44th IEEE Southeastern Symposium on System Theory
University of North Florida, Jacksonville, FL
March 11-13, 2012

An Efficient Algorithm for Dual-Voltage Design Without Need for Level Conversion

Mridula Allani
Intel Corporation
1501 S. Mopac Expy., Suite 400, Austin, TX 78746
mridula.allani@intel.com

Vishwani Agrawal
Auburn University
200 Broun Hall, ECE Dept., Auburn, AL 36849
vagrawal@eng.auburn.edu

***Abstract*—We propose a technique to use dual supply voltages in digital designs to reduce energy consumption. New algorithms are proposed for finding and assigning a lower voltage in a dual voltage design. Given a circuit and a supply voltage and an upper bound on the critical path delay, the first algorithm finds an optimal lower supply voltage and a second algorithm assigns that lower voltage to selected gates. A linear time algorithm described in the literature is used for computing slacks for all gates of the circuit for a given supply voltage. For the computed gate slacks and the lower supply voltage, all gates are divided into three groups such that no gate in the first group can be assigned the lower supply, all gates in the second group can be simultaneously set to lower supply while maintaining positive slack for every gate, and gates in the third group are assigned low voltage, iteratively, in selected subsets at a time. The gate slacks are recalculated after each such voltage assignment. Thus, the overall complexity of this reduced power dual voltage assignment procedure is $O(n^2)$. SPICE simulations of ISCAS'85 benchmark circuits using the 90-nm bulk CMOS technology results show up to 60% energy savings.**

Index Terms: Dual-voltage design, Clustered Voltage Scaling (CVS), gate slack, critical path, level converter, topological constraints, energy savings.

I. Introduction

Power and performance are two main constraints in modern VLSI design. An effective design is obtained with the optimization of these two parameters. This requirement for higher performance and lower power directly results from the increasing demand for portable electronic devices. Decreasing the supply voltage results in decreased performance. Hence, a trade off is necessary between power consumption and circuit delay. The use of multiple supply voltages to reduce energy consumption is a very commonly used technique in CMOS circuits. This results from the fact that the dynamic power of a CMOS circuit is directly proportional to the square of its supply voltage [6], [17]. The underlying idea of this technique is to trade timing slacks for power reduction under given timing requirements. Generally, the gates in the critical path are kept at high supply voltage and the gates in non-critical paths are put to lower supply voltage, thus avoiding timing violation.

Usami and Horowitz [19] describe the clustered voltage scaling (CVS) technique in which the cells driven by each power supply are grouped together and use of level converter is avoided by not allowing a V_L gate feeding a V_H gate. The extended clustered voltage scaling (ECVS) proposed in [20] removes this restriction by allowing the usage of a level converter [15] at such boundaries. This is referred to as asynchronous level conversion. Both CVS and ECVS aim at utilizing the surplus timing of the non-critical paths in a circuit by applying a lower supply voltage on gates that are on these paths. This reduces dynamic power dissipation and hence lowers system level power dissipation. In our work, we do not use level converters, due to the overheads associated with them [4], [8], [11], [14].

Several other algorithms have been proposed in the literature for dual/multiple-voltage assignments modifying CVS and ECVS algorithms. A greedy algorithm (GECVS) is used in [14] to group gates for V_L assignment based on a sensitivity measure derived from the gate slacks, change in total power and delay. The authors claim to improve power savings up to 28% and 13% with respect to CVS and ECVS, respectively.

In [12], the authors use a mixed integer linear programming (MILP) technique to find the V_L value for circuits operating in the subthreshold region. Then an ECVS kind of method is used with multiple logic gates between the V_L and V_H boundaries instead of level converters.

There are many algorithms for assigning a fixed low voltage value to the gates of a circuit, but relatively fewer algorithms to find the lower voltage. Many of these works related to voltage assignment have used a V_L value of 70% of V_H [5], [9], [14], [16], [17]. The

978-1-4577-1491-7/12/$26.00 ©2012 IEEE

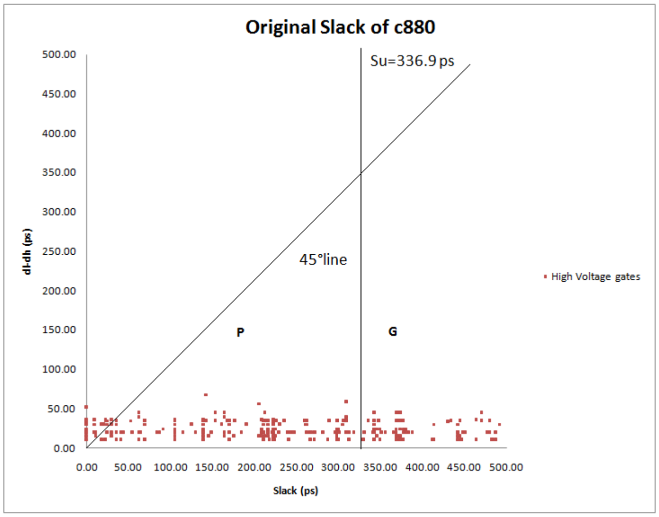


Fig. 1. Delay increment versus initial slack for gates of c880 circuit, $V_H = 1.2V,\ V_L = 0.49V$.

work described in [7], [14], [18] claims that the optimal value of V_L for minimizing total power is 50% of V_H. Authors in [21] describe an algorithm to find the lowest feasible supply voltages according to their slacks from a set of given voltages. An additional 19.55% power savings were obtained by this technique over the CVS method [19]. An algorithm to find optimum V_L value is described in [11]. The authors assign a low voltage value to a group of gates based on a modification of CVS algorithm and then calculate energy over a set of low voltages. The V_L value resulting in minimum energy is chosen to be the lower voltage. This algorithm requires the voltage assignment to be done for each voltage value and is exhaustive in nature.

In Section II, we propose an algorithm to find the lower supply voltage which maximizes the power savings obtained by the number of gates in a specific group. As determined by their slacks these gates can be assigned the lower supply voltage without violating the critical path constraint irrespective of the voltage assignment of other gates. Section III describes an algorithm to assign this V_L to the gates in the circuits using gate slack. All algorithms are written in Perl programming language. The results are discussed in Section IV and Section V concludes the paper.

II. Slack-Based Algorithm For Finding An Optimal Lower Supply Voltage

In our work, slack of a gate is defined as the difference of the critical path delay of the circuit and the delay of the longest path through that gate [11], [13]. Thus, each gate has its own slack and the gates with same slack fall on the same path unless there are two paths with equal delays. We propose three theorems to categorize the gates in any given circuit based on the gate slacks.

A. Theorem I

Consider all gates with the high voltage supply, initially. The increase in the individual gate delay due to lowering of its supply voltage as a function of its slack in the all high voltage circuit is shown in Figure 1.

Statement: *All gates that fall above the 45^o line in the ‘Delay increment versus slack’ plot cannot be assigned lower supply voltage without violating the positive slack constraint. These gates belong to group 1.*

B. Theorem II

We define two new variables introduced in a recent work [10], [13], β *and* S_u. S_u, the upper slack time is the lower bound of slacks of the gates which can be unconditionally assigned low voltage without affecting the critical timing of the circuit. These include gates with a large slack, i.e., shorter paths which do not affect the critical path time even when their delay is increased as result of assigning all the gates in these paths a lower supply voltage.

Statement:

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

where β is the ratio of the low voltage delay and the high voltage delay of each gate and T_c is the critical path delay.

$$\beta_i = \frac{d_{li}}{d_{hi}}$$

where d_{li} is the low voltage delay and d_{hi} is the high voltage delay of gate ’i’ . The maximum value of β, β_{max}, will give us the lower bound on the gate slacks. The number of gates with slacks greater than S_u in a given circuit are represented by G and these gates are grouped under group 2.

C. Theorem III

The set of gates whose slack fall below the 45^o line in the ‘Delay increment versus slack’ plot of Figure 1 and whose slacks are also less than S_u forms group 3. The cardinality of this set is P.

Statement: *There exists a group of gates within P which can be assigned lower supply voltage simultaneously without violating the positive slack constraint satisfying the condition*

$$\sum_{i=1}^{m} y_i \leq min\{slack_{hi}\}$$

where y_i is the difference of the low voltage delay and the high voltage delay of each gate.

$$y_i = d_{li} - d_{hi}$$

where d_{li} is the low voltage delay of gate 'i' *, d_{hi} is the high voltage delay of gate* 'i' *and $slack_{hi}$ is the slack of the gate* 'i' *when it is in high voltage."*

The proofs of the above theorems are found in the literature [3]. Having described the relevant theorems, we now describe our algorithms.

D. Algorithm 1

Step 1: We use the $O(n)$ slack calculation algorithm proposed in [13], [10] to find out the gate slacks for a given circuit.

Step 2: The gates are divided into groups 1, 2 and 3 as described by Theorems I, II and III.

Step 3: Once this is done, we estimate the dynamic energy savings for the gates in group 2 and 3 together for the circuit.

The dynamic energy when all gates in groups 2 and 3 together are assigned a high supply voltage will be proportional to

$$E_{H1} = V_H^2 \times (G + P)$$

Similarly, the dynamic energy when all gates in groups 2 and 3 together are assigned a low supply voltage will be proportional to

$$E_{L1} = V_L^2 \times (G + P)$$

Then the energy savings for groups 2 and 3 together is estimated as

$$E_{save1} = \frac{E_{H1} - E_{L1}}{E_{H1}} \times 100 = \frac{V_H^2 - V_{L1}^2}{V_H^2} \times (G+P) \times 100 \quad (1)$$

Step 4: Now, repeat steps 2 and 3 for each value of V_L within the specified range of voltages

Step 5: Find the low voltage V_{L1} as the V_L when $E_{save1}$is maximum.

Empercially, the optimum lower voltage for no level converters allowed in the circuit is found to be

$$V_L = V_{L1} \quad (2)$$

Thus, we obtain the value of the lower supply voltage giving a maximum energy savings. The gate slacks are compared to the $d_l - d_h$ values for each gate and for each value of V_L available. And then the gates are divided into respective groups based on their slacks. Then the energy savings is calculated for each value of V_L available and a minimum value is found; the corresponding V_L is the required lower voltage value. Since the number of voltages available for V_L is negligible when compared to the number of gates in the circuit, this algorithm is proportional to the number of gates in the circuit. Thus, its complexity is linear, or $O(n)$ for n gates.

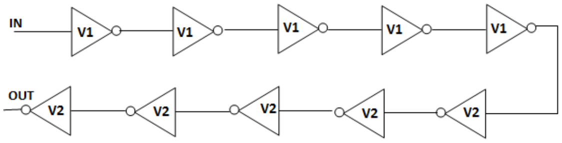
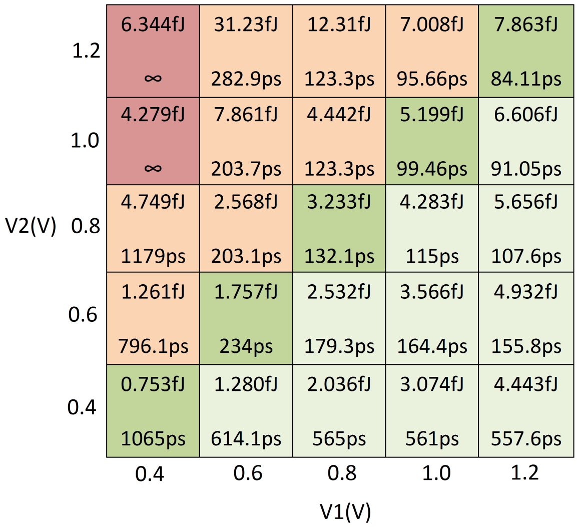


Fig. 2. A chain of ten inverters.



V2(V)					
1.2	6.344fJ ∞	31.23fJ 282.9ps	12.31fJ 123.3ps	7.008fJ 95.66ps	7.863fJ 84.11ps
1.0	4.279fJ ∞	7.861fJ 203.7ps	4.442fJ 123.3ps	5.199fJ 99.46ps	6.606fJ 91.05ps
0.8	4.749fJ 1179ps	2.568fJ 203.1ps	3.233fJ 132.1ps	4.283fJ 115ps	5.656fJ 107.6ps
0.6	1.261fJ 796.1ps	1.757fJ 234ps	2.532fJ 179.3ps	3.566fJ 164.4ps	4.932fJ 155.8ps
0.4	0.753fJ 1065ps	1.280fJ 614.1ps	2.036fJ 565ps	3.074fJ 561ps	4.443fJ 557.6ps
	0.4	0.6	0.8	1.0	1.2
			V1(V)		

Fig. 3. Energy and delay measurements at various values of V_1 and V_2 for the inverter chain of Figure 2.

III. Slack-Based Algorithm For Dual Voltage Assignment

A. An example of a chain of inverters

Consider the chain of inverters shown in Figure 2. We simulated this circuit in using Synopsys HSPICE program [2], with voltages V_1 and V_2 as 0.4V, 0.6V, 0.8V, 1.0V and 1.2V. A 1 GHz 50% duty-cycle clock is applied at the input and a capacitance of 6fF has been used as the load at the output. The results for 90nm technology are presented in Figure 3. It reports the total energy consumption and delay for the circuit at various values of V_1 and V_2.

The energy values shown in the green squares are when V_1 and V_2 are equal, corresponding to a single voltage operation. The values reported in blue squares below the $V_1 = V_2$ diagonal are the values when V_1 is greater than V_2, i.e., when a high voltage gate is feeding a low voltage gate. The squares above this diagonal represent the operation when V_2 is greater than V_1, i.e., when a low voltage gate feeds a high voltage gate. We observe that the delay measurement in top cells fails, represented as infinite delay, when the voltage difference is large. For all cases above the diagonal, although logic 1 level matched V_{DD}, 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 boundry. The use of level converters has been studied in the literature [4], [8]. However, the design of such devices is still evolving and as 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, further limiting the capability for power reduction.

Constraints on circuit topology which restrict a low voltage gate driving a high voltage gate are called topological constraints in our work. For all cases where a high voltage gate feed low voltage gate, energy savings are seen. These results demonstrate the effectiveness of using topological constraints in dual-V_{DD} designs. In the remaining part of this section we describe a new algorithm that assigns low voltage to the gates of a given circuit using topological constraints.

B. Algorithm 2

Step 1: Initially we assume that all gates are at high voltage, i.e., all the gates are connected to V_H supply voltage.

Step 2: Once we obtain the V_L value using Algorithm 1, we assign the gates in group 2 the lower supply voltage.

Step 3: Then we recalculate the slack. Theorem II mandates that no negative slack occurs during this voltage assignment.

Step 4: The gates are again divided into new groups 1, 2 and 3.

Step 5: The circuit is levelized and starting from the primary outputs, we take a small group of high-voltage gates out of group 3 satisfying the condition stated in Theorem 3 and assign them low voltage.

Step 6: Recalculate slacks.

Step 7: Once we assign low voltage to this group of gates, we check whether there are any low voltage gates driving high voltage gates anywhere in the circuit.

Step 8: If this occurs, the supply of that low voltage gate is changed back to the high voltage.

Step 9: Gate slacks are calculated again.

Step 10: The gates are redivided into groups 1, 2 and 3.

Step 11: Steps 5 to 10 repeated on the remaining gates in a reverse levelized manner until we reach the primary inputs.

Thus, we have described an algorithm to assign low voltage to the gates which puts restrictions on the circuit structure.

IV. Results

We run our algorithm on ISCAS'85 benchmark circuits. These benchmark circuits are synthesized using a small set of 90nm standard cells consisting of an inverter, INV, a two-input NAND gate, NAND2, a three-input NAND gate, NAND3, and a two-input NOR gate NOR2 using bulk-CMOS models [1]. Each circuit is simulated with a logic simulator with randomly generated input vectors to determine its average activity. The gate delays, average activity of each node and node capacitances of each circuit are obtained from SPICE simulations done for supply voltages ranging from 0.4V to 1.2V in 0.01V steps. These values are also tabulated for each standard cell with fanout of one to four [3]. For all SPICE simulations, 90nm bulk CMOS predictive technology models with 0.3V threshold voltage are used at room temperature and the higher supply voltage used is V_H = 1.2V. One hundred random input vectors are used for simulations and energy per vector is found.

Algorithms 1 and 2 are used on each of the ISCAS'85 benchmark circuits to obtain the lower supply voltage, V_L, the number of gates which can be assigned low voltage using our algorithms, the final energy savings and the CPU time to run the algorithms. E_{save} for the ISCAS'85 benchmark circuits, using V_L calculated from Algorithm 1, is reported in Table I. Also, we compare these with the results obtained when $V_L = 0.7 \times V_H$ and when $V_L = 0.5 \times V_H$ following the previous work. It is observed that the expected energy saving is large when we use V_L given by Algorithm 1. As great as 20% more savings are seen for circuit c2670 when compared to the savings obtained by using $0.7 \times V_H$.

In Table II, $E_{single\ VDD}$ and $E_{dualVDD}$ are the SPICE results for energy per vector consumed in single-voltage design and that in dual-voltage design respectively. Also, $E_{savg.-expc.}$ is the energy savings estimated using the lower supply voltage value given by Algorithm 1. The actual energy savings reported by SPICE for dual-voltage design is $E_{savg.-obs.}$ and it is observed to be very close to the estimated values. Also, we compare these results with the results

TABLE I

OPTIMAL LOWER SUPPLY VOLTAGE VALUES (V_L) AND ENERGY SAVINGS ESTIMATE USING ALGORITHMS 2 WHEN LEVEL CONVERTERS ARE NOT ALLOWED FOR ISCAS'85 BENCHMARK CIRCUITS. $V_H = 1.2V$.

Benchmark circuit	Total gates	Algorithm 1			$V_L = 0.84$V		$V_L = 0.6$V	
		V_L volt	Gates in low voltage	$E_{savg.}$ %	Gates in low voltage	$E_{savg.}$ %	Gates in low voltage	$E_{savg.}$ %
c432	154	0.8	8	2.9	8	2.65	8	3.9
c499	493	0.76	113	13.73	121	12.52	56	8.52
c880	360	0.49	213	49.3	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	28.4
c2670	901	0.48	570	53.14	570	32.27	570	47.45
c3540	1270	0.52	149	9.53	149	5.98	149	8.8
c5315	2077	0.7	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

TABLE II

OPTIMAL LOWER SUPPLY VOLTAGE VALUES (V_L) AND ENERGY SAVINGS USING ALGORITHMS 1 AND 2 FOR ISCAS'85 BENCHMARK CIRCUITS, $V_H = 1.2V$.

Benchmark circuit	Total gates	Algorithms 1 & 2				SPICE Results			Reference [11]	
		V_L volt	Gates in low voltage	$E_{savg.-expc.}$ %	CPU s*	$E_{single\ VDD}$ fJ	$E_{dualVDD}$ fJ	$E_{savg.-obs.}$ %	$E_{savg.-obs.}$ %	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	427	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

* Intel Core i5 2.30GHz, 4GB RAM

reported by authors in [11]. The observed savings using our algorithms are as much as 14% more than those reported in [11] for circuit c2670 and the CPU time is 20 times less. In general, our algorithms give higher energy savings and execute in lower CPU time.

Figures 1 and 4 show the delay increment versus slack graphs for the initial slacks and final slacks, respectively, for the c880 circuit. The brown markers indicate gates in high voltage and blue markers indicate gates in low voltage. It is seen that initially all gates are in high voltage. After Algorithms 1 and 2 are used and slacks are recalculated, all gate slacks are reduced. High voltage gates tend to concentrate at lower slack values and many have also moved above the 45^o line.

The low voltage gates that are still below the 45^o line are gates with very large slacks and the high voltage gates still lying below the line indicate the gates that cannot be put in low voltage due to topological constraints imposed by Algorithm 2. Also, we can see a few brown dots to the right of S_u line. These are again the gates which cannot be put in low voltage due to topological constraints, but have slacks greater than S_u, as seen in previous section.

V. CONCLUSION

An algorithm that finds a lower supply voltage maximizing the energy savings from a specific group of gates in a given circuit is proposed. This algorithm is proportional to the number of gates in the circuit as each gate slack is compared to the difference between the low voltage delay and high voltage delay of the gate. Thus, its complexity is linear, i.e.; $O(n)$, where n is the number of the gates. Another algorithm is proposed which assigns this lower supply voltage found using the gate slacks. The gate slacks are recalculated after each iteration of voltage assignment. Both the

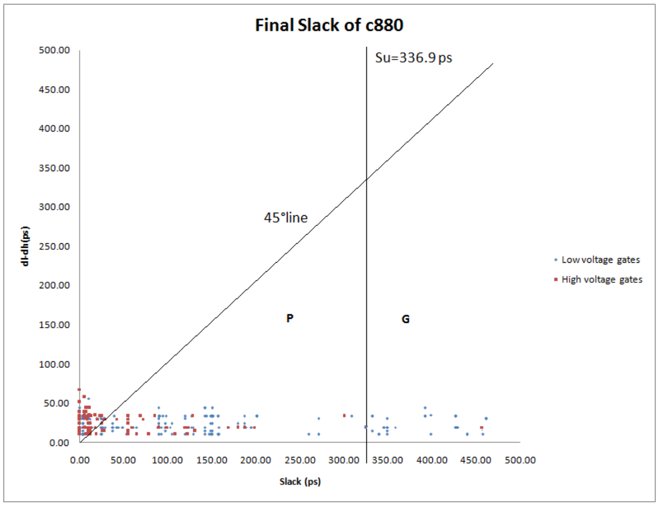


Fig. 4. Delay increment versus final slack for gates of c880 circuit, $V_H = 1.2V,\ V_L = 0.49V$.

voltage assignment and the slack calculating algorithms are linear in time. Thus, the overall complexity of this method is quadratic, i.e., $O(n^2)$, where n is the number of gates. But in practice, it is found to use less time than that taken by other algorithms proposed in the literatute [11]. Energy savings of up to 60% are seen for ISCAS'85 benchmark circuits. Such high savings have not been reported in the earlier work. Sufficient theoretical and experimental work has been done to validate these results.

References

[1] "Latest PTM Models." http://ptm.asu.edu/ (accessed on Dcember 11, 2011).

[2] *HSPICE®Reference Manual: Commands and Control Options, Version D-2010.03-SP1*, June 2010. http://www.synopsys.com/Tools/Verification/AMS Verification/ 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, Auburn University, ECE Dept., Auburn, AL, Dec. 2011.

[4] 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*, vol. 10, no. 1, pp. 55–60, Mar. 2010.

[5] S. Augsburger and B. Nikolic, "Combining Dual-Supply, Dual-Threshold and Transistor Sizing for Power Reduction," in *Proceedings of the IEEE International Conference on Computer Design*, 2002, pp. 316–321.

[6] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-Power CMOS Digital Design," *IEEE Journal of Solid-State Circuits*, vol. 27, no. 4, pp. 473–484, Apr. 1992.

[7] C. Chen, A. Srivastava, and M. Sarrafzadeh, "On Gate Level Power Optimization Using Dual-Supply Voltages," *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 5, pp. 616–629, Oct. 2001.

[8] D. G. Chinnery and K. Keutzer, *Closing the Power Gap Between ASIC and Custom - Tools and Techniques for Low Power Design*. Springer, 2007.

[9] M. Hamada, Y. Ootaguro, and T. Kuroda, "Utilizing Surplus Timing for Power Reduction," in *Proceedings of the IEEE Custom Integrated Circuits Conference*, 2001, pp. 89–92.

[10] K. Kim, *Ultra Low Power CMOS Design*. PhD thesis, Auburn University, ECE Dept., Auburn, AL, May 2011.

[11] K. Kim and V. D. Agrawal, "Dual Voltage Design for Minimum Energy Using Gate Slack," in *Proceedings of the IEEE International Conference on Industrial Technology*, Mar. 2011, pp. 419–424.

[12] K. Kim and V. D. Agrawal, "Minimum Energy CMOS Design With Dual Subthreshold Supply and Multiple Logic-Level Gates," in *Proceedings of the 12th International Symposium on Quality Electronic Design*, Mar. 2011, pp. 689–694.

[13] K. Kim and V. D. Agrawal, "Ultra Low Energy CMOS Logic Using Below Threshold Dual Voltage Supply," *Journal of Low Power Electronics*, vol. 7, no. 4, pp. 460–470, Dec. 2011.

[14] S. H. Kulkarni, A. N. Srivastava, and D. Sylvester, "A New Algorithm for Improved V_{DD} Assignment in Low Power Dual V_{DD} Systems," in *Proceedings of the International Symposium on Low Power Design*, 2004, pp. 200–205.

[15] S. H. Kulkarni and D. Sylvester, "High Performance Level Conversion for Dual V_{DD} Design," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 9, pp. 926–936, Sept. 2004.

[16] T. Kuroda and M. Hamada, "Low-Power CMOS Digital Design with Dual Embedded Adaptive Power Supplies," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 4, pp. 652–655, Apr. 2000.

[17] M. Pedram and J. M. Rabaey, *Power Aware Design Methodologies*. Boston: Springer, 2002.

[18] 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*, pp. 10718–10719, 2004.

[19] K. Usami and M. Horowitz, "Clustered Voltage Scaling Technique for Low-Power Design," in *Proceedings of the International Symposium on Low Power Design*, 1995, pp. 23–26.

[20] 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*, vol. 33, no. 3, pp. 463–472, Mar. 1998.

[21] Y.-J. Yeh and S.-Y. Kuo, "An Optimization-Based Low-Power Voltage Scaling Technique Using Multiple Supply Voltages," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, volume 5, 2001, pp. 535–538.