Low-Overhead Circuit Synthesis for Temperature Adaptation Using Dynamic Voltage Scheduling

Swaroop Ghosh, Swarup Bhunia*, and, Kaushik Roy
School of Electrical and Computer Engineering, Purdue University, IN, USA
*Electrical Engineering and Computer Science, Case Western Reserve University, OH, USA

Abstract—Increasing power density causes die overheating due to limited cooling capacity of the package. Conventional thermal management techniques e.g. logic shutdown, clock gating, frequency scaling, simultaneous voltage-frequency tuning etc. increase the design complexity and/or degrade the performance significantly. In this paper, we propose a novel design technique, which makes a circuit amenable to temperature adaptation using dynamic voltage scheduling (DVS). It is accomplished by a synthesis technique that (a) isolates and predicts the set of paths that may become critical under variations, (b) ensures they are activated rarely, and (c) tolerates possible delay failures (at reduced voltage) in these paths by adaptive clock stretching. This allows us to schedule a lower supply voltage during increased temperature without requiring frequency tuning. Simulation results on an example pipeline show that proposed design yields similar temperature reduction as conventional design with only 11% performance penalty and 14% area overhead. The conventional pipeline design, on contrary, leads to 50% performance degradation due to reduced operating frequency.

1. INTRODUCTION
Aggressive technology scaling has been the driving force of semiconductor industry for decades to achieve nearly exponential growth in computational power and device integration. However, a major design challenge that has emerged with technology scaling is the increasing power density of a chip (due to integration of large number of devices in unit area as well as increased dynamic and leakage power [1]). The increased power density translates to excessive heat generation while the cooling capability of the package remains limited; giving rise to elevation of die temperature as well as localized heating of a chip (called “hot-spots”) [2].

Traditionally, a circuit is designed to operate at worst-case temperature which is inefficient. Dynamic thermal management techniques like logic shutdown, clock gating, frequency scaling [1, 2] etc. have been proposed in the past. Although these techniques are capable of bringing down power density and temperature, they also degrade the performance considerably. Dynamic voltage scaling can be an effective way to reduce temperature in the event of overheating because it reduces both the switching as well as leakage power. However, if the clock frequency is not scaled simultaneously, then increased delay (due to lower voltage) may result into wrong computation. Combined V-f control technique [3] reduces leads to cubic benefit in terms of power saving at the cost of complex control circuitry. In [3], a dedicated micro-controller is used for V-f control.

In this paper, we present a novel design technique which improves the opportunity for aggressive DVS during elevated temperature with small area and performance overhead. We achieve this by employing critical path isolation technique [4]. The notion of critical path isolation indicates the confinement of critical paths in a synthesized design to known logic blocks. The proposed design methodology isolates the critical paths and makes them predictable and rare even under variations. Possible delay errors are predicted ahead of time (by decoding few input bits) and avoided by stretching the clock to two clock cycles (assuming all standard operations are single cycle). This lets us scale the supply voltage aggressively for thermal management without large performance penalty or complex logic for frequency tuning. We propose an automatic synthesis process based on the above design concept. The synthesis process isolates the critical paths and makes them predictable (based on few primary inputs). The isolated critical paths operate with single cycle at nominal supply voltage under nominal temperature. However, at increased temperature in a thermally constraint environment, the supply voltage is reduced and since possible delay errors in critical paths are deterministic, they are avoided by stretching the clock to two-cycles on critical path activation.

The synthesis process restricts occurrences of the above two-cycle operations by reducing critical path activation probability. It also increases the delay margin between critical and non-critical paths by both logic synthesis and proper gate sizing to allow aggressive DVS at elevated temperature while maintaining good performance.

The design technique presented in [4] also relies on the idea of critical path isolation. However, the objective of [4] is to re-design the circuit for operation at fixed low supply voltage with occasional clock stretchings. In this work, the circuit is synthesized in a way that it operates in single-cycle at normal temperature (and nominal supply) while at high temperature, it operates at lower supply voltage and maintaining frequency with occasional clock stretchings. Consequently, reduction in temperature can be achieved with small performance overhead. Possible delay errors due to critical paths are deterministic and can be avoided by stretching the clock to two cycles.

The proposed technique has following major advantages over existing thermal management methods [1, 2, 3]: (a) it allows us to dynamically schedule the supply voltage during overheating without tuning the clock frequency; (b) the pipeline need not be stalled for more than two-cycles (not even during the DVS process); and (c) the control overhead is negligible compared to existing (V, f) control techniques.

II. PRELIMINARY ANALYSIS

In this section, first we present a pipeline design to illustrate the proposed DVS approach for thermal management. Next, we present a design flow that extends the approach to random logic circuits.

A. Voltage scaling and clock stretching in a pipeline design

For the sake of simplicity, we choose two 4-bit ripple carry adders (RCA) as pipeline stages (Fig. 1) to illustrate the concepts. If $P_0$, $P_1$, $P_2$, and $P_3$ are the propagate signals ($P_i = A_i + B_i$) and $C_{i,0}$ is carry-in signal then the critical path is activated only when $P_0$, $P_1$, $P_2$, $P_3$, and $C_{3,0} = 1$. Since the probability of such occurrence is very low, one can scale down the supply voltage when the circuit is overheated (to reduce the temperature) such that: (a) all operations with $P_0$, $P_1$, $P_2$, $P_3$, $C_{3,0}$ can still be performed in one-cycle; (b) during critical path activation, the possible delay error due to scaled supply can be avoided by performing addition in two clock cycles. The activation of critical path can be predicted by pre-computation of $P_0$, $P_1$, $P_2$, $P_3$, $C_{3,0}$. Note that this approach incurs penalty of an extra clock cycle when the set of critical paths are activated at low supply voltage. However, by ensuring small activation probability of critical paths, it may be possible to maintain temperature by rarely paying penalty of an extra clock cycle.

To evaluate the feasibility of this idea, we simulated the pipeline (Fig. 1) with $V_{DD} = 1V$ in Hspice using BPTM 70nm devices [5]. The critical path delay of the RCAs were found to be 260ps. Based on the critical path delays, we selected clock period to be 270ps. The average power dissipation of the entire design was 26.1uW. Assuming clock period to be 270ps, we reduced the supply voltage to 0.80V. The actual critical path delays of RCAs were found to be 530ps with scaled supply. The new power dissipation was 14.7uW (44% saving). The performance penalty due to two-cycle operations (by clock stretching) of pipeline stages at 0.8V is observed to be small (about 5.8% assuming 50% switching activity of $P_0$, $P_1$, and $C_{3,0}$) and may occur only during overheating of the chip.
The overall DVS scheme (Fig. 1) contains two decoders $D_1$ and $D_2$ (essentially, few AND gates) to predict the activation of critical paths of adder-1 and adder-2 stages. A temperature sensor [6] indicates if the current die temperature violates the thermal constraint. Under nominal temperatures, the pipeline operates in one-cycle. However, during elevated die temperature ($\triangleright$ reference temperature $T_{REF}$), the output of the sensor becomes ‘1’ and the supply voltage is reduced from $V_DH$ to $V_{DDL}$. Under this condition, the pipeline is stalled (controlled by signal freeze) whenever the critical path activation is predicted by $D_1$ or $D_2$. Once the temperature drops below $T_{REF}$, the supply is raised back to nominal value and the pipeline operates in nominal condition.

### B. Design flow

In subsection IIA, we presented the idea of temperature-aware DVS for a pipeline, where the critical path in a stage was unique. To exercise similar DVS technique in random logic circuits (usually with skewed path distribution as shown in Fig. 2(a)), we need to ensure that, (a) the critical paths are confined to a logic section whose activation can be predicted; and, (b) the non-critical paths remain non-critical under process, temperature and supply voltage variations by maintaining a safe timing margin. Fig. 2(b) illustrates such a path delay distribution.

To obtain the delay distribution shown in Fig. 2(b), the design needs to be partitioned and synthesized in such a way that the paths are divided into several logic blocks so that; (a) these logic blocks are active or remain idle, based on the state of primary inputs; and, (b) the probabilities of activation of the logic blocks containing critical paths (called critical block) are very low. Therefore, it will be possible to predict the activation of a logic block just by decoding the states of certain inputs. Next, gate sizing is performed on the partitioned logic to maximize the slack between critical and non-critical blocks. Finally, DVS can be performed during overheating such that non-critical blocks meet the desired timing yield with respect to one-cycle delay target whereas critical block meet the yield with respect to stretched clock period. As illustrated in Fig. 2(b), slack $S$ can be utilized to assign low supply voltage at elevated temperature. Under low voltage, the critical block will operate in stretched cycle while the non-critical ones will operate in single-cycle with limited performance degradation (since activation probability of the critical block is very low).

Note that the delay distribution as shown in Fig. 2(b) allows us to assign only a single lower voltage during elevated temperature. However, there can be a situation where the die may require further lower voltages (at the cost of more performance degradation). To encounter such circumstances, one may modify the sizing strategy after partitioning to obtain delay distribution as depicted in Fig. 2(c). This allows assignment of two lower voltages ($V_{DDL1} < V_{DDL2}$). At $V_{DDL2}$ (i.e., nominal voltage), there is no performance penalty. However, at $V_{DDL1}$, the design has a small performance penalty ($p'$) due to occasional clock stretching operations of critical block. At $V_{DDL2}$, the design has performance penalty $p''$ ("sp").

The overall design flow is shown in Fig. 3. Next, we present a mathematical analysis to obtain the conditions under which the proposed temperature-aware design approach can be advantageous compared to conventional design running at half clock frequency during low voltage in the event of thermal emergency.

### C. Analysis of the proposed design methodology

Let us consider two designs for a circuit namely, design-A and design-B with timings as shown in Fig. 2(a) and (c). Design-A is conventional design whereas design-B is the proposed design. Critical paths of both design-A and design-B have a slack of $S_1$ (at voltage $V_{DDH}$) with respect to the clock period $T$. (Fig. 2(a) and (c)). However, the critical paths of design-B are confined to a logic block whose activation can be predicted. A set of non-critical paths (referred as “secondary critical paths”) in design-B maintain a slack of $S_2$ (Fig. 2(c)). Note that the activation of these paths can also be predicted based on the states of few inputs. The other paths of design-B have a slack of $S$. Two extra decoders are needed in design-B for predetermining the occurrences of critical and secondary critical path activation. Let us now compare the power dissipation (and subsequently temperature) of design-A and design-B. In this analysis, we assume that design-A operates with half clock frequency at reduced voltage. Since voltage is proportional to (delay)$^2$, the scaled voltage ($V^2_{DDL1}$) for design-A is given by,

\[ V_{DDL1} \propto \frac{1}{T} - S_1 \quad \text{and} \quad V_{DDL2}^A \propto \frac{1}{2T} \Rightarrow V_{DDL2}^A = V_{DDL1} \left(1 - \frac{S_1}{T}\right) \quad (1) \]

Similarly, $V_{DDL1}^B = V_{DDL1} \left(1 - \frac{S_1}{T}\right)$ and, $V_{DDL2}^B = V_{DDL1} \left(1 - \frac{S_2}{T}\right) \quad (2)$

If the performance penalty due to clock stretching in design-B at $V_{DDL1}$ and $V_{DDL2}$ is $p'$ and $p''$ respectively, then change in steady state temperature from ambient value ($\Delta$Temp$=P R R_{hip}$ where $P$ is power and $R_{hip}$ (in $\Omega$) is thermal resistance as mentioned in equation (14) of both designs are given by

\[ \Delta$Temp$^A = C^A (V_{DDL1}^A)^2 \left(\frac{1}{2T} \right) \propto \frac{(V_{DDL1}^A)^2}{T} \quad (3) \]

\[ \Delta$Temp$^B = (C^A + C^B) (V_{DDL1}^B)^2 \left(\frac{2}{T + p'T} \right) \propto \frac{(V_{DDL1}^B)^2}{T(1 + p')} \]

\[ \Delta$Temp$^B = (C^A + C^B) (V_{DDL2}^B)^2 \left(\frac{1}{T + p'T} \right) \propto \frac{(V_{DDL2}^B)^2}{T(1 + p')} \quad (4) \]

where $C^A$ ($C^B$) is average switched capacitance, $R^A$ ($R^B$) is thermal resistance of design-A (design-B) and $C^A$ is the average switched capacitance.
capacitance of decoding logic (to determine critical path activation).

The $\Delta Temp$ ratio is given by

$$\Delta Temp^4 = \left( \frac{V_d}{V_{th}} \right) (1 + p') T_c = \left( \frac{V_d}{V_{th}} \right) \left( 1 + p' \right)$$

Putting the values of $V_d/V_{th}$ and $V_{th}/V_{dd}$, and rearranging we get,

$$\Delta Temp^4 = \left( \frac{1 - S_{norm}}{1 - S_{norm}} \right) \left( 1 + p' \right) \left( 1 - \frac{S_{norm}}{1 - S_{norm}} \right)$$  \hspace{1cm} (4)

Similarly,

$$\Delta Temp^2 = \left( 1 - S_{norm} \right) \left( 1 - S_{norm} \right) \left( 1 + p' \right) \left( 1 - \frac{S_{norm}}{1 - S_{norm}} \right)$$  \hspace{1cm} (5)

where $S_{norm} = (S_{norm1}, S_{norm2})$ is the normalized slack of design-A (design-B).

From the above discussion, it is obvious that design-B can be effective for reducing temperature than design-A if $\Delta Temp^4/\Delta Temp^2 \geq 1$ and $\Delta Temp^2/\Delta Temp^4 \geq 1$. Since $(1 + p') < 2$ and $(1 + p') < 2$, a necessary condition for design-B to be better or equal to design-A is $S_{norm} \gg S_{norm}$, i.e., $S_2 > S_1 > S_1$. Therefore, a larger value of $S_2$ and $S_1$ is better for lowering the temperature. However, the upper bound of $S_2$ and $S_1$ is determined by $S_2, S_1 < T_c$ and the area overhead to achieve it. From first order analysis, conventional design may appear to be better than proposed design in terms of temperature reduction. However, the issues involved in design-A are as follows: (a) it degrades the performance more than the proposed design ($2T_c$ vs. $T_c(1+p')$ or $T_c(1+p')$); (b) it may not be possible to scale the supply beyond $3V_{dd}$ limit due to increased delay spread at low voltages, and (c) it may be possible to sufficiently reduce temperature in a thermally constrained environment by small supply voltage reduction.

### III. DESIGN METHODOLOGY

Based on the analysis presented above, here we describe in details each step of the design flow (Fig. 3) and present simulation results.

#### A. Circuit partitioning and synthesis for critical path isolation

To perform an input-based partitioning of the circuit for isolating critical paths and reducing their activation probabilities, we have used Shannon expansion based partitioning [4, 7, 8] scheme. This method is similar to [4] and will be described here briefly for the sake of clarity.

Shannon expansion partitions a Boolean expression ($f$) as follows:

$$f(x_1, x_2, \ldots, x_n) = f(x_1, x_2, \ldots, x_n = 1, \ldots, x_n) + \sum_{j=1}^{n} x_j f(x_1, x_2, \ldots, x_n = 0, \ldots, x_n)$$  \hspace{1cm} (6)

$$CF_1 = f(x_1, x_2, \ldots, x_n = 1, \ldots, x_j) \quad CF_2 = f(x_1, x_2, \ldots, x_n = 0, \ldots, x_j)$$

where $x_j$ is called the control variable, and $CF_1$ and $CF_2$ are called cofactors. The outputs of cofactors are merged using amux controlled by $x_j$. If $f$ contains sub-expressions independent of control variable $x_j$, then we may also have a Shared Cofactor. Note that, in equation (6), the activation probability of each cofactor is 50%. However, by performing multi-level expansion, it is possible to reduce the activation probability of the resulting cofactors further. For example, a 2nd level expansion of $f$ (equation (7)) reduces activation probability of resulting cofactors to 25%.

$$f(x_1, x_2, \ldots, x_n) = \sum_{j=1}^{n} x_j CF_1 + \sum_{j=1}^{n} x_j CF_2 + \sum_{j=1}^{n} x_j CF_3$$  \hspace{1cm} (7)

Control variable selection plays a very important role in achieving the desired goals in Shannon’s based partitioning. If $a_i, b_i$ is the literal count of variable $x_i$ in true (complement) form in the critical function (or output in $f$), then following metric can be used for critical path isolation of the circuit after expansion [4]:

$$M_i = \frac{1}{\max(a_i, b_i)}$$  \hspace{1cm} (8)

To achieve the dual objectives of isolating the critical paths to a cofactor and reducing its activation probability during partitioning and synthesis, first, the circuit is partitioned and the cofactors containing critical paths are determined. Then, the cofactors marked for further expansion (to reduce the activation probability of critical paths). The process is repeated under a given area and delay constraint.

#### B. Gate sizing for further isolation and creation of slack

Once the critical paths are isolated to a cofactor with small activation probability, the next step is to size the resulting cofactors individually to (a) further isolate the critical paths and, (b) create sufficient timing slack between critical and non-critical cofactors to achieve the desired delay distribution (Fig. 2(c)). To achieve it, the critical cofactor is downsized to meet a slack of $S_1$ with respect to the clock period $T_c$. Further, secondary critical cofactor is sized to meet a slack of $S_2$. All other cofactors are upsized to meet the slack $S_3$ (where $S_1 < S_2 < S_3$). An example of the proposed sizing approach after partitioning is shown in Fig. 4. The original circuit is partitioned into four cofactors, $CF_{01}, CF_{02}, CF_{03}$, and $CF_{04}$. Cofactors $CF_{03}$ and $CF_{04}$ are at the highest level of hierarchy (or # of control variables). $CF_{03}$ is downsized selectively to make it further critical while $CF_{04}$ is upsized selectively to make it secondary-critical. The other cofactors are upsized aggressively under area constraint to make them more non-critical. Note that, in this work hierarchy and # of control variables are used interchangeably.

A Lagrangian Relaxation (LR) based gate sizing [9] is used in the proposed sizing procedure (Table 1). Given a delay target ($T_c$), it tries to meet the yield requirement with minimum area. The procedure takes $\lambda List$ (i.e., list of cofactors) and determines the highest level of hierarchy, maxLevel (Step 1). To compute the path delays, statistical timing analysis (SSTA) [10] is performed (Step 2) on each of the cofactors from $\lambda List$. In Step 3, a cofactor at maxLevel with critical paths is selected as critical block candidate. Next a cofactor with hierarchy equal or one less than maxLevel is selected as secondary critical cofactor (scrt(CF)) (Step 4). Now, the critical (secondary critical) logic block candidates are downsized (upsized) selectively based on their slack requirements (Step 5 to 8). Note that the mux delays are subtracted from the delay target to derive the cofactor delay targets. The non-critical cofactors are selectively upsized while
meeting the yield target (Steps 9 to 13). Finally, a complete graph G is created and returned (Step 14 and 15).

C. Determination of supply voltages

After circuit partitioning and sizing, we obtain a netlist with defined critical, secondary critical and non-critical logic blocks. The last objective is to determine supply voltages VDDL1 and VDDL2 for DVS. VDDL1 is determined by scaling down the supply voltage and performing SSTA till the point where delay of secondary critical cofactor meets the yield constraint for one-cycle delay target (Tc). Similarly, VDDL2 is determined by scaling the supply voltage until the delay of non-critical cofactors meets the yield constraint for target Tc.

D. Simulation results and discussion

The experiments are performed on a set of MCNC benchmark circuits. We used Synopsys Design Compiler [11] for logic optimization in our synthesis flow. For a basis of comparison, the original benchmarks are also optimized for area in Synopsys. The mapping is done to a standard cell library. The circuit delays are computed by using SSTA for BPM 70nm technology with 1V nominal supply voltage. The delay target (Tc) for sizing procedure is chosen by plotting the area-delay curve of the circuit and selecting the knee point delay. Slack targets S1, S2 and S3 for sizing are chosen to be 0.01T, 0.3T, and 0.4T, respectively. The area and delay constraints for Shannon partitioning are kept 40% and 5% more than actual area and delay of the circuit respectively, while the yield targets of original circuit and the cofactors for gate sizing are set to 95%. The yield target of cofactors operating on one-cycle or stretched cycle at lower supply voltages during DVS is also fixed to 95%. Reduced voltage of original circuit is determined by iteratively reducing the supply voltage to meet 95% timing yield corresponding to half frequency. Supply voltages VDDL1 and VDDL2 for proposed circuit are determined as described in previous subsection.

The path delay distribution of two benchmarks (cht and cmb) after partitioning and sizing is shown (Fig. 5). Org and New represent path delay distribution of the original and the proposed design, respectively. The path distribution of critical cofactor is also plotted indicating that critical paths are isolated to critical cofactor. Note that the critical cofactor is activated by 4 control variables whereas secondary critical cofactor is activated by 3 control variables. The delays of secondary critical cofactor and non-critical cofactors normalized with respect to critical cofactor delay of the benchmarks are shown in Fig 6(a). The corresponding low supply voltages for conventional design (VDDL1) and proposed design (VDDL1, VDDL2) are also shown (Fig 6(b)) for 1V nominal supply voltage.

In Fig. 6(c), we compare the steady-state temperature of original circuit running at nominal voltage (with rated frequency) and reduced voltage (VDDL with half frequency) with proposed circuit running at VDDL1 and VDDL2. The temperature is given by T = Tambient + PR, where P is power, R is thermal resistance (given by ρ/α), and Tambient is ambient temperature. In our simulation, we assumed ρ = 10^-7 mK/W, = 0.1mm [12] and Tambient = 100°C. Power is computed in Hspice by application of 200 random patterns. Fig. 6(c) indicates that proposed design (i.e., New) running at VDDL1 and VDDL2 can reduce the temperature similar to Org running at VDDL except for benchmarks cmb and alu2 (due to small available slack for DVS, Fig. 6(a)). Note that, the temperature changes are small in this experimentation due to small benchmark sizes and corresponding values of power.

The average temperature reduction for Org(VDDL), New(VDDL1) and New(VDDL2) is similar (i.e., 0.78°C, 0.70°C and 0.80°C, respectively). However, Org(VDDL) incurs 50% performance degradation whereas New(VDDL1) and New(VDDL2) incurs only 5.8% and 15.2% degradation respectively, assuming 50% switching activity of control variables. The penalty reduces to 0.16% and 1.47% if switching activity is 20%. The average area overhead of the proposed methodology is about 14% (Fig. 6(d)).

IV. TEMPERATURE-ADAPTIVE PIPELINE DESIGN

In Section III, we presented a design methodology for random logic circuits to achieve temperature adaptation. In this section, first we present its application to pipeline-based design where each stage is designed using this methodology and discuss the performance overhead due to pipeline stalls for stretching the clock to two-cycles during die overheating. Next, we propose an architecture suitable for performing temperature-adaptive DVS. Finally, we present simulation results for a 3-stage pipeline design.

A. Performance analysis

Consider an N-stage linear pipeline where each stage is designed using the proposed technique. The pipeline is stalled for an extra cycle whenever a clock stretching is performed by one or more stages in the current cycle. Probability of pipeline stall (p_stall) is given by

\[ p_{\text{out}} = 1 - (1 - p_k)(1 - p_2)...(1 - p_k) \]

where \( p_i \) is the activation probability of critical block of \( i^{th} \) stage. If critical block of each stage has same number of control variables \( p_1 = p_2 = ... = p_k = \text{input switching activity}^k \)

where \( k \) is the hierarchy or # control variables of critical block. Hence,

\[ p_{\text{out}} = 1 - (1 - p)^k \]

If the circuit is designed to operate at the two lower voltages (Fig. 2(c)), then the probability of clock stretching becomes

\[ p_{\text{out}} = 1 - [(1 - p)(1 - p)]^k \]
the performance penalty due to designed, synthesized circuits as outputs. The stage delays are pipeline stages and the target voltages as inputs and produce re-
critical B. Design methodology
penalty at V_{critical}
blocks further (i.e., by increasing control variables; and, (b) reduce the activation probability of the control variable selection metric to pick low switching inputs as following techniques for reducing the performance penalty, (a) tune can be large for deep pipeline designs (i.e. large restricted within 10% (25%) for variables have low switching activity (~20%), then the penalty can be shown in Fig 7(b). It can be observed from this plot that if the control
variables have low switching activity (~20%), then the penalty can be
restricted within 10% (25%) for k=4 at V_{DDL1} (V_{DDL2}). The penalty can be large for deep pipeline designs (i.e. large N). We suggest the following techniques for reducing the performance penalty, (a) tune the control variable selection metric to pick low switching inputs as control variables; and, (b) reduce the activation probability of critical blocks further (i.e., by increasing k).
B. Design methodology

B. Design methodology is based on a given set of reduced supply voltage constraints (i.e., V_{DDL1} and V_{DDL2}). The procedure is shown in Table 2. It takes a target yield, the list of pipeline stages and the target voltages as inputs and produce re-designed, synthesized circuits as outputs. The stage delays are computed by sizing the design as explained in Section IIIIB. The maximum stage delay is chosen as the target delay (T_{c}) for all the stages (step-1). Next, one design is picked at a time and circuit partitioning is performed as explained in Section III (step-3). Note that the slacks S_{r}, S_{2} and S_{3} are computed in Steps 4-6 by using SSTA at specified supply voltages (V_{DDH}, V_{DDL1} and V_{DDL2}). The output of step-3 is a list of cofactors which is sized to meet the required slacks (Step-7). Steps 2-5 are performed on each of the pipeline stages and the list of pipeline stages is returned as output.

The architecture of a temperature-adaptive pipeline (with two lower voltages) is shown in Fig. 8. Decoders D_{11}, D_{21} and D_{31} determine critical cofactor activation of one of the stages. Similarly, decoders D_{12}, D_{22} and D_{32} determine the activation of secondary critical cofactor of the stages. These two set of decoders are OR-ed to predict the activation of critical and secondary critical cofactors of pipeline stages. The predictions are gated with thermal sensor outputs. A freeze signal is generated only when the temperature of the chip crosses the threshold value and, a clock stretching is predicted by the decoders. When the temperature goes above T_{REF1}, the supply voltage is automatically reduced by the multiplexer to V_{DDL1}. However, if T_{REF1} is violated, then supply voltage is reduced to V_{DDL2} and both critical and secondary critical cofactors are operated in stretched clock. Since the performance of pipeline at V_{DDL2} is lower than that of V_{DDL1}, a trade-off can be made between temperature and performance.

Note that traditional supply reduction techniques may require pipeline stalling for few cycles to tune the clock frequency. However, the proposed pipeline need not be stalled during DVS process. This is true because activation of critical paths is predicted ahead in time and corrective action is taken. To avoid delay failures when the supply is brought back from low to nominal voltage, we place a small counter to

---

TABLE-2

<table>
<thead>
<tr>
<th>Procedure pipelineDesign()</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong>: (Y, list of circuits(dList), V_{DDL1}, V_{DDL2}; <strong>Output</strong>: list of re-designed circuits (dList);</td>
</tr>
<tr>
<td>1 target delay (T_{c}) = max(stage delays);</td>
</tr>
<tr>
<td>2 for each design D ∈ dList</td>
</tr>
<tr>
<td>3 gList = performPartitioning(D, V_{DDH});</td>
</tr>
<tr>
<td>4 S_{r} = Slack of critical in gList at V_{DDH} to meet yield Y;</td>
</tr>
<tr>
<td>5 S_{2} = Slack of critical in gList at V_{DDH} to meet yield Y;</td>
</tr>
<tr>
<td>6 S_{3} = Slack of other cofactors in gList at V_{DDH} to meet yield Y;</td>
</tr>
<tr>
<td>7 D_{i} = performSizing(gList, T_{c}, Y, S_{r}, S_{2}, S_{3});</td>
</tr>
<tr>
<td>8 end for</td>
</tr>
<tr>
<td>9 return dList;</td>
</tr>
</tbody>
</table>

where p^{*} is the activation probability of secondary critical block of every stage.

If the ideal clock cycle-per instruction (CPI) of the pipeline is given by CPI_{ideal}, then new CPI is CPI_{new} = CPI_{ideal} + P_{stall}(stall penalty) and the performance penalty due to clock cycle stretching is given by

\[ \text{Performance penalty} = \frac{\text{CPI}_{\text{new}} - \text{CPI}_{\text{ideal}}}{\text{CPI}_{\text{ideal}}} = \frac{P_{stall}(\text{stall penalty})}{1 + P_{stall}} \]  \quad (13) \]

The performance penalty for different N and k (for input switching activity of 20%) is shown in Fig. 7(a). The solid (dashed) lines denote penalty at V_{DDL1} (V_{DDL2}). In these plots, we assume that the secondary critical cofactor of each stage has one less control variable than critical cofactor. The penalties for switching activity of 50% are shown in Fig 7(b). It can be observed from this plot that if the control variables have low switching activity (~20%), then the penalty can be restricted within 10% (25%) for k=4 at V_{DDL1} (V_{DDL2}). The penalty can be large for deep pipeline designs (i.e. large N). We suggest the following techniques for reducing the performance penalty, (a) tune the control variable selection metric to pick low switching inputs as control variables; and, (b) reduce the activation probability of critical blocks further (i.e., by increasing k).

Fig. 6: (a) Normalized delays of critical, secondary critical and non-critical cofactors; (b) temperature of conventional and proposed design at nominal and reduced voltages; and, (c) area of conventional and proposed design

Fig. 7: Performance penalty for different pipeline depths at uniform input switching activity of (a) 20%; (b) 50%
extend the freeze signal (Fig. 8). Although the proposed adaptive DVS technique has been demonstrated for 3-stage pipeline, it can be extended for N-stages.

C. Thermal model for electro-thermal simulation

To demonstrate the impact of DVS on the transient behavior of temperature, we follow lumped RC thermal model [12]. The thermal resistance ($R_{\text{chip}}$) and capacitance ($C_{\text{chip}}$) of the circuit is given by

$$R_{\text{chip}} = \frac{\rho}{A}; \quad C_{\text{chip}} = \epsilon \cdot A \cdot l$$  \hspace{1cm} (14)

where $\rho$ ($=10^3$ mK/W) is thermal resistivity, $\epsilon$ ($=10^0$ J/m$^3$K) is thermal capacitance, $l$ ($=0.1$ mm) is the thickness of the wafer and $A$ is the die area. The transient temperature at $t^0$ instant is determined as follows

$$T_i = T_{\text{amb}} + [T_{\text{MAX}} - T_{\text{amb}}] \cdot (1 - \exp(-t/\tau_{\text{chip}})); \quad \text{if} \quad (T_{\text{amb}} + R_{\text{chip}} P) > T_{\text{amb}}$$

$$= T_{\text{amb}} \cdot \exp(-t/\tau_{\text{chip}}); \quad \text{otherwise} \hspace{1cm} (15)$$

where $t$ is the time step. $\tau_{\text{chip}} = R_{\text{chip}} C_{\text{chip}}$ and $T_{\text{MAX}}$ is maximum allowable temperature. Equation (15) is based on simple RC circuit charge/discharge expression.

D. Experimental results and discussion

The timing yield targets of original circuit and the cofactors for gate sizing are set to 95%. After partitioning and sizing, the supply voltages $V_{\text{DDL1}}$ and $V_{\text{DDL2}}$ are determined as explained in Section IIIID. Once we get the new pipeline design with nominal and reduced supply voltages, we perform electro-thermal simulation to solve the self-consistent loop of power and temperature at each time step. $R_{\text{chip}}$ and $C_{\text{chip}}$ of the circuit are computed by using equation (14). For power estimation, the circuit is simulated in Hspice at ambient temperature ($100^\circ$C). After determination of power, the transient temperature is computed using equation (15). The computed temperature is used for determination of power in the next time step. The $T_{\text{REF1}}$ and $T_{\text{REF2}}$ are kept at $104^\circ$C and $110^\circ$C respectively.

The path delay distribution of two of the pipeline stages (i.e. $c_{\text{ht}}$ and $c_{\text{mb}}$ of Fig. 8) is shown in Fig. 5. The target delay was 100ps and ($V_{\text{DDL1}}, V_{\text{DDL2}}$) were found to be (0.8V, 0.75V) with $V_{\text{DDL}}$=1V. The thermal simulation result for our example pipeline (Fig. 8) is shown in Fig. 9. It can be observed that the temperature rises from the ambient value and saturates after some time. At $t=4x10^5$ns, the temperature of the circuit goes above $T_{\text{REF1}}$ and the supply voltage is reduced to $V_{\text{DDL}}$ (i.e. 0.8V) based on sensor’s output. As expected, the temperature reduces to ~100$^\circ$C. At this lower supply, only the critical cofactors of the pipeline stages operate in stretched clock. This simulation demonstrates the self adaptive nature of the system with respect to temperature. For the sake of comparison, we also plotted the

Fig. 9: Transient temperature profile for the 3-stage pipeline temperature profile of the pipeline with conventionally synthesized circuits. It can be noted from the figure that final temperature of original pipeline is lower than re-designed pipeline by approximately 0.05$^\circ$C. This is true because area of the re-designed circuit is larger than the original circuit (by approximately 22%). This raises $R_{\text{chip}}$ and thereby the steady-state temperature by a small fraction (as $T_{\text{final}}=T_{\text{amb}} + R_{\text{chip}} P_{\text{chip}}$). However, the difference lies in performance. In the event of high temperature, the supply of original pipeline is also reduced and operated at half frequency leading to 50% (as $P_{\text{final}} = P_{\text{amb}}$ in equation (13)) performance penalty. On the other hand, the proposed design needs only occasional clock stretching at $V_{\text{DDL}}$. The performance penalty at $V_{\text{DDL1}}$ was found to be 11% (for input signal probabilities of 50%).

V. CONCLUSION

We proposed a novel temperature-adaptive circuit design technique for dynamic temperature control based on critical path isolation. The proposed technique makes delay errors due to supply voltage scaling predictable and rare such that they are avoided by adaptively stretching the clock period. It helps maintain reasonably high performance even under supply voltage scaling at higher chip temperature. The synthesis technique isolates critical paths to few logic blocks by Shannon decomposition and gate sizing. Simulation results on a pipeline-based design show that the proposed design can maintain a target temperature with small area and performance penalty.

Acknowledgement: This research was partly supported by GigaScale System Research Center and Semiconductor Research Corp (#1078.002). We would like to thank Dr. Saibal Mukhopadhyay, Purdue University for valuable discussions.

REFERENCES