, 2000.
[11] A. Srivastava and D. Sylvester. Minimizing total power by simultaneous Vdd-Vth assignment. In *IEEE TCAD*, 2004.
[12] S. Paik I. Shin and Y. Shin. Register allocation for high-level synthesis using dual supply voltages. In *DAC*, 2009.
[13] S.P. Mohanty and E. Kougianos. Simultaneous power fluctuation and average power minimization during nano-CMOS behavioral synthesis. In *VLSID*, 2007.
[14] F. Wang and *et al.* Variability-driven module selection with joint design time optimization and post-silicon tuning. In *ASPDAC*, 2008.
[15] Synopsys. Primetime VX. http://www.synopsys.com/.
[16] NCSU. 45nm FreePDK. http://www.eda.ncsu.edu/wiki/FreePDK.
[17] N. Kim and et al. Leakage current: Moore's Law meets static power. In *IEEE Computer*, 2003.
[18] K. Cao and et al. BSIM4 gate leakage model including source-drain partition. In *IEDM*, 2000.
[19] S. Kulkarni and *et al.* A new algorithm for improved Vdd assignment in low power dual Vdd systems. In *ISLPED*, 2004.
[20] S. Tawfik and V. Kursun. Multi-Vth level conversion circuits for multi-VDD systems. In *ISCAS*, 2007.
[21] E. Aarts and J. K. Lenstra. *Local Search in Combinatorial Optimization*. Princeton University Press, 2003.
[22] A. Raghunathan and N.K. Jha. An iterative improvement algorithm for low power data pathsynthesis. In *ICCAD*, 1995.
[23] F. Ishihara, F. Sheikh, and B. Nikolic. Level conversion for dual-supply systems. In *IEEE TVLSI*, 2004.