



# Energy-Efficient Dual-Voltage Design Using Topological Constraints

Mridula Allani<sup>1</sup> and Vishwani Agrawal<sup>2,\*</sup>

<sup>1</sup>Intel Corporation, Austin, TX, 78746, USA

<sup>2</sup>Electrical and Computer Engineering, Auburn University, Auburn, AL 36849, USA

(Received: 31 March 2013; Accepted: 10 August 2013)

We propose a method for dual supply voltage digital design to reduce energy consumption without violating the given performance requirement. Although the basic idea of placing low voltage gates on non-critical paths is well known, a new two-step procedure does it so more efficiently. First, given a circuit and its nominal single supply voltage, we find a suitable value for a lower second supply voltage that is likely to give the best advantage in power reduction. Besides, using the critical path timing constraint and a linear-time gate slack calculation we also classify gates into three groups. All gates in Group 1 can be simultaneously assigned the lower voltage. Any gate in Group 2 can be assigned the lower voltage but then gate slacks must be recalculated because the group classifications may change. No gate in Group 3 can be assigned the lower voltage. A second step then assigns the lower voltage to the largest possible number of gates using the gate classifications and imposing a topological constraint, preventing any low voltage gate from feeding into a higher voltage gate, thus avoiding the use of level converters. SPICE simulation of dual-voltage ISCAS'85 benchmark circuits using the 90nm bulk CMOS PTM (predictive technology model) shows energy savings of up to 60% with no increase in the original critical path delay and up to 70% with relaxed critical path delay.

**Keywords:** Dual-Voltage Design, Gate Slack, Low Power Design, Energy-Efficient Design, Topological Constraints, Timing Constraints.

## 1. INTRODUCTION

Increasing popularity of portable devices like smart phone, ipad, tablet and notebook has created an overwhelming demand for extended battery life and low power circuits. Power reduction techniques at various levels of abstraction are used in modern digital designs. These techniques include power gating, clock gating, multiple-supply voltages and multiple threshold devices. Our focus in this work is on computationally efficient algorithms for dual-supply voltage digital design.

Decreasing the supply voltage reduces power but results in reduced performance, requiring a trade-off between power consumption and circuit delay. The use of multiple supply voltages to reduce energy consumption is a commonly used technique for CMOS circuits.<sup>7,9,10,13,16,21–25,34,38,40–43,50</sup> The dynamic power of a CMOS circuit is directly proportional to the square of its supply voltage<sup>8,37</sup> and the underlying idea in this technique

is to trade available timing slack off to reduce power. Generally, the gates on critical paths are kept at high supply voltage and those on non-critical paths are put to lower supply voltage, thus avoiding any specified timing violations.

The slack of a gate is defined<sup>21–25</sup> as the difference between the critical path delay of the circuit and the delay of the longest path through that gate. Thus, each gate has its own slack and the gates with the same slack may fall on the same path unless there are multiple paths with equal delay. Also, a positive slack for a gate implies that the timing constraints are met, making any negative slack unacceptable. In this work we use a linear time slack analysis algorithm<sup>25</sup> to calculate gate slacks.

This paper is organized as follows. Sections 2 and 3 outline the motivation contributions of this work. Section 4 provides three theorems that form the basis for the slack-based design in this work. Section 5 gives the algorithms for finding the value of a low voltage and its assignment to gates. Topological constraints are discussed in Section 6. Experimental results are given in Section 7 and we conclude with Section 8. See Section 1.

\* Author to whom correspondence should be addressed.  
Email: vagrawal@eng.auburn.edu

## 2. BACKGROUND AND MOTIVATION

Previous literature<sup>46,48</sup> provides two ways of assigning a lower voltage in a dual-voltage design. The first algorithm<sup>46</sup> is called clustered voltage scaling (CVS). This method puts a topological restriction, referred to as topological constraint in our work, on the dual-voltage design. Accordingly, a low voltage gate cannot feed into a high voltage gate. The second method, extended clustered voltage scaling (ECVS) algorithm,<sup>48</sup> allows a low voltage gate to drive a high voltage gate with the inclusion of an asynchronous level converter. Both CVS and ECVS aim at utilizing the surplus timing slack in non-critical paths by applying a lower supply voltage to gates on those paths. This results in an overall reduction in the dynamic power.

Several other algorithms have been proposed for dual/multiple-voltage assignments modifying the CVS and ECVS ideas. Reference<sup>47</sup> describes a methodology to synthesize circuits for the CVS and ECVS structures and authors claim to improve power savings by up to 28% and 13% over the original CVS and ECVS, respectively.

Another paper<sup>39</sup> describes three algorithms for dual voltage design based on linear programming models. The first, PROUD, is essentially a linear programming model to minimize the power consumption. The second, PRHEUDENT, is a heuristic for reducing the computation time. The third algorithm numerically rounds a non-integral delay to the next higher integer and uses PROUD for power minimization. All three algorithms use level converters.

Another technique that optimizes gate sizing, threshold voltage and supply voltage simultaneously using linear programming is discussed in Ref. [11]. In Ref. [24], the authors use a mixed integer linear programming (MILP) technique to find a lower voltage  $V_L$ , given a higher voltage  $V_H$ , where both voltages are in the sub-threshold range. An ECVS type of method is used with multiple logic level gates interfacing the low and high voltage boundaries.

The complexity of these linear programming voltage assignment algorithms is often polynomial. Thus, we are motivated to propose a quadratic (closer to linear) time algorithm for dual-voltage assignment.

In contrast to many algorithms for assigning a given low voltage to the gates of a circuit, relatively few attempt to find the best value of the lower voltage. An often used low voltage  $V_L$  is 70% of the high voltage  $V_H$ .<sup>6,13,28,31,37</sup> However, some authors<sup>9,28,43</sup> suggest that the optimal value of  $V_L$  for minimizing total power is 50% of  $V_H$ . Authors in<sup>51</sup> assume that all gates initially have the lowest possible  $V_L$ . Their procedure then increases the supply voltage of a group of paths having path delays greater than some given clock period,  $T_c$ , and continues until no path has delay greater than  $T_c$ . An additional 19.55% power savings were reported by this technique over the CVS method.

Published work also reports rules of thumb<sup>13</sup> for optimum voltage ratios in multiple- $V_{DD}$  circuits. Thus, for

dual-supplies,  $V_H > V_L$ ,

$$\frac{V_L}{V_H} = 0.5 + 0.5 \frac{V_{th}}{V_H}$$

where  $V_{th}$  is the threshold voltage. For three supplies,  $V_H > V_{L1} > V_{L2}$ ,

$$\frac{V_{L1}}{V_H} = \frac{V_{L2}}{V_{L1}} = 0.6 + 0.4 \frac{V_{th}}{V_H}$$

and for four supplies,  $V_H > V_{L1} > V_{L2} > V_{L3}$ ,

$$\frac{V_{L1}}{V_H} = \frac{V_{L2}}{V_{L1}} = \frac{V_{L3}}{V_{L2}} = 0.7 + 0.3 \frac{V_{th}}{V_H}$$

The authors claim<sup>13</sup> that these rules of thumb give supply voltages, which reduce power to within 1% of the theoretical minimum for an assumed triangle shaped path delay distribution. These results show that the power savings tend to saturate as the number of supplies is increased and also that the savings decrease as the supply voltage is scaled and when  $V_{th}/V_H$  is higher. Using the equation for two supplies, we get only one value of  $V_L$  for all circuits. However, we observe that the  $V_L$  resulting in the least power dissipation depends not only on  $V_{th}$  and  $V_H$  but also on the circuit topology and performance requirement.

Algorithms for finding an optimum  $V_L$  for a given  $V_H$  in dual voltage operation have also been reported.<sup>21-25</sup> One assigns a low voltage value to a group of gates based on a modified CVS algorithm and then calculates energy over a set of low voltages. The value of  $V_L$  resulting in minimum energy is accepted. This algorithm requires the voltage assignment to gates to be done for each voltage value and is exhaustive in nature. We recognize that the lower voltage for minimum energy operation of a dual-supply design is dependent on the circuit topology and is not the same for all circuits. That motivated us to develop a new linear complexity algorithm to find a circuit specific value for  $V_L$ .

## 3. UNIQUE CONTRIBUTIONS

We provide a method (Algorithm 1) to determine a lower supply voltage for maximizing the energy saving from a subset of gates that can be assigned the lower voltage without exceeding the given critical path delay. The complexity of this algorithm is linear, i.e.,  $O(n)$ , where  $n$  is the total number of gates in the circuit. This is because the voltage selection is based upon gate slacks that can be calculated in linear time. We then propose another method (Algorithm 2), which assigns the selected lower supply voltage to the largest number of gates in an iterative manner without violating the given critical path constraint. Gate slacks are recalculated in each iteration, resulting in a quadratic or  $O(n^2)$  complexity for this algorithm. However, in practice we observed that the computation time is closer to being linear. The quality and efficiency of the two algorithms is derived from proven results (Theorems I, II

and III). An additional feature is that the voltage assignment honors a topological constraint that would not allow a low voltage gate to drive a high voltage gate. This eliminates the use of level converters that would otherwise have their delay and energy penalties. Both algorithms are programmed in Perl language. Energy savings of up to 60% are observed for ISCAS'85 benchmark circuits. When the critical timing is relaxed, savings of up to 70% are observed. Such high savings have not been reported in earlier works. Circuit level spice simulation is used to validate the results.

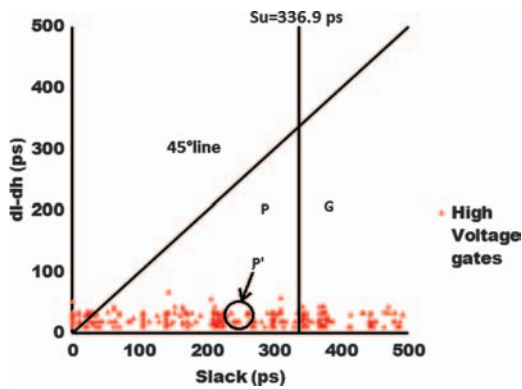
Parts of this work have appeared in a conference publication<sup>3</sup> and detailed experimental data are available in a recent thesis.<sup>4</sup>

#### 4. THEOREMS FOR SLACK-BASED DUAL-VOLTAGE DESIGN

A dual voltage design begins with a specified clock period,  $T_c$ , which requires that the critical path delay of the circuit must not exceed  $T_c$ . Our method is based on three theorems that categorize gates of a given circuit based on their slacks. In this work the slack of a gate is defined<sup>21–25</sup> as the difference between the critical path delay and the delay of the longest path through that gate.

The following discussion is based on the construction shown in Figure 1. For illustration this graph contains the gate slack data for the benchmark circuit c880 using 90 nm bulk CMOS predictive technology model (PTM)<sup>1</sup> with two supply voltages, a higher voltage  $V_H = 1.2$  V and a lower voltage  $V_L = 0.49$  V. A gate can be assigned any of these two voltages. Every gate is represented by a point whose abscissa is the slack and ordinate is the change in its delay if its voltage were to change from  $V_H$  to  $V_L$ . Figure 1 shows the slack data on the  $x$ -axis when all gates are assigned  $V_H$ . Applications of this data will be discussed in later sections.

Suppose the delay of gate  $i$  is  $d_i$ , which equals  $d_{hi}$  when it is assigned  $V_H$  and equals  $d_{li}$  when it is assigned  $V_L$ .



**Fig. 1.** Gate delay increment for  $V_L = 0.49$  V versus gate slack for all gates assigned  $V_H = 1.2$  V in c880 circuit and clock period  $T_c$  equals critical path delay.

The slack of gate  $i$  is computed as,

$$s_i = T_c - \sum_{j \in LP_i} d_j \quad (1)$$

where  $LP_i$  is the set of gates on the longest delay path through gate  $i$  and  $T_c$ , as stated before, is the clock period that must not be less than the delay of any path in the circuit, when operated at supply voltage  $V_H$ . Therefore,  $s_i \geq 0$ , for all  $i$ , is a feasibility condition for a clocked circuit. We refer to this as the *positive slack constraint*.

##### 4.1. Theorem I

*Given two voltages  $V_H$  and  $V_L$ ,  $V_H > V_L$ , no  $V_H$ -gate that falls above the 45° line in the “delay increment versus slack” plot can be assigned  $V_L$  without violating the positive slack constraint. (We classify these as Group 3 gates.)*

**PROOF.** Consider a circuit in which all gates have been assigned a voltage, either  $V_H$  or  $V_L$ . Consider gate  $i$  that has been assigned  $V_H$  and its present slack  $s_{hi}$  is given by (1). If we change its voltage to  $V_L$  keeping all other gates as before, then the new slack  $s_{li}$  of gate  $i$  is obtained as,

$$s_{li} = s_{hi} + d_{hi} - d_{li} \quad (2)$$

The  $V_L$  assignment to gate  $i$  is feasible only if  $s_{li} \geq 0$ . Therefore, the present slack  $s_{hi}$  must satisfy the relation:

$$s_{hi} \geq d_{li} - d_{hi} \quad (3)$$

Because (3) cannot be satisfied by a gate above the 45° line,  $s_{hi} = d_{li} - d_{hi}$ , in Figure 1, the statement of the theorem is proven.

Next, we define a delay ratio  $\beta_i$  for gate  $i$ :<sup>25</sup>

$$\beta_i = \frac{d_{li}}{d_{hi}} \geq 1, \quad \forall \text{ gates } i \quad (4)$$

and an upper bound on the gate delay ratio for the circuit as,

$$\beta_{\max} = \max_{\forall \text{ gates } i} \{\beta_i\} \quad (5)$$

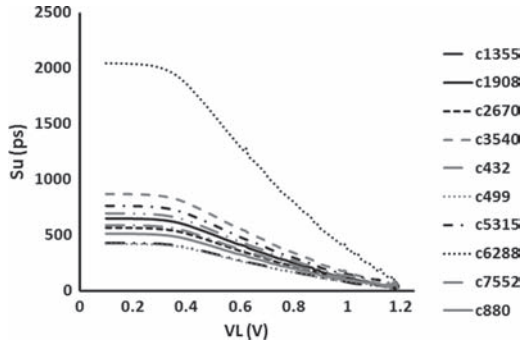
##### 4.2. Theorem II

*Any gate  $i$  whose slack,  $s_i \geq S_u$ , where*

$$S_u = \frac{\beta_{\max} - 1}{\beta_{\max}} \times T_c \quad (6)$$

*can be assigned the lower voltage  $V_L$  without violating the positive slack condition, independent of low voltage assignments made to any other gates with slack greater than  $S_u$ .*

**PROOF.** The slack of gate  $i$ , as computed by (1), is only affected by delays  $\{d_{hj}\}$  of gates on  $LP_i$ , the longest path through  $i$ . When any  $V_H$  gate  $j$  on  $LP_i$  is assigned  $V_L$ , its



**Fig. 2.**  $S_u$  from Eq. (6) versus  $V_L$  for ISCAS'85 benchmark circuits when  $V_H = 1.2$  V and clock period  $T_c$  equals critical path delay.

delay is increased to  $\beta_j d_{hj}$ . As a result the slack  $s_i$  of gate  $i$  is reduced by an amount  $\beta_j d_{hj} - d_{hj}$ . In general, either all gates or a subset of all gates on path  $LP_i$  would have slack greater than  $S_u$ . Considering the extreme case, we assume that all gates on path  $LP_i$  are changed from  $V_H$  to  $V_L$ . Then the new slack of  $i$  is expressed as,

$$s'_i = s_i - \sum_{j \in LP_i} (\beta_j - 1) d_{hj} \geq s_i - (\beta_{\max} - 1) \sum_{j \in LP_i} d_{hj} \geq 0 \quad (7)$$

where the last inequality ensures the feasibility condition of the new slack being positive. From (1), we have

$$\sum_{j \in LP_i} d_{hj} = T_c - s_i \quad (8)$$

Substituting (8) in (7), we get

$$s_i - (\beta_{\max} - 1)(T_c - s_i) \geq 0 \quad (9)$$

$$s_i \geq S_u \quad (10)$$

with  $S_u$  given by (6). Theorem II is thus proved.

Figure 2 shows the slack  $S_u$  as defined by (6) for ISCAS'85 benchmark circuits. The details of technology (90 nm CMOS), synthesis and analysis of these circuits are given in Section 7. The high voltage  $V_H = 1.2$  V and the slack boundary  $S_u$  of (6), as used in Theorem II is shown by a vertical line in Figure 1. The gates in region  $G$  on the right of the the  $S_u$  line are defined as Group 1 gates.

The triangular region  $P$  in Figure 1 that is bounded by the  $slack = S_u$  vertical line, the  $45^\circ$  line, and  $x$ -axis, defines Group 2 gates. According to (3), any single  $V_H$  gate that lies below the  $45^\circ$  line in Figure 1 can be assigned  $V_L$  without causing negative slack. However, Group 2 gates being on the left of the  $S_u$  line do not satisfy the condition of Theorem II. The  $V_L$  assignment to a Group 2 gate can potentially reduce the slacks of other Group 2 gates pushing them above the  $45^\circ$  line. Typically, one may iteratively assign  $V_L$  to a single gate and recalculate slack.<sup>25</sup> The result of the following theorem can speed up the iterative process.

### 4.3. Theorem III

All gates in a subset  $P'$  of  $V_H$  gates within Group 2 can be simultaneously assigned  $V_L$  without producing any negative slack provided  $P'$  satisfies the following condition:

$$\sum_{j \in P'} y_j \leq \min_{i \in P'} \{x_i\} \quad (11)$$

where  $(x_i, y_i)$  are the coordinates of the point representing gate  $i$  on the "delay increment versus slack" graph (see Figure 1), that is,  $x_i = s_{hi}$  is the slack of gate  $i$  when assigned  $V_H$  and  $y_i = d_{li} - d_{hi}$  is the delay increase for gate  $i$  when its voltage is changed from  $V_H$  to  $V_L$ .

**PROOF.** Consider a  $V_H$  gate  $i$  in  $P'$ , represented by point  $(x_i, y_i)$  on the "delay increment versus slack" graph. Suppose,  $LP_i$  is the longest delay path through  $i$ . If the voltage of  $i$  is changed to  $V_L$  then its slack will reduce to  $x_i - y_i$ . Other gates on  $LP_i$  that had the same slack  $x_i$  will also have a similar slack reduction. There can be other paths through gate  $i$  but their delays are smaller than that of  $LP_i$  and the slacks of gates on them, though reduced by amount  $y_i$  are greater than  $x_i - y_i$ . Thus, as long as  $x_i - y_i \geq 0$ , we are sure that no gate will violate the positive slack condition due to the  $V_L$  assignment of gate  $i$ .

Next, we reduce the voltages of all  $V_H$  gates  $i$  in  $P'$  to  $V_L$ . Let us consider two extreme cases:

*Case 1:* Gates in  $P'$  are on disjoint paths and slack reductions of gates do not influence each other. Therefore, the non-negative slack condition is,

$$x_i - y_i \geq 0, \quad \forall i \in P'$$

Or

$$y_i \leq x_i, \quad \forall i \in P' \quad (12)$$

*Case 2:* Slacks of all paths in  $P'$  are determined by the same path  $LP_i$ , i.e.,  $x_i = x_j = x_k = \dots$ , although  $y_i \neq y_j \neq y_k \neq \dots$ , for  $\{i, j, k, \dots\} \in P'$ . Now the delay of path  $LP_i$  is increased by an amount  $y_i + y_j + y_k + \dots$ , and the slack of each gate in  $P'$  is reduced by the same amount. Hence, the non-negative slack condition is,

$$x_i - \sum_{j \in P'} y_j \geq 0, \quad \forall i \in P'$$

Or,

$$\sum_{j \in P'} y_j \leq x_i, \quad \forall i \in P'$$

That is,

$$\sum_{j \in P'} y_j \leq \min_{i \in P'} \{x_i\} \quad (13)$$

Note that the weaker condition 12 is subsumed by 13, which proves the theorem for the two boundary cases. All other cases will lie in between these two cases.

## 5. SLACK-BASED ALGORITHMS FOR DUAL-VOLTAGE DESIGN

### 5.1. Estimated Energy Saving

The estimated energy saving for a circuit is computed as:

$$E_{\text{save\_est}} = \frac{V_H^2 - V_L^2}{V_H^2} \times \frac{N}{n} \times 100 \text{ percent} \quad (14)$$

where  $n$  is the total number of gates in the circuit,  $N$  is the number of gates in low voltage  $V_L$ , and  $V_H$  is the higher supply voltage. The energy consumed by the circuit at  $V_H$  is proportional to  $nV_H^2$  and the energy consumed by the circuit in dual voltage design will be proportional to  $(n-N)V_H^2 + NV_L^2$ . Hence, the percentage energy saving is,

$$\frac{nV_H^2 - [(n-N)V_H^2 + NV_L^2]}{nV_H^2} \times 100 = \frac{V_H^2 - V_L^2}{V_H^2} \times \frac{N}{n} \times 100$$

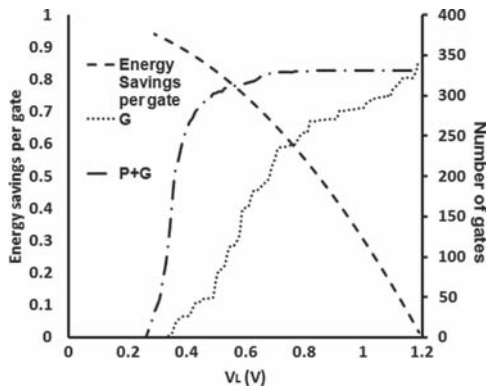
We define energy saving ratio as

$$\frac{V_H^2 - V_L^2}{V_H^2} \times \frac{N}{n}$$

Figure 3 shows how the energy saving per gate, which is  $(V_H^2 - V_L^2)/V_H^2$ , varies with  $V_L$ . The figure also shows the variation of the number of gates  $P + G$  below the 45° line from the “delay increment versus slack” graph, and the number of Group 1 gates  $G$  whose slacks are greater than  $S_u$ . We see that as  $V_L$  gets closer to  $V_H$ , the energy saving per gate decreases even though the numbers of gates in Groups 1 and 2 continue to increase. Hence, we need to find a trade-off to obtain an optimum value of  $V_L$  for maximum energy saving.

### 5.2. Algorithm 1

Given a gate level netlist with specified gate delays and a supply voltage  $V_H$ , the following algorithm finds a second lower supply voltage  $V_L$  for dual voltage low power operation.



**Fig. 3.** The ratio of energy saving per gate  $(V_H^2 - V_L^2) \div V_H^2$ , number of Group 1 gates  $G$ , and number of Group 1 and 2 gates  $P + G$ , as functions of  $V_L$  for circuit c880 when  $V_H = 1.2$  V and clock period  $T_c$  equals critical path delay with all gates assigned  $V_H$ .

For the voltage interval between the threshold voltage  $V_{th}$  and  $V_H$  we select values  $V_{Li}$  at closely spaced intervals. We estimate all gate delays at  $V_{Li} \in \{V_{th}, V_H\}$ . Then carry out the following steps:

*Step 1:* Use an  $O(n)$  slack calculation algorithm proposed in Refs. [21–25] to find out the gate slacks for the given circuit at voltage  $V_H$ . The slack computation also finds the critical path delay for the circuit. The clock period  $T_c$  used for slack calculation, can be set to any value that equals or exceeds the critical path delay, depending on the performance requirement of the circuit. Carry out Steps 2 and 3 for all  $V_{Li} \in \{V_{th}, V_H\}$ :

*Step 2:* Classify gates into Groups 1, 2 and 3 as described in Section 4 (also see Figure 1).

*Step 3:* Estimate the dynamic energy savings for the gates in Groups 1 and 2 together. The dynamic energy if all gates in Groups 1 and 2 were assigned high supply voltage is proportional to,

$$E_H = V_H^2 \times (G + P)$$

Similarly, the dynamic energy when all gates in Groups 1 and 2 were to be assigned low supply voltage is proportional to,

$$E_{Li} = V_{Li}^2 \times (G + P)$$

The maximum possible energy saving from Groups 1 and 2 from dual voltages,  $V_{Li}$  and  $V_H$ , is estimated as,

$$\begin{aligned} E_{\text{save\_est\_i}} &= \frac{E_H - E_{Li}}{E_H} \times 100 \\ &= \frac{V_H^2 - V_{Li}^2}{V_H^2} \times (G + P) \times 100 \end{aligned} \quad (15)$$

*Step 4:* Select the voltage  $V_L$  as that  $V_{Li}$  for which  $E_{\text{save\_est\_i}}$  is maximum. This is given as optimal value for  $V_L$ .

Note that in Algorithm 1 we assumed that all Group 2 gates could have low voltage. While that is true for many gates it is not so for all. The optimistic assumption allows us to quickly estimate  $E_{\text{save\_est\_i}}$  for  $V_{Li}$  without actually assigning voltages to gates as will be done by Algorithm 2 described next. Thus, Algorithm 1 selects a  $V_L$  with highest potential to save energy. Experiments in Section 7 will verify this strategy.

In Algorithm 1, gate slacks are computed only once for  $V_H$  assigned to all gates. Gate delay increments  $d_l - d_h$  are computed for all gates and for each value  $V_{Li}$  to repeatedly obtain Group 1, 2 and 3 classifications. Since the number of voltages  $V_{Li}$  in the range  $\{V_{th}, V_H\}$  need not grow with the number of gates in the circuit, the complexity of this algorithm is based upon slack calculation at a single voltage,  $V_H$ . This complexity is linear, i.e.,  $O(n)$  for  $n$  gates in the circuit.<sup>21–25</sup>

### 5.3. Algorithm 2

Having found an optimum value of  $V_L$  from Algorithm 1, Algorithm 2 assigns this low voltage to largest number of gates in the given circuit (specified by netlist and gate delays), such that no gate has a negative slack.

*Step 1:* Initially assign all gates to high voltage  $V_H$ . Calculate all gate slacks and the slack threshold  $S_u$  (Eq. (6)) if not already available from Algorithm 1. Classify gates into Groups 1, 2 and 3. Note that the three regions in the graph in Figure 1 remain unchanged throughout this algorithm. Delay increments also remain unchanged. Therefore, only gate slacks will be repeatedly calculated.

*Step 2:* Assign  $V_L$  to all Group 1 gates. Theorem II mandates that no negative slack occurs by this voltage assignment.

*Step 3:* Check topological constraints (see Section 6), i.e., if any  $V_L$  gate is driving a  $V_H$  gates, then change it to  $V_H$ . Recalculate slacks and reclassify gates into groups.

*Step 4:* Using the leveled netlist of the circuit, starting from the primary outputs, select a set of  $V_H$  gates from Group 2 satisfying the condition stated in Theorem III and assign them the low voltage  $V_L$ .

*Step 5 (similar to Step 3):* Check topological constraints, i.e., if any  $V_L$  gates is driving a  $V_H$  gates, change it to  $V_H$ . Recalculate slacks and reclassify gates into groups.

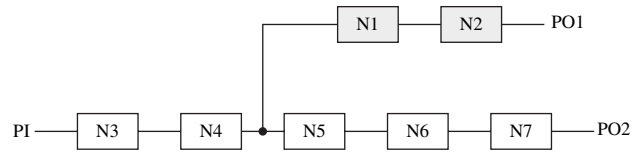
*Step 6:* Repeat Steps 4 and 5 until all  $V_H$  gates in Groups 1 and 2 have a topological constraint, i.e., they are feeding into other  $V_H$  gates.

Algorithm 2 iterates on the Group 2 gates whose number is proportional to all gates in the circuit. Each iteration uses a linear time slack calculation algorithm. Thus, its worst-case complexity is quadratic, i.e.,  $O(n^2)$  for  $n$  gates in the circuit.

## 6. TOPOLOGICAL CONSTRAINTS

In a multi-voltage design, when a lower voltage signal feeds into a higher voltage gate, the operation of the latter requires a careful examination. Because of lower driving input (typically when the input signal is logic 1), the high voltage gate may have higher leakage and a noisy output. To remedy the situation, if level converters are used, then their delay and power overheads must be accounted for in the dual-voltage design. In the present work, we avoid the use of level converters by not allowing a low voltage gate to feed into a high voltage gate. This condition is termed as topological constraint and its choice is justified in this section.

According to Theorem II, all Group 1 gates ( $G$  in Figure 1) have slacks greater than  $S_u$  and they can be simultaneously assigned the lower supply voltage because such assignment will not cause negative slack for any gate. The proof of Theorem II considers the entire longest delay path from primary input to primary output for a Group 1 gate. It is found that when all gates on this path



**Fig. 4.** Group 1 gates  $N1$  and  $N2$  with slack greater than  $S_u$  form a partial path and are assigned low voltage (indicated by shading) as they do not violate the topological constraint.

are assigned to low voltage no gate in the entire circuit will have negative slack. This is a pessimistic condition because, in general, the longest path can also have gates that do not belong to Group 1 and hence will not be set to low voltage.

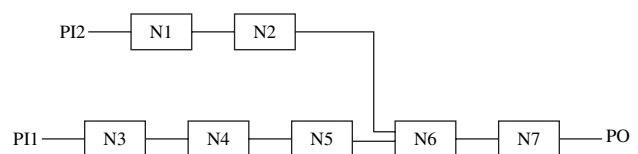
When all gates on a input to output path belong to Group 1 they can all be set to low voltage without violation the topological constraint. However, when a path only partially contains Group 1 gates, there are instances where some Group 1 gates cannot be assigned the low voltage due to the topological constraint as following examples illustrate.

Figures 4 and 5 show paths between primary inputs (PI) and primary outputs (PO). Each block is a gate with some delay. In both figures, suppose the slack of  $N1$  and  $N2$ , controlled by the shorter four-gate path, is less than  $S_u$  and so these gates belong to Group 1. Gates  $N3$  through  $N7$  are on a longer five-gate path, giving them a lower slack, which excludes them from Group 1. By Theorem II, only  $N1$  and  $N2$  can be simultaneously assigned to a lower voltage without causing a negative slack for any gate.

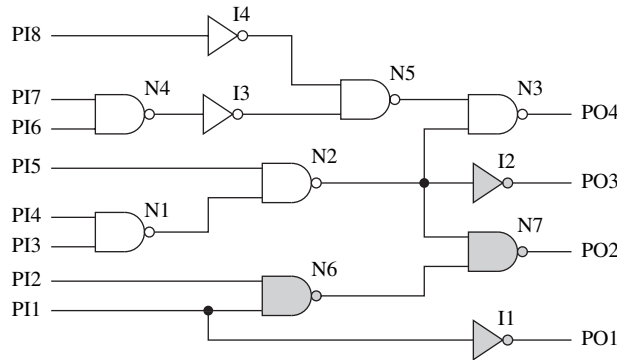
Next, considering the topological constraint that forbids a low voltage gate from feeding into a high voltage gate, Group 1 gates  $N1$  and  $N2$  will be assigned low voltage only in Figure 4 but not in Figure 5 (see Step 3 of Algorithm 2).

Suppose gates  $N3$  through  $N7$  belong to Group 2, i.e., some of them, though not all, can be assigned to low voltage in Steps 4–6 of Algorithm 2. Considering that the topological constraint is not violated if we select gates starting from a primary output, Step 4 uses an output to input strategy. In Figure 5 if  $N7$  and  $N6$  get assigned to low voltage then Group 1 gates  $N1$  and  $N2$  also become eligible for low voltage in Step 6.

When dual-voltage combinational blocks using topological constraints have to be interfaced with other



**Fig. 5.** Group 1 gates  $N1$  and  $N2$  with slack greater than  $S_u$  form a partial path but are not assigned low voltage because that will violate the topological constraint.



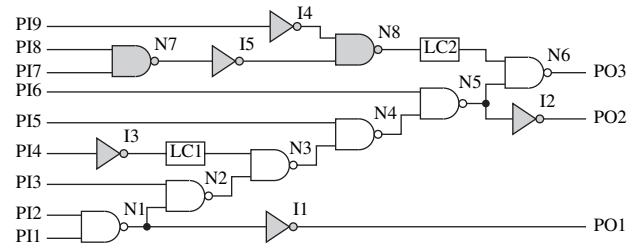
**Fig. 6.** A dual voltage circuit using topological constraint and no level converter. Gates  $N1$ ,  $N2$  and  $N3$  form the critical path. Gates shown with shading are assigned low voltage.

combinational blocks operating at higher supply voltages, level converting flip-flops and buffers are used at the inputs and outputs of a dual-voltage block to account for the changed logic levels. The design of level converting flip-flops is studied widely. Interested readers can refer to [14, 17, 26, 33, 35, 36, 45].

The restrictions on the circuit topology can be lifted by using a level converter at the interface where a low voltage gate has a high voltage gate at its fanouts. Many level converter designs have been proposed [5, 12, 15, 17–20, 27, 29, 30, 32, 44, 49]. A recent study [4] shows that the use of level converters is associated with delay and energy overheads and in many cases can reduce the possible energy saving. For circuit structures like c880, when we used Algorithm 2 but allowed level converters by removing topological constraints, we found that the energy savings were only 48.45% [4] as compared to 58.29% with the topological constraint (see Table II). For c6288 circuit, we found that the energy saving increased to 7.82% with level-converters as compared to 3.26% with topological constraint [4].

In general, benefits of topological constraint versus level converters are highly circuit dependent. The circuit of Figure 6 is a case for the former. Gates  $N1$ ,  $N2$  and  $N3$  form the critical path because  $N2$  has a large delay due to fanouts. Gates  $N4$ ,  $N5$  and  $I3$ , being on a near critical path, have small slacks and cannot be assigned low voltage. Gates  $N6$ ,  $N7$ ,  $I1$  and  $I2$  are in low voltage. Only  $I4$  has large slack but must keep high voltage because of topological constraint. If we were to use level converter, only gate affected is  $I4$  though it is doubtful whether the benefit of low voltage assignment to  $I4$  will offset the power and delay penalties of the level converter.

Next, consider the circuit of Figure 7, where there is one critical path ( $N1$  through  $N6$ ) and many shorter delay paths that feed into the critical path. Very few low voltage gates such as  $I1$  and  $I2$  satisfy the topological constraint. Most other low voltage gates ( $I3$ ,  $I4$ ,  $I5$ ,  $N7$  and  $N8$ ) can only be assigned low voltage if level converters LC1 and LC2 are inserted. Note that the delay and power overhead



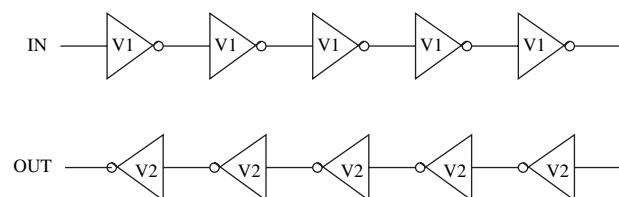
**Fig. 7.** A dual voltage circuit using level converters (LC1 and LC2). Gates  $N1$  through  $N6$  form the critical path. All shaded gates are assigned low voltage.

of each level converter must be balanced against the benefit it provides. Thus, LC1 that provides power saving only due to one gate  $I3$ , may not be useful.

Consider the chain of inverters shown in Figure 8. We simulated this circuit using Synopsys HSPICE program [2] with voltages  $V1$  and  $V2$  as 0.4 V, 0.6 V, 0.8 V, 1.0 V and 1.2 V. A 1 GHz 50% duty-cycle clock was applied at the input and a capacitance of 6fF, equivalent to four inverters, was used as the load at the output. The results for 90 nm PTM [1] are presented in Figure 9. It reports the total energy consumption and delay for the circuit at various values of  $V1$  and  $V2$ .

The energy values shown in the diagonal squares are for  $V1 = V2$  and correspond to single voltage operation. The values in the lower triangle are for  $V1 > V2$ , i.e., when a higher voltage gate is feeding a lower voltage gate. The upper triangle represent gives operation when  $V2 > V1$ , i.e., when a low voltage gate feeds a high voltage gate. We observe that the delay measurement in two of the top cells fails as shown by an infinite delay for large voltage. For all cases above the diagonal, although logic 1 level matched higher supply voltage, logic 0 levels for the five inverters near the output were higher than ground. That produced significantly higher leakage. This indicates the necessity for level conversion at the voltage boundary. However, the designs of such devices are still evolving and problems with their performance have been reported. Especially, their performance in terms of power and delay overheads deteriorates as the difference between the two voltages increases, i.e., when they are needed most.

For all cases where a high voltage gate feeds a low voltage gate, energy savings are seen. These results demonstrate the effectiveness of using a suitable topological constraint.



**Fig. 8.** A chain of ten inverters.

1.2	6.344fj	31.23fj	12.31fj	7.008fj	7.863fj
	$\infty$	282.9ps	123.3ps	95.66ps	84.11ps
1.0	4.279fj	7.861fj	4.442fj	5.199fj	6.606fj
	$\infty$	203.7ps	123.3ps	99.46ps	91.05ps
0.8	4.749fj	2.568fj	3.233fj	4.283fj	5.656fj
	1179ps	203.1ps	132.1ps	115ps	107.6ps
0.6	1.261fj	1.757fj	2.532fj	3.566fj	4.932fj
	796.1ps	234ps	179.3ps	164.4ps	155.8ps
0.4	0.753fj	1.280fj	2.036fj	3.074fj	4.443fj
	1065ps	614.1ps	565ps	561ps	557.6ps
	0.4	0.6	0.8	1.0	1.2

Fig. 9. Energy and delay measurements at various values of  $V_1$  and  $V_2$  for the inverter chain of Figure 8.

7. EXPERIMENTAL RESULTS

We used ISCAS’85 benchmark circuits for experiments. Our circuits were synthesized using a small set of 90nm standard cells consisting of inverter, INV, two-input NAND gate, NAND2, three-input NAND gate, NAND3, and two-input NOR gate NOR2. The cells were characterized for 90 nm bulk PTM<sup>1</sup> CMOS, 0.3 V threshold voltage and room temperature using Synopsys HSPICE program.<sup>2</sup> For supply voltages ranging from 0.4 V to 1.2 V in 0.01 V steps, cell delays and output node capacitance data were tabulated for output fanout load varying from 1 to 4 inverters. This cell data allowed us to obtain the delay of each gate in a dual-voltage circuit for logic simulation, which would determine the number of signal transitions at each node (gate output) for given stimuli. Dynamic energy consumed by a gate is then computed as the product of its output transitions, output capacitance, supply

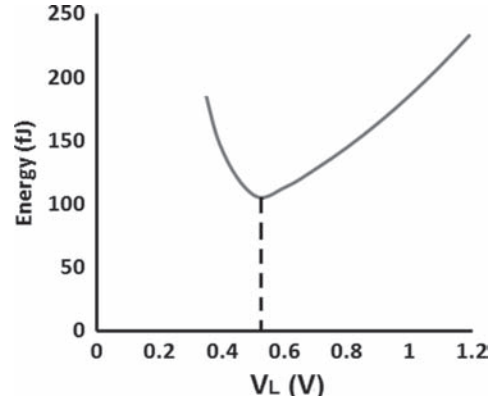


Fig. 10. Energy versus  $V_L$  for c880 with gate voltages assigned by Algorithm 2.  $V_H = 1.2$  V and clock period  $T_c$  equals critical path delay when all gates are assigned  $V_H$ .

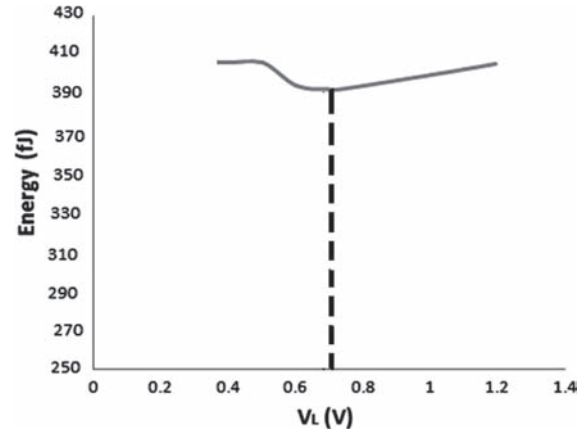


Fig. 11. Energy versus  $V_L$  for c1355 with gate voltages assigned by Algorithm 2.  $V_H = 1.2$  V and clock period  $T_c$  equals critical path delay when all gates are assigned  $V_H$ .

voltage squared, and 0.5. Each circuit was simulated in HSPICE with 100 randomly generated input vectors to determine the average node activity  $\{\alpha_i\}$  for all nodes  $i$ , to be used for actual energy calculation from simulation.

Table I. Optimal lower supply voltage  $V_L$  for  $V_H = 1.2$  V obtained from Algorithm 1 and energy saving estimate by Eq. (14) for ISCAS’85 benchmark circuits.

Benchmark circuit	Total gates	Algorithm 1			$V_L = 0.7 \times V_H = 0.84$ volt		$V_L = 0.5 \times V_H = 0.6$ volt	
		$(V_L)$ (volt)	Gates in low voltage	$E_{save\_est}$ Eq. (14) (%)	Gates in low voltage	$E_{save\_est}$ Eq. (14) (%)	Gates in low voltage	$E_{save\_est}$ Eq. (14) (%)
c432	154	0.8	8	2.90	8	2.65	8	3.90
c499	493	0.76	113	13.73	121	12.52	56	8.52
c880	360	0.49	213	49.30	229	32.44	229	47.71
c1355	469	0.77	76	9.53	76	8.27	64	10.24
c1908	584	0.60	221	28.38	221	19.3	221	28.40
c2670	901	0.48	570	53.14	570	32.27	570	47.45
c3540	1270	0.52	149	9.53	149	5.98	149	8.80
c5315	2077	0.49	1220	48.95	1240	30.45	1220	44.05
c6288	2407	0.55	75	2.46	77	1.63	75	2.34
c7552	2823	0.54	1582	44.69	2359	42.62	1672	43.92
Average			422.7	26.261	505	18.813	426.2	24.533



**Table II.** Optimal lower supply voltage values ( $V_L$ ) and energy savings using Algorithms 1 and 2 for ISCAS'85 benchmark circuits, when clock period  $T_c$  equals critical path delay of circuit with single voltage,  $V_H = 1.2$  V.

Benchmark circuit	Total gates	Algorithms 1 and 2				HSPICE <sup>2</sup> energy computation			Reference <sup>23</sup>	
		$V_L$ (volts)	No. of $V_L$ gates	$E_{save\_est}$ Eq. (14) (%)	CPU* (s)	$E_{singleVDD}$ (fJ)	$E_{dualVDD}$ (fJ)	$E_{save\_act}$ (%)	$E_{save\_act}$ (%)	CPU* (s)
c432	154	0.8	8	2.9	1.78	161.3	155.4	3.66	3.9	15.8
c499	493	0.76	113	13.73	9.41	463.0	427.0	7.8	5.9	194.4
c880	360	0.49	213	49.3	5.39	277.6	115.8	58.29	50.8	62.1
c1355	469	0.77	76	9.53	8.75	455.2	433.1	4.86	4.3	132
c1908	584	0.60	221	28.38	11.43	496.5	378.3	23.81	19.0	247.8
c2670	901	0.48	570	53.14	23.49	660.3	251.5	61.9	47.8	480.7
c3540	1270	0.52	149	9.53	45.44	1843	1620	12.23	9.6	1244
c5315	2077	0.49	1220	48.95	109.47	2320	1272	45.17	NA	NA
c6288	2407	0.55	75	2.46	154.94	1932	1869	3.26	2.6	6128
c7552	2823	0.54	1582	44.69	191.04	2465	1562	36.63	NA	NA
Average				26.26				25.76		17.99

Notes: \*Intel core i5 2.30 GHz, 4 GB RAM.

Node capacitances  $\{C_i\}$  for all nodes were also extracted for actual energy calculation from simulation.

For  $V_H = 1.2$  V, Algorithm 1 was used to determine  $V_L$  for each ISCAS'85 benchmark circuit. Table I gives the result and the estimated energy saving computed from Eq. (14) with number of low voltage gates as the sum of Group 1 and 2 gates. For comparison, the table also gives energy saving corresponding to  $V_L = 0.7V_H$  and  $V_L = 0.5V_H$ , two values suggested in the literature. It is observed that the expected energy saving is larger for most circuits when we use  $V_L$  given by Algorithm 1.

As pointed out in Section 5 Algorithm 1 selects  $V_L$  using an optimistic assumption that all Group 1 and 2 gates could be assigned low voltage. In reality, this depends upon circuit topology. To justify the assumption, we examine two cases. A circuit c880, which has fewer long paths and can be optimized to obtain a considerably high energy saving. Another circuit c1355, which has a large number of paths with delays close to that of the critical path and is difficult to optimize. Figures 10 and 11 show the energy as a function of  $V_L$  for dual voltage designs of c880 and c1355, respectively, where  $V_H = 1.2$  V. The energy was calculated as follows:

$$E_{dualVDD} = 0.5 \sum_{i=1}^p \alpha_i \times C_i \times V_H^2 + 0.5 \sum_{i=1}^q \alpha_i \times C_i \times V_L^2$$

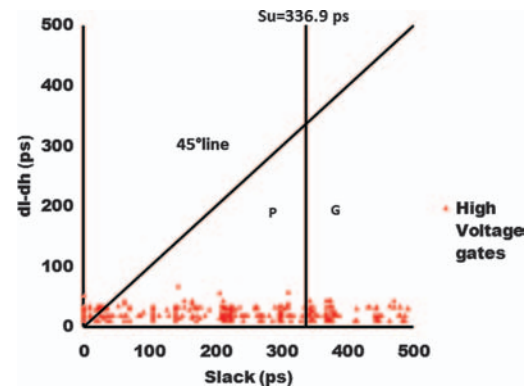
where  $\alpha_i$  is the average activity and  $C_i$  is the capacitance of the  $i$ th node,  $p$  is the number of gates in high voltage and  $q$  is the number of gates in low voltage after voltage assignment by Algorithm 2.

We set  $V_L$  to successive values between the threshold voltage and 1.2 V and find the energy saving in each case for c880 and c1355. From these graphs,  $V_L$  for minimum energy is 0.5 V for c880 and 0.7 V for c1355. We observe that these values are close to optimum  $V_L$  values obtained from Algorithm 1 as reported in Table I, which are 0.49 V and 0.77 V, respectively.

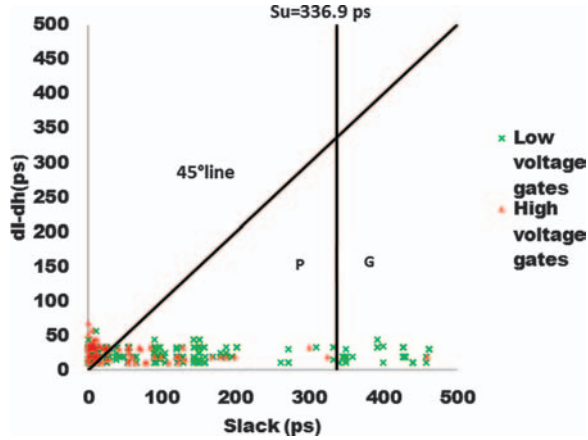
In Table II,  $E_{singleVDD}$  and  $E_{dualVDD}$  are HSPICE<sup>2</sup> results for average energy per vector for single voltage design and that for dual-voltage design, respectively, obtained by simulating a set of 100 random vectors. The actual energy savings reported by HSPICE<sup>2</sup> for dual-voltage design is  $E_{save\_act}$ . Also,  $E_{save\_est}$  is the maximum energy saving estimated by Algorithm 1 to select  $V_L$ .

Table II shows that the actual energy savings  $E_{save\_act}$  are generally close to the estimated values  $E_{save\_est}$  obtained from Eq. (14). Note that the saving estimate of Eq. (14) is optimistic because all Group 2 gates are assumed to be in low voltage when in reality only a subset of those are assigned low voltage by Algorithm 2. Still in some cases, such as c880, the actual saving is greater. One possible reason is the reduction of glitches in the dual voltage circuit due to near balancing of paths.

The last two columns of Table II give results of a previously published slack-based algorithm<sup>23</sup> that also uses topological constraints but, as discussed in Section 4, the proposed algorithm provides higher energy saving and lower computation complexity.



**Fig. 12.** Gate delay increment for  $V_L = 0.49$  V versus gate slack for all gates in c880 when all gates are assigned  $V_H = 1.2$  V. Clock period  $T_c$  equals critical path delay.  $V_L$  was obtained by Algorithm 1.



**Fig. 13.** Gate delay increment versus final slack for gates of c880 circuit after voltage assignment by Algorithm 2,  $V_H = 1.2$  V,  $V_L = 0.49$  V.  $V_L$  was obtained by Algorithm 1.

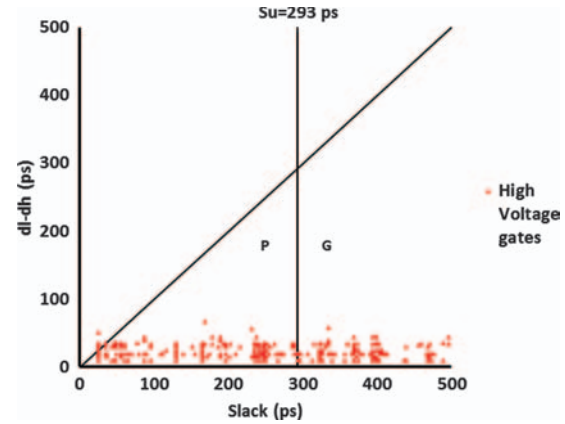
Figures 12 and 13 show delay increment versus slack graphs for the all- $V_H$  design and the dual-voltage design, respectively, for c880 circuit. The triangular markers (in red) indicate gates in high voltage and cross markers (in green) indicate gates in low voltage. In Figure 12 all gates are in high voltage. After Algorithms 1 and 2 were used and slacks recalculated, as shown in Figure 13 most gate slacks are reduced. All high voltage gates tend to concentrate at lower slack values and many gates have now moved above the 45° line.

In Figure 13, there are some low voltage gates below the 45° line, i.e., in Groups 1 and 2. These are gates with very large initial slack. A few high voltage gates (red triangles) still lying below the 45° line are the gates that could not be assigned low voltage due to topological constraints imposed by Algorithm 2. Also, we can see a few triangular markers (in red) to the right of  $S_u$  line. These are again the gates that cannot be put in low voltage due to topological constraints, but have slacks greater than  $S_u$ , as explained in Section 6.

**Table III.** Optimal lower supply voltage values ( $V_L$ ) and energy savings using Algorithms 1 and 2 for ISCAS'85 benchmark circuits, when clock period  $T_c$  is 5% greater than the critical path delay of the circuit with single voltage,  $V_H = 1.2$  V.

Benchmark circuit	Total gates	Algorithms 1 and 2				HSPICE <sup>2</sup> energy computation		
		$V_L$ (volts)	No. of $V_L$ gates	$E_{save\_est}$ Eq. (14) (%)	CPU* (s)	$E_{singleVDD}$ (fJ)	$E_{dualVDD}$ (fJ)	$E_{save\_act}$ (%)
c432	154	1.08	154	19.00	1.70	161.3	123.9	23.19
c499	493	1.03	493.0	26.33	9.18	463	321.9	30.48
c880	360	0.67	344	65.77	4.32	277.6	83.86	69.79
c1355	469	1.06	469	21.97	8.52	455.2	339.9	12.15
c1908	584	1.00	584	30.56	8.56	496.5	445.0	10.37
c2670	901	0.81	899	54.32	15.81	660.3	257.3	61.03
c3540	1270	0.90	1270	43.75	28.22	1843	949.5	48.48
c5315	2077	0.72	2077	64.00	61.77	2320	716.8	69.11
c6288	2407	1.07	2407	20.49	108.39	1932	1464	24.22
c7552	2823	0.68	2816	67.72	175.07	2465	677.2	72.28
Average				41.39			42.11	

Notes: \*Intel core i5 2.30 GHz, 4 GB RAM.

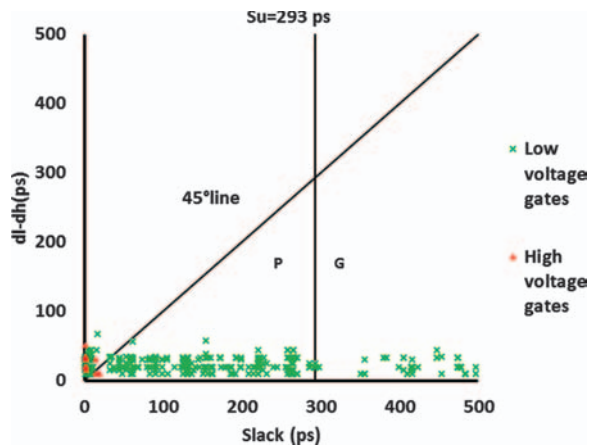


**Fig. 14.** Gate delay increment for  $V_L = 0.67$  V versus gate slack for all gates assigned  $V_H = 1.2$  V in c880. Clock period  $T_c$  is 5% longer than critical path delay.  $V_L$  was obtained by Algorithm 1.

In the results described thus far, we used a clock period  $T_c$  that was equal to the critical path delay of the all- $V_H$  circuit. In general, greater energy saving is possible if the circuit is slowed down. Next, we apply Algorithms 1 and 2 to ISCAS'85 benchmark circuits allowing a 5% increase in the clock period  $T_c$ .

The results are shown in Table III. We note that Algorithm 1 now selects a higher values for  $V_L$  but Algorithm 2 assigns  $V_L$  to a larger number of gates providing higher overall energy saving. For example, consider c880 in Table III. Algorithm 1 gives  $V_L = 0.67$  V, which is assigned to 344 out of 360 gates, providing 69.79% actual energy saving. In comparison, when  $T_c$  is not to exceed the all- $V_H$  critical path delay, Table II has  $V_L = 0.49$  V assigned to 213 gates, with an actual energy saving of 58.29%. CPU times in tables are comparable.

Figures 14 and 15 show delay increment versus slack graphs for the initial and final slacks, respectively, for c880 circuit when clock period  $T_c$  is 5% longer than the critical path delay of the all- $V_H$  circuit and  $V_L = 0.67$  V.



**Fig. 15.** Gate delay increment versus gate slack after all gates of c880 were assigned either  $V_H = 1.2$  V or  $V_L = 0.67$  V by Algorithm 2. Clock period  $T_c$  was 5% longer than the critical path delay of all- $V_H$  circuit.  $V_L$  was obtained by Algorithm 1.

In Figure 14 all gates are assigned  $V_H = 1.2$  V. Comparing with Figure 12, slacks are increased and  $S_u$ , now 293 ps, is lower. This puts larger number of gates in Groups 1 and 2, and fewer in Group 3. From the graphs it can be seen that the slacks of the gates have moved towards the right due to increased critical path delay, which in turn increases the gate slacks. Also, the final number of gates in high voltage is lower, which can be seen by the reduced density of triangle-shaped (red) dots. Although not obvious from graphs, Figure 15 has 344  $V_L$  gates as compared to 213 gates in Figure 13.

## 8. CONCLUSION

This work introduces two new algorithms for dual voltage design. Given a voltage  $V_H$ , the first algorithm finds an optimal voltage  $V_L$  using an  $O(n)$  algorithm (for  $n$  gates) to compute the slacks of all gates. The second algorithm determines a set of gates that can be assigned  $V_L$  without violating the positive slack constraint. The gates are divided into groups based on their slack and the delay increase due to the lower voltage. Energy savings of up to 60% were observed by this method. Also, the results are obtained at lower CPU times than the previously published results.<sup>23,28</sup> Here we use the  $O(n)$  complexity slack calculation algorithm iteratively. If we put one gate at a time to low voltage the complexity of this algorithm will be  $O(n^2)$ . In practice, it is observed to be close to linear time. This is because we take groups of gates at a time for low-voltage assignment.

The first algorithm searches for a voltage  $V_L$  in the range between the threshold voltage and a given supply voltage  $V_H$ . Efficient search algorithms, e.g., binary search, etc., can be explored in the future.

The second algorithm uses topological constraints producing a dual-voltage design that does not use level converter. Alternative algorithms that do not impose

topological constraints are possible. These must account for the delay and energy consumption of the specific type of level converters used in the design.<sup>4</sup>

## References

1. Latest PTM Models, <http://ptm.asu.edu/> (accessed on December 11, 2011).
2. HSPICE registered Reference Manual: Commands and Control Options, Version D-2010.03-SP1, June 2010, <http://www.synopsys.com/Tools/Verification/AMSVVerification/CircuitSimulation/HSPICE/Pages/default.aspx> (accessed on October 14, 2011).
3. M. Allani, Polynomial-time algorithms for designing dual-voltage energy efficient circuits, Master's Thesis, ECE Department, Auburn University, Auburn, AL, December (2011).
4. M. Allani, V. D. Agrawal, An Efficient algorithm for dual-voltage design without need for level conversion, *Proc. 44th IEEE Southeastern Symp. System Theory*, March (2012), pp. 51–56.
5. J.-Y. An, H. S. Park, and Y. H. Kim, Level up/down converter with single power supply voltage for multi  $V_{DD}$  Systems. *Journal of Semiconductor Technology and Science* 10, 55 (2010).
6. S. Augsburger and B. Nikolic, Combining dual-supply, dual-threshold and transistor sizing for power reduction, *Proceedings of the IEEE International Conference on Computer Design* (2002), pp. 316–321.
7. N. Chabini, I. Chabini, E. M. Aboulhamid, and Y. Savaria, Methods for minimizing dynamic power consumption in synchronous designs with multiple supply voltages. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 22, 346 (2003).
8. A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, Low-power cmos digital design. *IEEE Journal of Solid-State Circuits* 27, 473 (1992).
9. C. Chen, A. Srivastava, and M. Sarrafzadeh, On gate level power optimization using dual-supply voltages. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 9, 616 (2001).
10. D. Chen, J. Cong, and J. Xu, Optimal module and voltage assignment for low-power, *Proceedings of the Asia and South Pacific Design Automation Conference* (2005), pp. 850–855.
11. D. G. Chinnery and K. Keutzer, Linear programming for sizing,  $v_{th}$  and  $v_{dd}$  assignment, *Proceedings of the International Symposium on Low Power Electronics and Design* (2005), pp. 149–154.
12. D. G. Chinnery and K. Keutzer, Closing the Power Gap Between ASIC and Custom—Tools and Techniques for Low Power Design, Springer (2007).
13. M. Hamada, Y. Ootaguro, and T. Kuroda, Utilizing surplus timing for power reduction, *Proceedings of the IEEE Custom Integrated Circuits Conference* (2001), pp. 89–92.
14. M. Hamada, M. Takahashi, H. Arakida, A. Chiba, T. Terazawa, T. Ishikawa, M. Kanazawa, M. Igarashi, K. Usami, and T. Kuroda, A top-down low power design technique using clustered voltage scaling with variable supply-voltage scheme, *Proceedings of the IEEE Custom Integrated Circuits Conference* (1998), pp. 495–498.
15. S. K. Han, K. Park, B.-S. Kong, and Y.-H. Jun, High speed low power bootstrapped level converter for dual supply systems, *Proc. IEEE Asia Pacific Conference on Circuits and Systems* (2010), pp. 871–874.
16. Y. Hu, Y. Lin, L. He, and T. Tuan, Simultaneous time slack budgeting and retiming for dual- $V_{dd}$  fpga power reduction, *Proceedings of the 43rd Annual Design Automation Conference* (2006), pp. 478–483.
17. F. Ishihara, F. Sheikh, and B. Nikolic, Level conversion for dual supply systems. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 12, 185 (2004).
18. M. C. Johnson and K. Roy, Optimal selection of supply voltages and level conversions during data path scheduling under resource constraints, *Proceedings of the International Conference on Computer Design* (1996), pp. 72–77.

19. M. C. Johnson and K. Roy, Datapath scheduling with multiple supply voltages and level converters. *ACM Transactions on Design Automation of Electronic Systems* 2, 227 (1997).
20. M. Keating, D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low Power Methodology Manual: For System-on-Chip Design*, Springer (2007).
21. K. Kim, Ultra Low power cmos design, Ph.D. Thesis, ECE Department, Auburn University, Auburn, AL (2011).
22. K. Kim and V. D. Agrawal, Proceedings of the 24th International Conference on VLSI Design.
23. K. Kim and V. D. Agrawal, Dual voltage design for minimum energy using gate slack. *Proceedings of the IEEE International Conference on Industrial Technology*, March (2011), pp. 419–424.
24. K. Kim and V. D. Agrawal, Minimum energy cmos design with dual subthreshold supply and multiple logic-level gates. *Proceedings of the 12th International Symposium on Quality Electronic Design* March (2011), pp. 689–694.
25. K. Kim and V. D. Agrawal, Ultra low energy cmos logic using below threshold dual voltage supply. *Journal of Low Power Electronics* 7, 460 (2011).
26. B.-S. Kong, S.-S. Kim, and Y.-H. Jun, Conditional-capture flip-flop for statistical power reduction. *IEEE Journal of Solid-State Circuits* 36, 1263 (2001).
27. K.-H. Koo, J.-H. Seo, M.-L. Ko, and J.-W. Kim, A new level up shifter for high speed and wide range interface in ultra deep sub micron. *Proc. IEEE International Symposium on Circuits and Systems* 2, 1063 (2005).
28. S. H. Kulkarni, A. N. Srivastava, and D. Sylvester, A new algorithm for improved  $v_{dd}$  assignment in low power dual  $V_{DD}$  systems. *Proceedings of the International Symposium on Low Power Design* (2004), pp. 200–205.
29. S. H. Kulkarni and D. Sylvester, Fast and energy efficient asynchronous level converters for multi- $v_{dd}$  design cmos ics]. *Proceedings of IEEE International Systems on Chip (SOC) Conference* September (2003), pp. 169–172.
30. M. Kumar, S. K. Arya, and S. Pandey, Level shifter design for low power applications. *International Journal of Computer Science and Information Technology* 2, 124 (2010).
31. T. Kuroda and M. Hamada, Low-power cmos digital design with dual embedded adaptive power supplies. *IEEE Journal of Solid-State Circuits* 35, 652 (2000).
32. V. Kursun and E. G. Friedman, *Multi-Voltage CMOS Circuit Design*, John Wiley & Sons (2006).
33. D. Li, P. Chuang, and M. Sachdev, Comparative analysis and study of metastability on high performance flip-flops. *Proc. 11th International Symposium on Quality Electronic Design* (2010), pp. 853–860.
34. Y. Lin and L. He, Dual- $V_{dd}$  interconnect with chip-level time slack allocation for fpga power reduction. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 25, 2023 (2006).
35. H. Mahmoodi-Meimand and K. Roy, Self-precharging flip-flop (spff): A new level converting flip-flop. *Proceedings of the 28th European Solid-State Circuits Conference* (2002), pp. 407–410.
36. B. Nikolic, V. G. Oklobdzija, V. Stojanovic, W. Jia, J. K.-S. Chiu, and M. M.-T. Leung, Improved sense-amplifier-based flip-flop: Design and measurements. *IEEE Journal of Solid-State Circuits* 35, 876 (2000).
37. M. Pedram and J. M. Rabaey, *Power Aware Design Methodologies*, Springer (2002).
38. K. Roy, L. Wei, and Z. Chen, Multiple  $V_{dd}$  multiple  $V_{th}$  CMOS (MVC MOS) for low power applications. *Proceedings of the IEEE International Symposium on Circuits and Systems* (1999), Vol. 1, pp. 366–370.
39. V. Sandrarajan and K. K. Parhi, Synthesis of low power cmos vlsi circuits using dual supply voltages. *Proceedings of the 36th Design Automation Conference* (1999), pp. 72–75.
40. F. Sheikh, A. Kuehlmann, and K. Keutzer, Minimum-power retiming for dual-supply CMOS circuits. *Proceedings of the 8th ACM/IEEE International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems* (2002), pp. 43–49.
41. D. Shin, J. Kim, and S. Lee, Low-energy intra-task voltage scheduling using static timing analysis. *Proceedings of the 38th Annual Design Automation Conference* (2001), pp. 438–443.
42. A. Srivastava, D. Sylvester, and D. Blaauw, Concurrent sizing,  $V_{dd}$  and  $V_{th}$  assignment for low-power design. *Proceedings of the Design, Automation and Test in Europe Conference* (2004) pp. 10718–10719.
43. A. Srivastava, D. Sylvester, and D. Blaauw, Power minimization using simultaneous gate sizing, dual- $V_{dd}$  and dual- $V_{th}$  assignment. *Proceedings of the Design Automation Conference* (2004), pp. 783–787.
44. C. Q. Tran, H. Kawaguchi, and T. Sakurai, Low power high speed level shifter design for block level dynamic voltage scaling environment. *Proc. International Conference on Integrated Circuit Design and Technology* (2005), pp. 229–232.
45. J. Tschanz, S. Narendra, Z. Chen, S. Borkar, M. Sachdev, and V. De, Comparative delay and energy of single edge-triggered and dual edge-triggered pulsed flip-flops for high-performance microprocessors. *Proc. International Symposium on Low Power Electronics and Design* (2001), pp. 147–152.
46. K. Usami and M. Horowitz, Clustered voltage scaling technique for low-power design. *Proceedings of the International Symposium on Low Power Design* (1995), pp. 23–26.
47. K. Usami and M. Igarashi, Low-power design methodology and applications utilizing dual supply voltages. *Proceedings of the Asia and South Pacific Design Automation Conference* (2000). pp. 123–128.
48. K. Usami, M. Igarashi, F. Minami, T. Ishikawa, M. Kanzawa, M. Ichida, and K. Nogami, Automated low-power technique exploiting multiple supply voltages applied to a media processor. *IEEE Journal of Solid-State Circuits* 33, 463 (1998).
49. S. N. Wooters, B. H. Calhoun, and T. N. Blalock, An energy efficient subthreshold level converter in 130 nm CMOS. *IEEE Transactions on Circuits and Systems II: Express Briefs* 57, 290 (2010).
50. C. Yeh, M.-C. Chang, S.-C. Chang, and W.-B. Jone, Gate-level design exploiting dual supply voltages for power-driven applications. *Proceedings of the Design Automation Conference* (1999), pp. 68–71.
51. Y.-J. Yeh and S.-Y. Kuo, An optimization-based low-power voltage scaling technique using multiple supply voltages. *Proceedings of the IEEE International Symposium on Circuits and Systems* (2001), Vol. 5, pp. 535–538.

**Mridula Allani**

Mridula Allani received B.E. in Electronics and Communications Engineering from Osmania University, India and an M.S. in Electrical Engineering from Auburn University. She works on low power DFX design at Intel Corporation in Austin, Texas.

**Vishwani D. Agrawal**

Vishwani D. Agrawal is the James J. Danaher Professor of Electrical and Computer Engineering at Auburn University, Alabama, USA. He has over forty years of industry and university experience, working at Bell Labs, Murray Hill, New Jersey; Rutgers University, New Brunswick, New Jersey; TRW, Redondo Beach, California; IIT, Delhi; EG&G, Albuquerque, New Mexico; and ATI, Champaign, Illinois. His areas of expertise include VLSI testing, low power design, and microwave antennas. He obtained his B.E. degree from the Indian Institute of Technology Roorkee, Roorkee, in 1964; ME degree from the Indian Institute of Science, Bangalore, in 1966; and PhD degree in electrical engineering from the University of Illinois at Urbana-Champaign, in 1971. He has published over 350 papers, has coauthored five books and holds thirteen United States patents. His textbook, *Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits* co-authored with M. L. Bushnell, was published in 2000. He is the founder and Editor-in-Chief (1990-) of the *Journal of Electronic Testing: Theory and Applications*, and a past Editor-in-Chief (1985-87) of the *IEEE Design and Test of Computers* magazine. He is the founder and Consulting Editor of the *Frontiers in Electronic Testing* Book Series of Springer. He is a co-founder of the *International Conference on VLSI Design*, and the *VLSI Design and Test Symposium*, held annually in India. He has served on numerous conference committees and is a frequently invited speaker. He was the keynote speaker at the 25th *International Conference on VLSI Design*, Hyderabad, in January 2012, an invited Plenary Speaker at the 1998 *International Test Conference*, Washington, D.C., and the Keynote Speaker at the *Ninth Asian Test Symposium*, Taiwan in 2000. During 1989 and 1990, he served on the Board of Governors of the *IEEE Computer Society*, and in 1994 chaired the *Fellow Selection Committee* of that Society. He has received eight Best Paper Awards and two Honorable Mention Paper Awards. He is the recipient of the 2012 *Lifetime Contribution Medal* from the *Test Technology Technical Council* of the *IEEE Computer Society*, and the 2006 *Lifetime Achievement Award* of the *VLSI Society of India* in recognition of his contributions to the area of VLSI test and for founding and steering the *International Conference on VLSI Design in India*. In 1998, he received the *Harry H. Goode Memorial Award* of the *IEEE Computer Society* for "innovative contributions to the field of electronic testing," and in 1993, received the *Distinguished Alumnus Award* of the *University of Illinois at Urbana-Champaign*, in recognition of his outstanding contributions in design and test of VLSI systems. Agrawal is a Fellow of *IETE-India*, a *Life Fellow* of the *IEEE*, and a *Fellow* of the *ACM*. He has served on Advisory Boards of ECE Departments of the *University of Illinois*, the *New Jersey Institute of Technology*, and the *City College of the City University of New York*. See his website: <http://www.eng.auburn.edu/~vagrawal>.