Design Considerations and Tools for Low-voltage Digital System Design

Anantha Chandrakasan, Isabel Yang, Carlin Vieri, Dimitri Antoniadis
Department of EECS,
Massachusetts Institute of Technology, Cambridge

ABSTRACT

Aggressive voltage scaling to 1V and below through technology, circuit, and architecture optimization has been proven to be the key to ultra low-power design. The key technology trends for low-voltage operation are presented including low-threshold devices, multiple-threshold devices, and SOI and bulk-CMOS based variable threshold devices. The requirements on CAD tools that allow designers to choose and optimize various technology, circuit, and system parameters are also discussed.

1. Introduction

The weight and size of batteries, which are major factors in portable systems, are primarily determined by the power dissipation of electronic circuits. The strict limitation on power dissipation which portability imposes, must be met by the designer while also meeting ever higher computational requirements. Over the last few years, significant advances have been made in developing methodologies for low-power design and two global system strategies have emerged which involve the reduction of switched capacitance and supply voltage scaling [1].

The switched capacitance can be reduced at various levels of the design including technology development (e.g., using Silicon-on-Insulator over bulk CMOS to lower node capacitances), circuit and logic design (e.g., data dependent shut down [2] or glitch elimination techniques), architecture design (e.g., optimizing resource sharing considering signal statistics), and algorithm selection (e.g., reducing operation count or optimizing data representation). To help designers perform trade-offs at various levels of the design abstraction, a variety of CAD tools have been developed to estimate switching activity.

The other key strategy for low-power design is voltage scaling. While operating supply voltages have dropped to 3V, it has been shown that aggressive voltage scaling down to 1V or below is the key technique to low-power design. An architectural voltage scaling strategy which trade-offs silicon area for lower power consumption has been proposed [1]. Another key strategy for sub-1V operation is to lower the threshold voltage of devices, which unfortunately comes at the cost of increased leakage power. CAD tools that support low-voltage, low-power design should explicitly take leakage into account.

The optimum selection of technology, circuit, and system parameters (e.g., threshold voltages or the operating power supply voltage) depends on the application being implemented, node and module switching activities, module access patterns, etc.

2. Analysis of Power Components

There are three main sources of power consumption in digital CMOS circuits: switching or dynamic power, short circuit power, and leakage power. The switching component is the energy dissipated during the charging or discharging of parasitic capacitors. In conventional process technology using "proper" circuit design, the switching component dominates and is given by:

$$P_{switching} = \alpha_{0\text{->}1}\, C_L V_{DD}^2 f_{clk} \quad (1)$$

where C_L is load capacitance, V_{DD} is the supply voltage, f_{clk} is the clock frequency, and $\alpha_{0\text{->}1}$ is the node transition activity factor or the fraction of the time the node makes a power consuming transition inside the clock period. The activity factor, $\alpha_{0\text{->}1}$, is a strong function of signal statistics. A variety of techniques have been proposed to exploit signal statistics to reduce power consumption at the logic, circuit, and architecture levels [1].

The load capacitance is non-linear (consisting of gate, junction, and interconnect components) and is a function of supply voltage. Figure 1 shows the switched capacitance as a function of operating power supply voltage for three different registers [3]. The increase in capacitance is attributed to the increase in effective gate capacitance with voltage. This figure indicates that it is

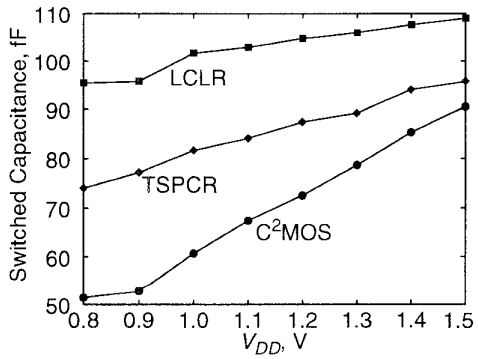

Figure 1: Non-linear dependence of C_L on V_{DD}.

33rd Design Automation Conference®

DAC 96 - 06/96 Las Vegas, NV, USA
©1996 ACM 0-89791-779-0/96/0006..$3.50

necessary to take capacitive non-linearities into account for accurate estimation of power consumption.

The short-circuit component arises when both the NMOS and PMOS transistors are "ON" simultaneously, providing a direct path from V_{DD} to ground. By sizing transistors such that the input and output rise-times are approximately equal, the short circuit component can be kept to less than 10% of the total power.

The third component of power is the leakage power, resulting from reverse biased diode conduction and subthreshold operation. The sub-threshold leakage occurs due to carrier diffusion between the source and the drain when the gate-source voltage, V_{gs}, has exceeded the weak inversion point, but is still below the threshold voltage V_T, where carrier drift is dominant. In this regime, the MOSFET behaves similarly to a bipolar transistor, and the subthreshold current is exponentially dependent on the gate-source voltage V_{gs}. Figure 2 shows the I_D vs. V_{gs} plot (log scale) for two different threshold voltages for an NMOS device in a Silicon-on-Insulator (SOI) process.

The current in the subthreshold region is well known and is given by:

$$I_{ds} = Ke^{\left(V_{gs}-V_T\right)/\left(nV_t\right)}\left(1-e^{-\frac{V_{ds}}{V_t}}\right) \qquad (2)$$

where K is a function of the technology, V_t is the thermal voltage (KT/q) and V_T is the threshold voltage and $n = 1 + \Omega\, t_{ox}/D$, where t_{ox} is the gate oxide thickness, D is the channel depletion width, and $\Omega = \varepsilon_{si}/\varepsilon_{ox}$. For $V_{ds} >> V_t$, $(1\text{-}e^{-Vds/Vt}) \approx 1$; that is, the drain to source leakage current is independent of the drain-source voltage V_{ds}, for V_{ds} approximately larger than 0.1V. Associated with this is the subthreshold slope S_{th} (=2.3 nV_t), which is the amount of voltage required to drop the subthreshold current by one decade. At room temperature, typical values for S_{th} lie between 60 to 90 mV/(decade current), with 60 mV/dec being the lower limit. Clearly, the lower S_{th} is, the better, since it is desirable to have the device "turn-off" as close to V_T as possible. When the threshold voltage is high, the subthreshold leakage is typically small. However, at low threshold voltages, the leakage current can be significant.

This can be see from Figure 2 which indicates that the leakage current for an NMOS device (i.e., with V_{gs} = 0V) is less than 10pA for V_T = 0.4V while it is 0.1μA for V_T=0.1V. Current power estimation tools (except at the SPICE level) do not take the subthreshold leakage component into account.

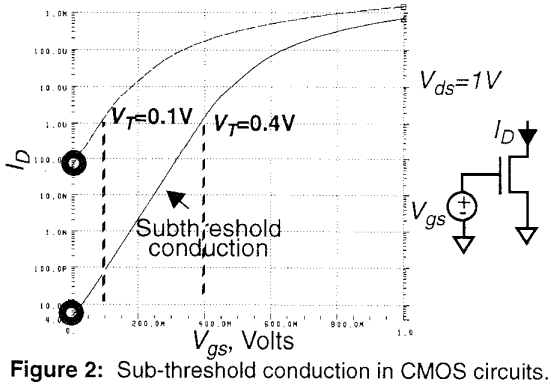

Figure 2: Sub-threshold conduction in CMOS circuits.

3. Threshold Voltage and Supply Selection for "Continuously Operational" Circuits

One important class of applications are those which have no advantage in exceeding a bounded computation rate, as found in real-time signal processing. To lower energy consumption in CMOS circuits, it is desirable to operate at the lowest possible power supply voltage. However, the individual circuit elements run slower at lower supply voltages and hence result in reduced performance. Reducing the threshold voltage (V_T) of the device allows the supply voltage to be scaled down (and therefore lower switching power) without loss in performance. Figure 3 shows an experimentally obtained plot of V_{DD} vs. V_T while keeping performance (gate delay) constant. These experimental plots are obtained from ring oscillator structures by adjusting the V_T (using the variable V_T SOI device described later) and V_{DD} for a fixed delay.

Since significant power improvements can be gained through the use of low-threshold MOS devices, the question of how low the thresholds can be reduced must be addressed. Figure 4 shows a plot of energy vs. threshold voltage for two different speeds of a 101 stage ring oscillator (e.g., all points on each curve have a fixed delay). Here, the power supply voltage is allowed to vary to keep the performance fixed. For a fixed delay, the supply volt-

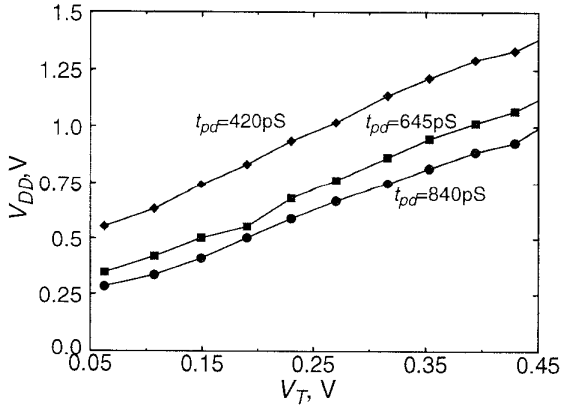

Figure 3: Experimental V_{DD} vs. V_T for a fixed delay.

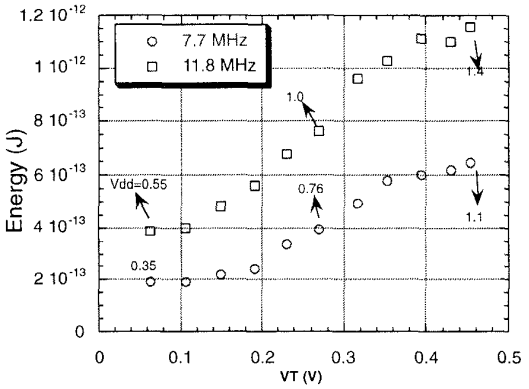

Figure 4. Experimentally derived optimum V_{DD}/V_T point.

age and therefore the switching component of power can be reduced while reducing the threshold voltage. However, at some point, the threshold voltage and supply reduction is offset by an increase in the leakage currents, resulting in an optimum threshold voltage and power supply voltage. That is, the optimum threshold voltage must compromise between improvement of current drive at low supply voltage operation and control of the sub-threshold leakage. It is interesting to note that the optimum voltage is significantly lower than 1V!

The switching activity plays a major role in determining the optimum threshold and power supply voltage. For example, a circuit which has very low switching activity will require a high-threshold voltage (and hence high power supply voltage).

4. Technologies for Event Driven Computation

Not all computations are continuously operational. An important class of high-performance computation is "event-driven" computation in which intermittent computation activity triggered by external events is separated by long periods of inactivity - examples include X-server, communication interfaces etc. An obvious mechanism for saving energy is to shut down parts of the system hardware that are idle because they are waiting for I/O from outside the system or from other parts of the system. For example, analyzing several traces obtained from real X-sessions indicates that the processor spends more than 95% of its time in the *off* state suggesting large energy reductions under ideal shutdown conditions [4].

While the processor is shutdown, the system should ideally consume near zero power. This is only possible if the devices consume low levels of leakage power - i.e., the devices have a high threshold voltage. However for low voltage high-performance operation, reduced threshold devices are required.To satisfy the contradicting requirements of high-performance during active periods and low-standby leakage, several device technologies have recently been introduced. This includes the control of threshold voltages in triple-well CMOS using backgate effect, multiple threshold devices, and dual-gate SOI technology. All these technologies provide a mechanism to trade-off speed and leakage power.

It is well known that the threshold voltage of MOS devices can be modulated using substrate bias. One proposal for low-voltage operation is to change threshold dynamically by changing the substrate bias - i.e., use a lower threshold voltage to operate at low power supply voltages during active periods and raise the threshold voltage during idle periods to lower leakage power [5]. One potential problem with this approach is that the threshold voltage changes in a square root fashion with respect to *source to bulk voltage* and therefore a large voltage may be required to change V_T by a few hundred mV.

Another approach to dynamically control leakage currents is to use a multiple threshold process [6]. The basic idea is that the logic circuits are implemented using low threshold devices and the low-V_T transistors are "gated" using high threshold switches which are in series. During idle periods, the high threshold devices are cut-off, significantly reducing the subthreshold conduction of the low V_T devices. During active periods, the high threshold switches are turned on and circuits resume normal low threshold high speed operation, provided proper device sizing.

A third approach is to use a silicon-on-insulator-with-active-substrate (SOIAS), to achieve dynamically variable threshold voltages [7]. Figure 5 shows the cross section of the SOIAS device. The back gate controls the threshold voltage (V_T) of the front gate device since the surface potentials at the front and back interfaces are coupled in fully depleted SOI devices. Leakage power and on-currents are controlled by lowering V_T when a circuit is active and raising V_T when the circuit is idle. This addresses the opposing requirements of high performance and low power, particularly at low power supply voltages. Figure 6 shows the measured I-V characteristics for an NMOS device for two different backgate voltages. A 250mV change in threshold voltage results in a 3.5-4 decade reduction in off current and an 80% switching current increase at 1V operation.

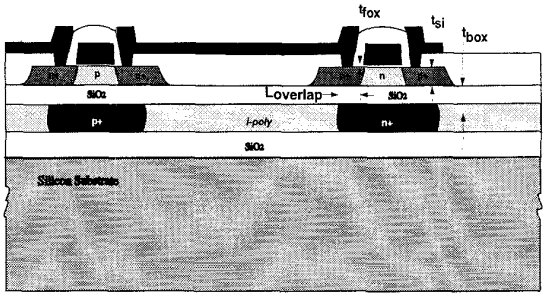

Figure 5: Silicon-on-Insulator Active Substrate.

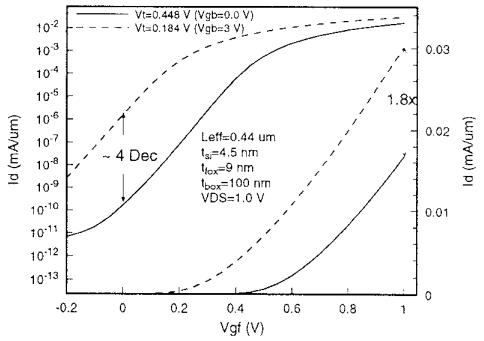

Figure 6: Measured I-V for a dynamically variable SOI NMOS.

5. Power Estimation of Burst Mode Technologies

The previous section described three different technologies for low-voltage operation. This section describes the power estimation tools required for low-voltage technologies. Specifically, the approach and tools required to compare the energy efficiency of two different technologies will be presented. A regular SOI device (with fixed low threshold devices) will be compared with the SOIAS device (with variable threshold devices) with respect to energy efficiency for a given *fixed* performance.

5.1 Definition of Activity Variables

In order to evaluate power consumption in low-voltage technologies which provide a trade-off between leakage power and speed, a few switching activity variables need to be defined. We will focus on SOIAS technology, but the analysis applies equally well to multiple-V_T technology or substrate controlled threshold voltages in bulk CMOS. Figure 7 shows a block diagram of an adder on an chip. Three activity variables are used in the model:

fga: The fraction of time the module (in this case the adder) is active. When the module is inactive, gated clocks (CLK ADD) can be use to "shut" down the unit to eliminate switching and conserve power.

bga: This represents the probability of a power consuming transition on the backgate for an SOIAS device. Clearly, as seen from Figure 7, the backgate activity (CLK BACKGATE) can be smaller than the front gate activity (for example, if the adder is on for a few cycles, then the backgate has to switch only once for the run of cycles the adder was on). If a multiple-V_T technology was used, then this activity would correspond to the activity of the control signal to switch ON/OFF the high V_T devices.

α: This is the individual node transition activity (assuming the module is always turned on) which is a strong function of signal statistics.

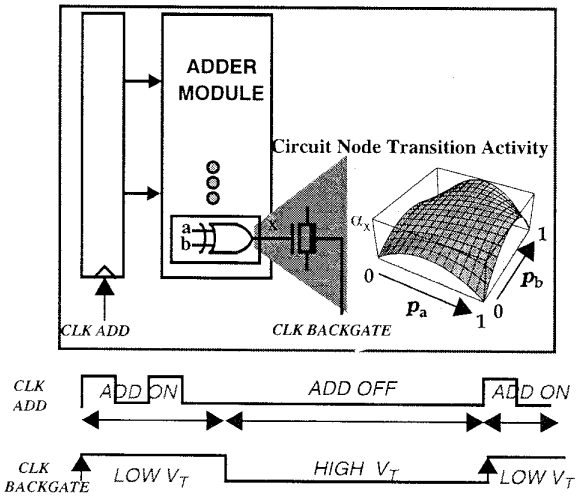

Figure 7: Activity considerations for SOIAS.

5.2 Energy Consumption Model

The first issue in developing a model for the operation of SOIAS back gated transistors is to determine the granularity of threshold voltage V_T control. The degree of V_T control ranges from affecting individual transistors to switching the V_T of the entire chip at once. Switching the threshold voltage of individual transistors could have applications in analog circuits or for large, infrequently used transistors in a digital system. In general however, controlling each transistor in a digital system individually would require a great deal of additional wiring to route the back gate control signals. Switching the entire chip, while requiring little wiring overhead, is only useful for systems which are idle for long periods of time but whose components are all active during system activity. We have chosen to assume a model of operation in which "functional units," or blocks, share a common V_T. Under this model, an active system's idle components are left in a low-leakage state.

The average energy per cycle in a standard SOI process (with a fixed low V_T) will be denoted as E_{SOI} and in a SOIAS process will be E_{SOIAS}. Standard SOI energy is the sum of the front gate energy, modeled as a CV^2 capacitive dissipation, plus the leakage current during the cycle. The front gate energy, which is the first term of Equation 3, has a factor *fga* • α since the front gate has activity modified due to block level power down (*fga*) and signal statistics (α). The device is continually leaking (since we assume that a standard SOI process has a fixed low threshold voltage), so the leakage energy is merely the leakage current $I_{leak(low)}$ times V_{DD} multiplied by the cycle time t_{cyc}. $I_{leak(low)}$ is the leakage current when the devices have low threshold voltage.

$$E_{SOI} = fga \bullet \alpha\, C_{fg}\, V_{DD}^2 + I_{leak(low)}\, V_{DD}\, t_{cyc} \quad (3)$$

The energy per cycle for a SOIAS gate has the same front gate component, but the back gate switches *bga* percent of the time. This energy is undesirable overhead dissipation. The compensation is that the leakage energy is reduced by a factor of nearly *fga*. The device is also leaking slightly, with a current $I_{leak(high)}$, even during the (1- *fga*) time the device V_T is high. SOIAS energy equations therefore are as follows:

$$E_{SOIAS} = fga \bullet \alpha C_{fg}\, V_{DD}^2 + bga\, C_{bg}\, V_{bg}^2 + fga\, I_{leak(low)}\, V_{DD}\, t_{cyc} + (1\text{-}fga)\, I_{leak(high)}\, V_{DD}\, t_{cyc} \quad (4)$$

5.3 Tools for estimating Activity Parameters

To evaluate trade-offs in power consumption, the various activity variables (*fga*, *bga*, α) have to be determined for a given application. For microprocessor applications, various code profiling packages exist, and the one we use for activity estimation is Pixie. These packages are generally designed to pinpoint code inefficiencies by noting the number of executions of subroutines or modules, or to guide the development of instruction set architectures through the measurement of instruction execution frequencies. For the purpose of measuring functional block activity, the execution frequency of individual assembly language instructions must be mapped to functional block use.

The first step in measuring functional block activity is to determine which assembly language instructions use which functional blocks. This requires that certain assumptions about the implementation be made. For instance, the ALU adder is generally used to compute load and store addresses and for comparison instructions. In our implementation, all add, compare, load, and store instructions use the ALU adder. A different implementation might use the ALU adder for more or fewer instructions. For each functional unit of interest, each instruction which uses that unit must be found. Pixie (or other profiling tools) can count the number of executions of each type of instruction. ATOM code is used to count the number of executions of all relevant instructions and compile a total [8]. Table 1 and Table 2 shows the output of ATOM for two different SPEC programs. We modified the tools to directly give *fga* and *bga*. Table 3 shows the activity values for a data encryption standard (IDEA).

Table 1. Profiling results for SPEC benchmark espresso

	Number	*fga*	*bga*
Total Instructions	900158847	-	-
Additions	543616709	0.6039	0.1954
Shifts	57000715	0.0633	0.0541
Multiplications	172883	0.0002	0.0002

Table 2. Profiling results for SPEC benchmark Li

	Number	*fga*	*bga*
Total Instructions	1737729538	-	-
Additions	661236960	0.6023	0.2233
Shifts	52224367	0.0087	0.0086
Multiplications	7088	0.0000	0.0000

Table 3. Profiling results for Data Encryption (IDEA)

	Number	*fga*	*bga*
Total Instructions	2125	-	-
Additions	1250	0.5882	0.2635
Shifts	186	0.0875	0.0753
Multiplications	3	0.0014	0.0014

As described earlier, *fga* is the ratio between the total number of uses of the functional block to the total number of executed instructions. *bga* is the ratio of the number of blocks of functional unit uses to the total number of executed instructions (so if all the uses of a block were sequential, *bga* would be 1/total instructions). ATOM is able to compute the profiling parameters for each functional block (each block is associated with a set of assembly language instructions which, for a typical implementation, use that block) in a single run and present the results (along with any other desired data) at the end of the run.

A variety of approaches can be used to determine the node switching activity factor α. This includes low-level circuit simulators such as SPICE, switch-level simulators (like IRSIM), or logic level simulators. Switch level simulators provide a compromise between simulation speed and accuracy. Our experiences with switch-level simulators shows that the estimated switched capacitance using calibrated technology files fits measured results within 10%.

Figure 8 shows the histogram of the transitions obtained from switch level simulation using IRSIM for an 8 bit adder, with random patterns applied to the inputs. Figure 9 is a result of one of the inputs fixed at 1 and the other input increments from 0 to 255. This plot shows that activity is significantly lower verifying that the node transition activity is a very strong function of signal statistics. This data includes the extra transitions due to glitching in static CMOS circuits.

5.4 Example: Application to X-server

The ratio of the total energy dissipation for SOIAS to SOI was analyzed as a function of algorithm and architecture dependent parameters (*fga* and *bga*). Figure 10 shows the log of the energy ratio, as a function of *fga* and *bga*. The plot also shows data points for two different applications obtained from the architectural profiling techniques described earlier. The zero contour (the dark solid

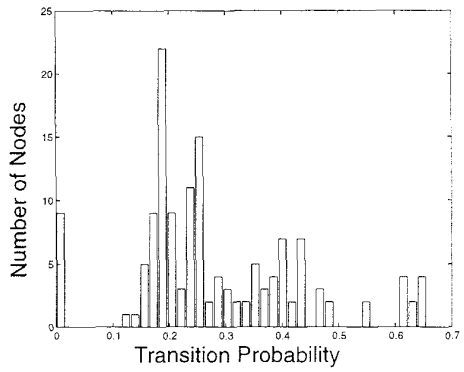

Figure 8: Histogram of transition activity for an 8-bit ripple carry adder with random inputs.

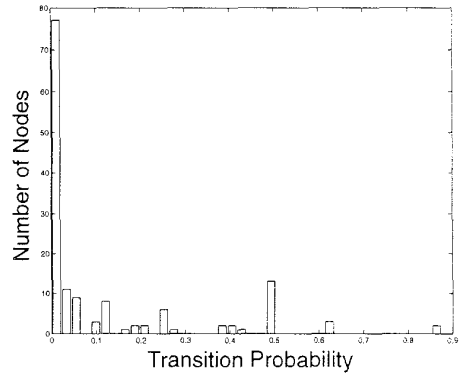

Figure 9: Histogram of transition activity for an 8-bit ripple carry adder with correlated inputs.

line) shows the breakeven point - i.e., points that lie below the line indicate a reduction in power using the SOIAS technology over a conventional SOI technology.

The top set of points for the adder, multiplier, and shifter are for the case when the processor is continuously active with the specific modules powered down when not in use. For this situation, there is little advantage going to the SOIAS technology. However, for a system which is frequently idle while awaiting I/O, such as an X server which is active 2% of the time, the SOIAS technology dissipates less energy than conventional SOI (the bottom set of points): 43% for the adder (*fga*=69.7%, *bga*=21.3%), 80% for the shifter(*fga*=10.9%, *bga*=8.7%), and 97% for the multiplier(*fga*=0.83%, *bga*=0.83%).

6. Conclusion

This paper described some of the key emerging technologies for low-voltage systems. For continuous mode circuits, optimizing threshold voltages enables supply voltages lower than 0.5V, yielding significant power reduction over current 3V operation without loss in performance. For event-driven computation, technologies

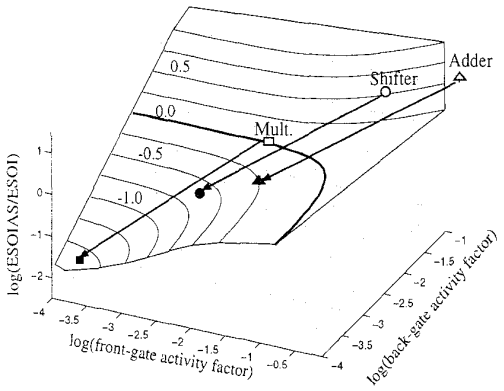

Figure 10: Log (E_{SOIAS} / E_{SOI}) as a function of activity variables.

which enable a trade-off between leakage current and speed are required. Tools to determine the block and node activities of logic elements were presented and the overall methodology to evaluate trade-offs between various low-power technologies was emphasized.

Acknowledgments

This work is sponsored by ARPA contract # DABT63-95-C-0088 and NSF Career Development Award MIP-9501995. I. Yang is supported by an AT&T fellowship.

References

[1] A. Chandrakasan, R. Brodersen, *Low Power Digital CMOS Design*, Kluwer Academic Publishers, July 1995.

[2] M. Alidina, J. Monteiro, S. Devadas, A. Ghosh, and M. Papaefthymiou, "Precomputation-Based Sequential Logic Optimization for Low Power," *1994 International Workshop on Low-power Design*, pp. 57-62, April 1994.

[3] Thomas Barber, "BodyLAN™: A Low-Power Communications System", SM Thesis, MIT, January 1996.

[4] M. Srivastava, A. P. Chandrakasan, and R.W. Brodersen, "Predictive System Shutdown and Other Architectural Techniques for Energy Efficient Programmable Computation," *IEEE Transaction on VLSI Systems*, pp. 42-55, March 1995.

[5] K. Seta, H. Hara, T. Kuroda, M. Kakumu, T. Sakurai, "50% Active-Power Saving Without Speed Degradation Using Standby Power Reduction (SPR) Circuit," *IEEE ISSCC 1995*, pp. 318-319, 1995.

[6] T. Sakata, M. Horiguchi, K. Itoh, "Subthreshold-Current Reduction Circuits for Multi-GIGABIT DRAM's," 1993 Symposium on VLSI Circuits, pp. 45-46, 1993.

[7] I. Yang, C. Vieri, A. P. Chandrakasan, and D. Antoniadis, "Back Gated CMOS on SOIAS for Dynamic Threshold Control," *IEEE 1995 IEDM*, December 1995.

[8] A. Eustace, A. Srivastava, "ATOM: A Flexible Interface for Building High Performance Program Analysis Tools", Digitial Equipment Corp. WRL Technical Note TN-44, July 1994.