Subthreshold Logical Effort: A Systematic Framework for Optimal Subthreshold Device Sizing

John Keane   Hanyong Eom   Tae-Hyoung Kim   Sachin Sapatnekar   Chris Kim

Department of Electrical Engineering

University of Minnesota, Minneapolis, MN

{jkeane, eomxx001, kimxx692, sachin, chriskim}@umn.edu

ABSTRACT

Subthreshold circuit designs have been demonstrated to be a successful alternative when ultra-low power consumption is paramount. However, the characteristics of MOS transistors in the subthreshold regime are significantly different from those in strong-inversion. This presents new challenges in design optimization, particularly in complex gates with stacks of transistors. In this paper, we demonstrate a new optimal sizing scheme for subthreshold designs which takes these issues into account. We derive a closed-form solution for the correct sizing of transistors in a stack, both in relation to other transistors in the stack, and to a single transistor with equivalent current drivability. Experimental results show that our framework provides a performance improvement of up to 13.5% over the conventional logical effort method on ISCAS benchmark circuits, while one component circuit demonstrated an improvement of 33.1%.

Categories and Subject Descriptors

B.7.2 [**Hardware**]: Integrated Circuits—Types and Design Styles.

General Terms

Algorithms, Performance, Design

Keywords

Subthreshold logic, logical effort, ultra low power design

1. INTRODUCTION

Due to the robust nature of static CMOS logic, circuits in this technology family can operate with supply voltages below the transistor threshold voltage (V_{th}), while consuming orders of magnitude less power than in the normal strong-inversion region. The operating frequency of subthreshold logic is much lower than that of regular strong-inversion circuits ($V_{dd} > V_{th}$) due to the small transistor current, which consists entirely of leakage current. The low operating frequency and low supply voltage combine to reduce both dynamic and leakage power, leading to the significant power savings seen in subthreshold designs.

Subthreshold logic holds promise for the growing number of applications in which minimal power consumption is the primary design constraint. Such circuits have received much attention in recent research, and a number of successful designs have been demonstrated. A multiplexer-based SRAM was proposed for

subthreshold operation by the authors of [1] at ISSCC 2004. They also introduced new tiny-XOR circuits and demonstrated their performance in a Fast Fourier Transform processor running at a supply voltage of 180mV. Dynamic voltage scaling down to the subthreshold region was demonstrated by Calhoun et al. [2]. Kim et al. showed device-level optimization of subthreshold double-gate transistors, revealing how the scaling trend of transistors for subthreshold operation should be different from those for normal strong-inversion operation [3]. In [4] Kim et al. built an ultra-low power adaptive filter using subthreshold logic for hearing aid applications. Subthreshold-friendly logic styles and massively parallel DSP architectures were used in that work to achieve ultra-low voltage operation

The characteristics of MOS transistors in the subthreshold region are significantly different from those in the strong-inversion region. The MOS saturation current, which was a near-linear function of the gate and threshold voltages in the strong-inversion region, becomes an exponential function of those values in the subthreshold regime [5]. In this work, we show that the sizing methods used to obtain maximum performance must be reformulated for use in subthreshold designs due to these different characteristics. In particular, we explain how the widely-used logical effort method must be modified, and we develop a new framework for optimal device sizing in subthreshold based on this method. A closed-form solution for the optimal sizing of stacked transistors is derived and shown to match experimental results. Finally, we present HSPICE simulation results from ISCAS benchmarks and component circuits demonstrating the advantage of our approach versus the conventional logical effort method. Improvements in performance of up to 33.1% are reported and justified with simple calculations based on our framework.

2. CONVENTIONAL LOGICAL EFFORT

The logical effort method was presented by Sutherland et al. as a simple way to both estimate and optimize the delay of CMOS circuits [6]. The gate delay (d) is modeled as $d = ghb + p$, where g is the logical effort, h is the electrical effort, b is a branching factor which accounts for off-path capacitance, and p is the parasitic delay. Logical effort is defined as the ratio of the input capacitance of a gate to that of an inverter delivering the same amount of output current. The electrical effort represents the ratio of output capacitance to input capacitance, the ghb product is called the stage effort, and the parasitic delay is defined as the delay of a gate driving no load. This final value is set by the parasitic junction capacitance.

In conventional logical effort calculations, the optimal ratio of PMOS width (W_P) to NMOS width (W_N) for achieving equivalent current drivability is approximately 2.5:1, due to the mobility difference between charge carriers in PMOS and NMOS

devices. In addition, the effective width of a transistor in a stack of n devices is roughly $1/n$ in the strong-inversion region. This means that in order for an n-stack to conduct the same amount of current as a single transistor, the devices in the stack must each be sized up by a factor of n. Selection of the proper $W_P{:}W_N$ ratio and effective width of stacked transistors is crucial for achieving optimal performance. We have found that the conventional logical effort framework based on strong-inversion operation fails to do so for subthreshold logic due to the difference in the transistor current behavior. In the strong-inversion regime, current is a first or second-order function of the four MOS terminal voltages. As stated in section 1, the drive-current in subthreshold designs is an exponential function of the terminal voltages. Hence we need a new design paradigm for optimal device sizing based on the exponential current equation in the subthreshold region.

3. SUBTHRESHOLD LOGICAL EFFORT

3.1 Optimal Stack Sizing

The first step we take in developing the subthreshold logical effort framework is finding the optimal width ratio between transistors in a stack for maximum drive-current. We present a closed-form expression for the relative sizing of two transistors in a stack, showing that it is beneficial to size up the transistor nearest to the supply rail (V_{dd} for PMOS, ground for NMOS). The starting point is the following pair of current equations for upper and lower transistors in the stack (as situated in an NMOS stack, so the lower device is connected to ground):

$$I_U = W_U e^{\frac{(V_{dd}-V_X)-(V_{t0}+\gamma V_X+\lambda_d(V_{dd}-V_X))}{mV_T}} \left(1 - e^{\frac{-(V_{dd}-V_X)}{V_T}}\right) \quad (1)$$

$$\approx W_U e^{\frac{(V_{dd}-V_X)-(V_{t0}+\gamma V_X+\lambda_d(V_{dd}-V_X))}{mV_T}}$$

$$I_L = W_L e^{\frac{V_{dd}-(V_{t0}+\lambda_d V_X)}{mV_T}} \left(1 - e^{\frac{-V_X}{V_T}}\right) \quad (2)$$

Here, W_U and W_L denote the upper and lower transistor widths, respectively, and V_X denotes the voltage at the node between those devices. The Drain-Induced Barrier Lowering (DIBL) coefficient (a negative number) is represented by λ_d, and γ is the body effect coefficient. The thermal voltage is represented by V_T, while V_{t0} stands for the nominal threshold voltage. According to simulation results, we can approximate $V_X \approx 0$V, and therefore $V_{dd}-V_X \approx V_{dd}$. Moreover, it may be noted that $e^{-(V_{dd}-V_X)/V_T} \approx 0$. We use the symbol

$$\alpha = e^{\frac{-\lambda_d V_{dd}}{mV_T}}, \quad (3)$$

as well as the fact that $m = 1+\gamma$, to further simplify calculations. Rewriting the two current equations and equating them yields the following relationship:

$$\alpha W_U e^{\frac{-V_X}{V_T}} = W_L \left(1 - e^{\frac{-V_X}{V_T}}\right) \quad (4)$$

Solving for V_X and using the definition $V_T = kT/q$ gives us

$$V_X = \frac{kT}{q} \ln\left(1 + \frac{\alpha W_U}{W_L}\right) \quad (5)$$

We then define $W_T = W_U + W_L$ to eliminate W_L, which results in the following current equation:

$$I_U = I_L = \frac{\alpha W_U (W_T - W_U)}{\alpha W_U + W_T - W_U} e^{\frac{V_{dd}-V_{t0}}{mV_T}} \quad (6)$$

We find the optimal size for W_U by setting $(\partial I_U / \partial W_U)$ equal to zero. Again using our definition of W_T, we then find the optimal size for W_L. This derivation shows that

$$W_U = \frac{W_T}{1+\sqrt{\alpha}} \quad (7)$$

$$W_L = \frac{W_T}{1+\sqrt{\alpha}} \sqrt{\alpha} \quad (8)$$

According to these results, we expect to achieve a higher drive-current through the two-transistor stack when the lower device is larger than the upper transistor by a factor of $\sqrt{\alpha}$. For example, with a W_U of 1μm the optimal W_L is 1.189μm at $V_{dd} = 0.3$V, and 1.122μm at $V_{dd} = 0.2$V. As shown in equation (3), α is a function of V_{dd} (see Table 1 for $1+\alpha$ values), resulting in the different optimal width ratios for different V_{dd} values.

HSPICE simulations using 0.13μ technology verify that the result of our derivation is correct, and that the benefit is more pronounced for larger α values (that is, when the supply voltage is at the higher end of the subthreshold range). PMOS transistor stacks exhibited the same sizing trends—optimal sizing requires the upper transistor (adjacent to the power supply) to be sized up by a factor of $\sqrt{\alpha}$. The results are displayed in Figure 1. Due to the small difference in current with the skewed sizing (~1% improvement, which is close to the theoretical improvement), we will use a 1:1 width ratio in stacks. This reduces the design complexity for a negligibly small performance penalty.
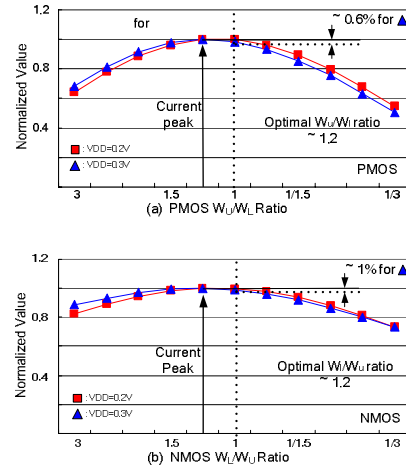

Figure 1: Current measured in DC for a range of $W_U{:}W_L$ sizing ratios.

After deciding to use a 1:1 ratio for the two devices in a stack, we must find the amount by which they should be sized up to drive the same current as a single transistor. Defining $W = W_U = W_L$ as the size of each transistor in the stack, we can modify equation (6) as follows:

$$I_U = I_L = \frac{\alpha W^2}{\alpha W + W} e^{\frac{V_{dd}-V_{t0}}{mV_T}} = \frac{\alpha}{1+\alpha} W e^{\frac{V_{dd}-V_{t0}}{mV_T}} \quad (9)$$

For a single transistor, the current equation is:

$$I = W_{eff} e^{\frac{V_{dd}-(V_{t0}+\lambda_d V_{dd})}{mV_T}} = \alpha W_{eff} e^{\frac{V_{dd}-V_{t0}}{mV_T}}, \quad (10)$$

where W_{eff} stands for the effective width of this device. From equations (9) and (10), we have the following relationship:

$$\alpha W_{eff} = \frac{\alpha}{1+\alpha}W \rightarrow W_{eff} = \frac{1}{1+\alpha}W \qquad (11)$$

According to this equation, two stacked transistors should be sized up by a factor of $1+\alpha$ in relation to a single transistor for the same current drivability. Table 1 lists $(1+\alpha)$ values for a number of different V_{dd} values. Our derivation indicates that stacks need to be sized up by a larger amount in the subthreshold region compared to the superthreshold region. For example, a single unit transistor is equivalent to a two-stack with transistor widths of 2.259 at 0.2V, 2.413 at 0.3V, and 1.6 at 1.2V. A larger transistor is needed in the stack with a 0.3V supply compared to a supply of 0.2V due to the larger α value. Note that stacked NMOS transistors are only sized up by a factor of 1.6 at 1.2V (rather than a factor of 2) due to velocity saturation.

Table 1: $1+\alpha$ values for stack sizing

V_{dd}	PMOS/NMOS	$1+\alpha$
0.2V	PMOS	2.428
	NMOS	2.259
0.3V	PMOS	2.707
	NMOS	2.413
1.2V	PMOS	2.1[*]
	NMOS	1.6[*]

(*Superthreshold values are not calculated with equation (3)—they are derived from DC simulation and fit the $1+\alpha$ sizing factor)

3.2 Optimal W_P:W_N Ratio

The optimal PMOS to NMOS width ratio in the subthreshold regime was found by simulating a chain of equally sized inverters and observing the rise and fall delays. Results show that a 1.5:1 ratio gives equal delays for the rise and fall transitions at $V_{dd} = 0.2$V, and a slightly smaller ratio is optimal for $V_{dd} = 0.3$V. The 1.5:1 ratio will be used in all subthreshold simulations to maintain consistency.

3.3 New Logical Effort Formulation

Based on the results from the previous sections, we can now summarize our new logical effort values for different types of gates operating in the subthreshold region. Figure 2 compares the logical efforts of standard logic gates in strong-inversion operation with those in the subthreshold region.
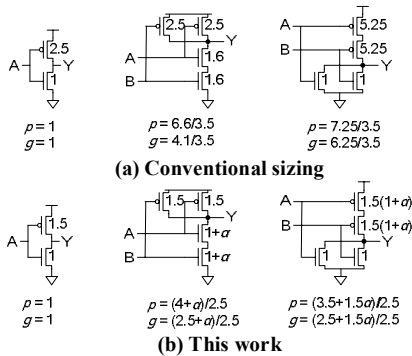

Figure 2: Parasitic delay (p) and logical effort (g) values

To verify the stack sizing factors based on our derivation, we ran DC simulations to compare the current through a single transistor to the current through a stack at different supply voltage levels. Each device in the stack was sized equivalently to the single transistor. The ratio of the currents indicates by how much the stack transistors must be sized up to achieve the same level of drive-current observed in the single device. Table 2 compares the simulation results with the stack scaling factor of $1+\alpha$ derived in section 3.1. The results of our derivation closely match the simulation results.

**Table 2: Measured and theoretical sizing factors for 2-stacks**

	$V_{dd} = 0.2$V		$V_{dd} = 0.3$V	
	Measured	Theoretical $1+\alpha$	Measured	Theoretical $1+\alpha$
PMOS	2.4	2.428	2.64	2.707
NMOS	2.25	2.259	2.44	2.413

3.4 Library Design: Arbitrary Stack Sizes

Building an extensive cell library based on our new logical effort framework requires us to extend our work to stacks of three or more devices. The derivation for the current equation of a three-stack, which follows a similar method as the derivation in section 3.1 gives us the following result:

$$I = \alpha \left[\frac{(W_T - W_1 - W_2)W_1 W_2}{\alpha(W_T - W_1 - W_2)(W_2 + W_1) + W_1 W_2} \right] e^{\frac{V_{dd} - V_{t0}}{mV_T}} \qquad (12)$$

W_1 and W_2 stand for the widths of the two lower transistors in the stack of NMOS devices (see notation in Figure 3). W_T is defined as $W_T = W_1 + W_2 + W_3$, and is used to eliminate W_3, the width of the upper device. This equation is symmetric with respect to the widths of W_1 and W_2 transistors, indicating that the optimal sizes for the lower two devices in the stack are equal. A straightforward direct proof confirms the symmetry of the lower n-1 transistor widths in an n-stack achieving maximum drive current.
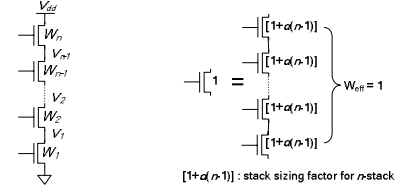
(a) n-stack notation   (b) n-stack sizing for equivalent width
Figure 3: NMOS n-stack

We have also proven that the optimal ratio between the n-1 lower devices and the upper device is $\sqrt{\alpha}$, which is equivalent to the two-stack case (equations (7) and (8)). As in the two-transistor stack, the scaling factor of $\sqrt{\alpha}$ leads to a trivial performance benefit, so sizing all stacked transistors equally is the best choice in terms of overall design complexity. Theory and simulation have both show that each device in an n-stack should be scaled up by a factor of $[1+\alpha(n-1)]$ to set the effective width of the stack equal to that of a single unit transistor. Note that all work done here again applies to PMOS stacks in a similar manner.

4. EXPERIMENTAL RESULTS

4.1 ISCAS Benchmark Results

We tested our sizing framework by synthesizing a number of ISCAS benchmark circuits, as well as component circuits used in that suite. Three cell libraries were created, each containing an inverter, a two-input NAND, and a two-input NOR. The cells in the first library were optimized for a supply of 1.2V with a 2.5:1 W_P:W_N ratio. The other two libraries contained cells optimized for supplies of 0.2V and 0.3V, which use a 1.5:1 W_P:W_N ratio.

Critical path delays through circuits using conventional superthreshold logical effort sizing and optimized subthreshold sizing are compared for 0.2V and 0.3V supplies in Table 4.

As these results demonstrate, our sizing framework consistently provides a performance benefit in subthreshold circuits. Improvements range from 4.38% to 33.1% in different cases because performance is highly dependent on circuit topology. This range of speedup values can be explained by examining simple cases with the logical effort model.

For instance, we will analyze the delay through a single NAND gate followed by a NOR, within a longer NAND-NOR chain, operating at 0.3V. The logical effort values for conventionally sized and optimized gates at this supply level are presented in Figure 4. Notice that the former set of gates have separate logical efforts for the pull-up (g_u) and pull-down (g_d) paths, because the reference gate is now the inverter seen in Figure 4(b)—that is, the inverter optimized for operation at 0.3V.

(a) Conventional: logical effort of pull-up and pull-down paths
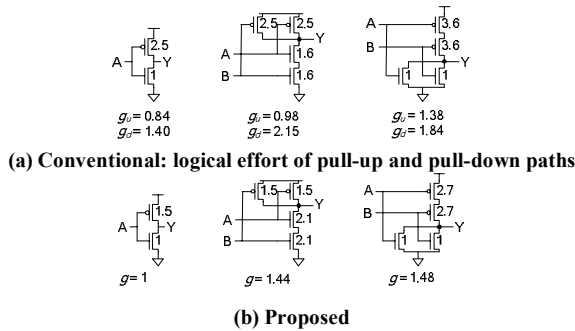

(b) Proposed

Figure 4: Logical effort values with a supply of 0.3V.

As an example, the logical efforts for the NAND gate in Figure 4(a) are computed as follows:

$$g_u = \frac{2.5+1.6}{2.5(2.5/1.5)} = 0.98; \qquad g_d = \frac{2.5+1.6}{2.5(1.6/2.1)} = 2.15 \qquad (13)$$

where the ratio in each denominator accounts for the difference between the conventional and optimal path sizes. The nominal delay through one NAND-NOR pair is computed with the following equation from logical effort theory:

$$delay = (g \cdot h \cdot b)_{NAND} + (g \cdot h \cdot b)_{NOR} + p_{total} \qquad (14)$$

where p_{total} represents the total parasitic junction capacitance in the two gates. The *delay* values for two different cases are displayed in Table 3. In both examples, the critical path travels through the stack of the NAND gate; however, in the first case, both branching factors are equal to one, whereas in the second case, the branching factor of the NAND gate is four. These simple calculations show that the 21% improvement seen in section 4.1, with no branching, and the performance gains of ~30% observed in the ISCAS benchmarks match theoretically attainable improvements. Smaller benefits are obtained with different combinations of logical effort values and branching factors.

Table 3: NAND-NOR delays at 0.3V computed with equation (14)

	Conventional	New	Improvement
No branching	8.52	6.84	20%
NAND b=4	15.74	11.29	28%

5. CONCLUSION

We have presented a new logical effort optimization framework for circuits operating in the subthreshold region. A closed-form solution for the optimal ratio of different devices within a stack, as well as the sizing factor for stacked devices, was presented and shown to closely match experimental results. Our optimization scheme resulted in performance gains of up to 13.5% for ISCAS benchmark circuits and 33.1% for component circuits operating in subthreshold, which was shown to match theoretically attainable improvements.

6. ACKNOWLEDGEMENTS

7. REFERENCES

[1] A. Wang, A.P. Chandrakasan, "A 180-mV subthreshold FFT processor using a minimum energy design methodology", IEEE JSSC, Volume 40, Issue 1, pp. 310-319, Jan. 2005.

[2] B. Calhoun, A. Chandrakasan, "Ultra-dynamic voltage scaling using sub-threshold operation and local voltage dithering in 90nm CMOS", ISSCC, pp. 300-301, 2005.

[3] J.J. Kim, K. Roy, "Double gate-MOSFET subthreshold circuit for ultra-low power applications", IEEE Transactions on Electron Devices, Volume 51, Issue 9, pp. 1468-1474, Sept. 2004.

[4] C.H. Kim, H. Soeleman, K. Roy, "Ultra-low-power DLMS adaptive filter for hearing aid applications", IEEE Transactions on VLSI Systems, Volume 11, Issue 6, pp. 1058-1067, Dec. 2003.

[5] E. Vittoz, J. Fellrath, "CMOS analog integrated circuits based on weak inversion operations", IEEE JSSC, Vol. 12, Issue 3, pp. 224-231, June 1977.

[6] I. Sutherland, B. Sproull, and D. Harris, Logical Effort: Designing Fast CMOS Circuits. San Francisco, CA: Morgan Kaufmann, Jan. 1999.

Table 4: Results from ISCAS benchmarks and component circuits ("CX": benchmarks; "74X": components)

Circuit	0.3V			0.2V		
	conventional	proposed	improvement	conventional	proposed	improvement
C432	12.93 ns	11.55 ns	10.67%	99.44 ns	89.38 ns	10.11%
C6288	24.71 ns	21.59 ns	12.63%	186.0 ns	170.6 ns	8.31%
C3540	35.06 ns	33.53 ns	4.38%	270.6 ns	253.6 ns	6.29%
C1355	12.40 ns	10.73 ns	13.46%	103.1 ns	90.41 ns	12.32%
74283	43.74 ns	41.45 ns	5.25%	340.7 ns	323.4 ns	5.08%
74181	47.70 ns	44.74 ns	6.20%	378.8 ns	353.1 ns	6.78%
74L85	22.88 ns	21.37 ns	6.59%	185.2 ns	170.7 ns	7.80%
74182	29.18 ns	19.52 ns	33.1%	215.3 ns	146.2 ns	32.1%