On Estimating the Performance of VLSI Circuits

Clark D. Thompson
Prabhakar Raghavan

Computer Science Division
573 Evans Hall
University of California
Berkeley, CA 94720

ABSTRACT

Metrics of energy, area, and time are defined for a graph-theoretic model of VLSI computation. A number of "technological constant factors" are introduced in order to take into account the effects of using different technologies for implementing logic circuits. Different constant factors are seen to be appropriate for different logic families. We examine seven such families: NMOS, CMOS, CMOS-SOS, I^2L, GaAs HEMT, JJ-CIL, and JJ-CS.

1. Introduction

The area of VLSI modeling has been the scene of a disparity between circuit theorists and complexity theorists. The former have developed methods for the detailed study of circuit behavior, while the latter have used graph-theoretic models to derive asymptotic results.

The circuit-theoretic approach, while accurate to within the stability and robustness of the numerical methods used, is time-consuming; circuits involving more than a few transistors take extremely long to simulate. Further, in the process of producing the required information they usually generate a good deal of data that is not directly usable. Finally, these circuit techniques do no give any idea of the area of the chips they attempt to model; for this, detailed layout must be resorted to. The factors governing these layouts and simulations vary widely from technology to technology, so that the techniques used are not general in any sense.

Complexity theorists have sought to avoid these limitations, and have succeeded in a way; however, their techniques have resulted in some difficulties of a different nature. This approach [11, 14, 15] utilizes the concept of embedding a circuit as a graph on a planar grid, making statements about the asymptotic growth of area and time in these embedded circuits. Elegant proof techniques involving graph theory and information flow can be applied. Unfortunately, the very

This work was supported by the Semiconductor Research Corporation Grant SRC 1-44247-52055.

simplicity of the graph abstraction proves to be its major shortcoming. Gates are abstracted as grid points; in essence all the area is assumed to be taken up by wiring.

More important, the physical phenomena underlying the devices used in VLSI make many of the asymptotic results difficult to interpret; we might, for instance be faced with a result such as "The area of a chip solving problem **P** is N^2 for input size N" whereas in fact we actually mean "The area of a chip solving problem **P** is N^2 for input size N, provided the area does not exceed 100 mm^2." While there is more truth to the latter statement, the relation of asymptotic limits to physical constraints is hard to perceive. Further, designers and circuit modelers are often interested in evaluating different approaches to solve a problem that might only differ by a constant factor; such distinctions are lost in asymptotic studies.

We propose a new model here that attempts to tackle some of these problems. While it would be tempting to report that our model captures the best of both worlds by combining the simplicity of the complexity theoretic approach with the realism of the circuit simulators, there are a number of problems that prevent us from attaining this happy state of affairs. We outline these problems later in this report.

We have, however, developed a graph-theoretic formulation that includes the notions of gate size, wire size, fanout, gate and wire delay, wiring layers and signal propagation. More importantly, we have classified the modes of energy consumption in integrated circuits and supplied the means for evaluating this quantity. Finally, we have equipped the model with a number of "technological constants", thereby furnishing it with the versatility to deal with various fabrication technologies and logic families.

Section 2 is a statement of the model; this is followed by a list of values for technological constants for present and future technologies in section 3. Section 4 explains how the model may be "fitted" on to seven current technologies, and discusses some features common to all of them. The next section describes each of the technologies in some detail, and furnishes a list of technological constants. The final section concludes by outlining some of the problems faced in applying this model, and describing the framework within which it should be used.

2. Model of Computation

In the following assumptions, greek letters are used for the technological constant factors. (There are two exceptions: δ and Δ bear their standard meaning of vertex in- and out-degree.) Sets and their elements are defined by capitalized and lower case roman letters, respectively.

1. *Sources* O_h, *sinks* I_h. A computation graph is a directed hypergraph $G = (V, H)$. A hyperedge h is denoted by an ordered pair (O_h, I_h) of vertex sets $O_h \subseteq V$, $I_h \subseteq V$. The vertices in O_h are the s*O*urces of h; the vertices in I_h are its s*I*nks.

2. *Edge fanout restrictions o_{max}, ι_{max}.* Each edge h has at least one, and at most $O(1)$, sources and sinks:

$$1 \le |O_h| \le o_{max}, \qquad 1 \le |I_h| \le \iota_{max}$$

Limits on vertex indegree δ and outdegree Δ are discussed in Assumption 10.

3. *Vertex widths λ_{gate}, $\lambda_{I/O}$.* Each vertex in a computation graph is embedded as a square region in the Euclidean plane. No two vertices overlap. The size of an embedded vertex depends upon its functionality: gates $v \in V_{gate}$ occupy λ_{gate}^2 area, while I/O ports $v \in V_{I/O}$ occupy $\lambda_{I/O}^2$ area.

4. *Edge width λ_{wire}, number of wiring layers μ.* An edge is embedded as a connected set of wire segments. Each wire segment is a rectangle of width λ_{wire} and arbitrary length, placed on one of μ planar wiring layers stacked above the plane of the vertices. A wire segment on the bottom wiring layer connects to the vertices it passes over. Two wire segments are connected to each other if they pass over the same point and if they are on either the same layer or an adjacent layer. (Note that $\lceil \mu/2 \rceil$ disconnected wire segments may pass over the same point in the vertex plane. Also note that any hyperedge h can be embedded as a tree of wire segments passing over the vertices in $O_h \cup I_h$.)

5. *Total area A, maximum total area α_{max}.* The total area A of an embedded computation graph is the area of the smallest square that encloses all its vertices and wire segments. The area of this square is bounded by a technological constant: $A \le \alpha_{max}$.

6. *Maximum edge length λ_{max}.* The total length $||h||$ of an (embedded) edge h is the sum of the lengths of its wire segments. Edge lengths are bounded by a constant: $\forall h \; ||h|| \le \lambda_{max}$.

7. *Votes $v(t)$, signals $h(t)$.* The state of the computation graph at any time t is defined by a vector $(V(t), H(t))$ of votes $v(t)$ and signals $h(t)$ associated with each vertex v and hyperedge h. The value of a vote or signal is taken from the ternary set $\{0, 1, u\}$: logic-0, logic-1, and undetermined. (An alternative formulation, found in [11] and in state-of-the-art circuit simulators, takes signal values from a two-dimensional set of voltages V and impedances R.)

8. *Maximum size of voting equivalence class ξ_{max}, edge delay d_h, time constants τ_{gate}, τ_{wire} and τ_{fanout}, transmission line indicator ς, signal rise time r_h.*

   a. In many technologies, the delay associated with a wire can be decreased by driving that wire with a larger transistor. Such high-power drivers can be represented by several (unit-power) sources with identical voting behavior. We are thus led to the following definition of equivalence classes $C_{h,i}$ on the voting behavior of the sources for each edge h:

$$v_1, v_2 \in C_{h,i} \iff (v_1, v_2 \in O_h) \wedge (\forall t \;\; v_1(t) = v_2(t))$$

A technological limit on driving power translates into a restriction on the size of (*i.e.*, number of vertices in) any voting equivalence class:

$$\forall h, i \quad | C_{h,i} | \leq \xi_{max}$$

b. At the time of circuit construction ($t = 0$), a fixed but indeterminate delay d_h is assigned to each edge h. An edge's delay (in a worst-case analysis) is proportional to its length $||h||$ and number of sinks $| I_h |$, and inversely proportional to the size of its smallest equivalence class $C_h = \min_i | C_{h,i} | \leq \xi_{max}$:

$$d_h = \tau_{gate} + \frac{||h||\tau_{wire} + | I_h | \tau_{fanout}}{C_h} \ (\pm\ 50\%)$$

(Indeterminacy is introduced into the definition of edge delays to force "realistic" design practices, *e.g.*, self-timed or clocked logic.)

c. We define r_h to be the rise time of a signal on edge h. For technologies in which wires are transmission lines, r_h is approximately equal to the gate delay τ_{gate}. We indicate this by assigning the value 1 to the 0-1 variable ς (a mnemonic is the common symbol Z for the impedance of a line). The other technological possibility is that the wires are essentially capacitive in nature (as long as their length does not exceed λ_{max}, as defined in Assumption 6). Thus

$$r_h = \begin{cases} d_h, & \text{if } \varsigma = 0 \\ \tau_{gate}, & \text{if } \varsigma = 1 \end{cases}$$

d. The value of a signal $h(t)$ is determined by the votes of its sources O_h, with delay d_h. We prevent the propagation of unreasonably-short signal pulsewidths by requiring the "election results" to be stable for at least r_h time units.

$$h(t) = \begin{cases} 1, \ \text{if } \exists v \in O_h \ \forall s \in [t-d_h, t-d_h+r_h] \ \ v(s) = 1, \ \ \textit{else} \\ 0, \ \text{if } \exists v \in O_h \ \forall s \in [t-d_h, t-d_h+r_h] \ \ v(s) = 0, \ \ \textit{else} \\ u \end{cases}$$

Note that this formulation allows "wire-oring": the signal on an edge becomes 1 if any of its source votes is 1 for at least r_h time.

9. *Symmetry indicator σ.*

a. Not all patterns of voting behavior are allowed in all technologies. One restriction is observed in the so-called "symmetric" technologies ($\sigma = 1$). In these technologies, the effects of logic-1 votes and logic-0 vote are symmetric, making "wire-oring" infeasible. (A system of "majority-rule" is conceivable but not observed in any present-day logic family, possibly because it would reduce noise margins.) To outlaw wire-oring, we permit just one equivalence class per edge:

$$(\sigma = 1) \implies (\forall t \ \forall h \ \forall v_1, v_2 \in O_h \ \ v_1(t) = v_2(t))$$

b. A second type of restriction on allowable voting behavior arises in the asymmetric ($\sigma = 0$) technologies. We must restrict the number of high-power logic-1 votes that appear at one time on an edge, to avoid exceeding the current density limit mentioned in Assumption 8a:

$$(\sigma = 0) \implies \left(\forall t \forall h \quad |\{v \in O_h \ : \ v(t) = 1\}| \le \xi_{max}\right)$$

10. *Logic family ϕ, power supply period τ_{supply}, I/O schedule S, external clock period $\tau_{I/O}$.*

a. A logic family ϕ is a technologically-constrained set of triples (δ, f, Δ). The first and third parts of a triple denote the indegree and outdegree of one type of gate. The second part of a triple defines a functionality, or voting behavior. A gate with the 'and' functionality, for example, is modelled by a vertex whose vote is the logical 'and' of the signals on its in-edges. As another example, the 'latch' function depends on a delayed feedback signal. Finally, the voting behavior of gates in the JJ-CIL technology depends upon the phase of their AC power supply. Thus, in the general case, the functionality f_v of a gate v has $2 + \delta_v$ parameters, and defines the gate's vote as follows:

$$v(t) = f_v\left(v(t - \tau_{wire}),\ c(t),\ h_1(t),\ h_2(t),\ \ldots\ h_{\delta_v}(t)\right)$$

where the phase of the power supply (assumed to have a 90% duty cycle) is

$$c(t) = \begin{cases} 0, & \text{if } t \le \left(.1 + \lfloor t/\tau_{supply} \rfloor\right) \tau_{supply} \\ 1, & \text{otherwise} \end{cases}$$

Note that voting is a zero-delay process, since gate delays were included in the definition of edge delay d_h.

b. An I/O port $v_i \in V_{I/O}$ has $\delta_v = 1$, $\Delta_v = 1$. Its voting is determined by an externally-imposed I/O schedule $S_i \in \{r_0, r_1, r_u, w_0, w_1, w_u\}^*$. Each literal in S_i indicates whether the I/O port is to read (r_0, r_1, r_u) or write (w_0, w_1, w_u) a '0', a '1', or a 'u'. The k-th literal in S_i refers to the k-th external clock period defined by $t \in ((k-1)\tau_{I/O}, k\tau_{I/O}]$, where $\tau_{I/O}$ is a technological constant. Thus, if the k-th literal is r_x, the port votes $v_i(t) = x$ during the k-th clock period. Alternatively, if the k-th literal is w_y, we say the schedule S_i is "satisfied" only if the port's in-edge h has signal $h(t) = y$, for all times t in the k-th clock period. (If the output bit for some time period is u, i.e. undetermined, we allow $h(t)$ to be any value.)

11. *Energy consumption $E_{standby}$, $E_{1\text{-}0}$, E_{wire}, E_{sink}, E.* Four modes of energy consumption are observed in physical realizations of computation graphs.

a. A constant power dissipation of $\epsilon_{standby}/\tau_{gate}$ is associated with every gate. The worst case (over all I/O schedules S) total "standby" energy dissipation over the period $[t_1, t_2]$ is thus defined as

$$E_{standby} = \max_{S} \sum_{v \in V_{gate}} \frac{(t_2 - t_1)}{\tau_{gate}} \epsilon_{standby}$$

b. In asymmetric (wire-or) technologies, a gate voting 1 consumes more power than a gate voting 0. We define energy $\epsilon_{1\text{-}0}$ so that the difference between these two levels of power consumption is $\epsilon_{1\text{-}0}/\tau_{gate}$. Total energy consumption in this mode is thus

$$E_{1\text{-}0} = \max_{S} \sum_{v \in V_{gate}} \int_{\substack{t_1 \le t \le t_2 \\ v(t) = 1}} \frac{\epsilon_{1\text{-}0}}{\tau_{gate}} dt$$

By Assumption 8d, a gate's vote can change a signal only if it persists for at least $\tau_h \ge \tau_{gate}$ time. We thus employ the following (approximate) expression for $E_{1\text{-}0}$:

$$E_{1\text{-}0} = \max_{S} \sum_{v \in V_{gate}} \sum_{\substack{t_1/\tau_{gate} \le k \le t_2/\tau_{gate} \\ v(k\tau_{gate}) = 1}} \epsilon_{1\text{-}0}$$

c. Each change in an edge's signal consumes energy proportional to the length of that edge. Assuming such signal changes occur at a frequency less than $1/\tau_{gate}$, we write

$$E_{wire} = \max_{S} \sum_{h} \sum_{\substack{t_1/\tau_{gate} \le k \le t_2/\tau_{gate} \\ h(k\tau_{gate}) \ne h((k+1)\tau_{gate})}} ||h|| \epsilon_{wire}$$

d. Energy E_{sink}, like E_{wire}, is a form of "switching energy." In this case, the energy consumption is proportional to the number of sinks:

$$E_{sink} = \max_{S} \sum_{h} \sum_{\substack{t_1/\tau_{gate} \le k \le t_2/\tau_{gate} \\ h(k\tau_{gate}) \ne h((k+1)\tau_{gate})}} |I_h| \epsilon_{sink}$$

e. The total energy consumed by a computation is $E = E_{standby} + E_{1\text{-}0} + E_{wire} + E_{sink}$.

3. Technological Parameters

In this section we give a list of technological constant values for seven technologies. Present-day values as well as projected values for circuits fabricated in the late eighties are listed.

Table 1 gives approximate values for the "constant factors" of seven VLSI technologies with current fabrication and circuit-design techniques.

An important feature of Table 1 is the diagonal structure of the entries for circuit energies $\epsilon_{standby}$, $\epsilon_{1\text{-}0}$, ϵ_{wire}, and ϵ_{sink}. When calculating total energy, contributions from entries below the diagonal can be ignored. For example, the technologies with $\epsilon_{standby} > 0$ have a nearly constant power dissipation per gate which does not increase by more than 10% when the gates change their state at maximum frequency.

Table 1 is not quite a complete list of the parameters in the model. The following are nearly constant over all current technologies:

$$\alpha_{max} = 10^8\ um^2, \quad \tau_{I/O} = 20\ ns,$$
$$\mu = 4 \text{ to } 6 \text{ layers}, \quad \lambda_{I/O} = 10^2\ um.$$

	I²L	JJ-CIL	NMOS	HEMT	JJ-CS	CMOS	CMOS-SOS	units
λ_{gate}	10	60	70	70	100	100	100	um
λ_{wire}	4	10	4	4	10	4	4	um
λ_{max}	10^4	10^5	10^4	10^4	10^5	10^4	10^4	um
$\tau_{gate}, \tau_{fanout}$	2000	20	500	50	500	2000	2000	ps
τ_{wire}	1	0.02	1	1	1	1	0.5	ps/λ_{wire}
τ_{supply}		1000						ps
$\epsilon_{standby}$	10	0.01	~0	~0	~0	~0	~0	fJ
ϵ_{1-0}			1	0.1	~0	~0	~0	fJ
ϵ_{wire}					0.02	1	0.01	fJ/λ_{wire}
ϵ_{sink}							1	fJ
ξ_{max}	1	1	10	10	1	10	10	--
σ	0	1	0	0	1	1	1	--
ς	0	1	0	0	0	0	0	--
o_{max}	10^2	1	10^2	10^2	1	$(=\xi_{max})$	$(=\xi_{max})$	--
ι_{max}	1	1	10^4	10^4	10^4	10^4	10^4	--

Table 1. Current constant factors for seven VLSI technologies.
(Multiplicative factors for units are $f = 10^{-15}$, $p = 10^{-12}$, $n = 10^{-9}$, and $u = 10^{-6}$.)

Note that, by Assumption 4, $\mu = 6$ corresponds to a three-level metal process. The other $\mu - 3$ layers are made of an insulating material, through which small square holes or "vias" are cut.

Table 2 presents technological constants for the future, (hopefully) valid for late-1980s fabrication. The following are nearly constant over all our futuristic technologies:

$$\alpha_{max} = 10^9\ um^2, \quad \tau_{I/O} = 10\ ns,$$
$$\mu = 4 \text{ to } 6 \text{ layers}, \quad \lambda_{I/O} = 50\ um.$$

Table 3 indicates the availability of gates in each of the seven technologies; we use these gates as the basis of our conservative estimates for various parameters in section 5. The use of pass-transistors on the input of a 2-input NOR in NMOS gives us a 4-input NOR in this family. (Following Assumption 8b, we adopt a negative-logic convention for NMOS. A signal voltage of V_{DD} is thus a logic-0, and a Mead-Conway-style NAND gate becomes a NOR gate.)

	I^2L	JJ-CIL	NMOS	HEMT	JJ-CS	CMOS	CMOS-SOS	units
λ_{gate}	4	15	7	7	40	10	10	um
λ_{wire}	0.5	2	0.5	0.5	4	0.5	0.5	um
λ_{max}	$5{\cdot}10^4$	10^5	$5{\cdot}10^4$	$5{\cdot}10^4$	10^5	$5{\cdot}10^4$	$5{\cdot}10^4$	um
$\tau_{gate}, \tau_{fanout}$	1000	5	100	10	200	100	100	ps
τ_{wire}	0.1	0.005	0.1	0.1	1	0.1	0.1	ps/λ_{wire}
τ_{supply}		1000						ps
$\epsilon_{standby}$	1	0.002	$\sim$0	$\sim$0	$\sim$0	$\sim$0	$\sim$0	fJ
ϵ_{1-0}			0.05	0.1	$\sim$0	$\sim$0	$\sim$0	fJ
ϵ_{wire}					0.02	0.05	0.001	fJ/λ_{wire}
ϵ_{sink}							0.05	fJ
ξ_{max}	1	1	10	10	1	10	10	--
σ	0	1	0	0	1	1	1	--
ς	0	1	0	0	0	0	0	--
o_{max}	10^2	1	10	10^2	1	$(=\xi_{max})$	$(=\xi_{max})$	--
ι_{max}	1	1	10^4	10^4	10^4	10^4	10^4	--

Table 2. Futuristic constant factors for seven VLSI technologies.

4. Features Common to All Technologies

In this section we concern ourselves with those features of the model common to all the technologies we will study in the next section.

4.1. Derivation of Model Parameters

The model we have just delineated includes a number of "technological constants" to cater to the wide variety of technologies and logic families available. The question then arises - how do we arrive at values for these constants? In other words, given a description of a process for manufacturing devices for digital logic and the circuit characteristics of these devices, we require a procedure for assigning values for these constants. We begin this section by providing some intuition into this process. Following this we deal with some aspects of circuit behavior common to all technologies, notably the properties of interconnect lines and their futuristic trends. A

$(\delta,\Delta).f$	I^2L	JJ-CIL	NMOS	HEMT	JJ-CS	CMOS	CMOS-SOS
(1,1).INVERTER			*	*	*	*	*
(1,4).INVERTER	*						
(1,1).CLOCKED INVERTER		*					
(2,1).NAND,NOR			*	*	*	*	*
(2,1).AND,OR		*			*	*	*
(2,1).ARBITRARY						*	*
(4,1).NAND						*	*
(4,1).NOR			*			*	*

Table 3. Gate availabilities in seven VLSI technologies (availability indicated by *).

detailed discussion of individual technologies and parameter values is deferred to the next section.

The most fundamental specification for any technology is the *linewidth*, or the size of the smallest feature that can be fabricated. It is reasonable to assume that the smallest wires have this width, and require as much spacing between adjacent wires. The parameter λ_{wire} is thus assumed to be twice the linewidth for most technologies. An exception is the Josephson current-steering technology (or, indeed, any technology that transmits information via a current loop rather than a voltage referenced to a universal ground plane); in this case we have to consider the fact that every physical wire is accompanied by a return path for closing the loop. We then take λ_{wire} to be *four* times the linewidth.

From the device characteristics, we could determine the types of gates that can be made. In the literature, this specification is usually given along with the process description. It is then possible to derive two features of interest to us. We directly have the feasible family of boolean functions (δ,f,Δ) available in the logic family ϕ. From the dimensions of the devices made in the process, we can then form an estimate of λ_{gate} by a simple layout process. Frequently, we are spared this task as figures for gate size (or, equivalently, gate density) are given by the manufacturers/researchers who describe the logic family. The figures reported in Table 3 reflect the size of the largest of the gates in ϕ for a given technology.

Timing estimates are generally the hardest to form from device-level data. The behavior of individual devices and gates differs significantly from the behavior of complex networks of such elements. The presence of long interconnect lines further compounds the problem, as the properties of these are known to fall into several regimes [2]. The primary objective of our imposition of the limit λ_{max} is, in fact, to avoid the problem of operating lines in the diffusion regime.

Most descriptions of new logic give delay figures for basic inverters (either by themselves or in a ring-oscillator configuration). While this is a good value to adopt for τ_{gate}, it gives little information about τ_{fanout} or τ_{wire}. Since the figure given for inverter/gate delay usually assumes that the gate is driving an identical gate, we can form a conservative estimate by setting τ_{fanout} also equal to this figure. Note that τ_{gate} includes two components - one for the intrinsic switching time of the gate (in some sense, a "no-load" switching time), and the other for driving the load (in most technologies, this appears to dominate). Only the latter component is augmented by the addition of fanout; hence the conservative nature of this estimate for τ_{fanout}.

τ_{wire} can only be determined if we have figures that give the reactive properties of the interconnect lines (capacitance/inductance per unit length). We then compare this figure with the capacitance/inductance (depending on whether signals are propagated as voltages/currents) of a gate input, and determine what fraction of a gate load a unit length of line constitutes. τ_{wire} is then given by the product of this fraction and τ_{fanout}.

4.2. Interconnect Lines

We now turn to the characteristics of interconnection lines, as their behavior exerts considerable influence on timing and energy dissipation. Recent work on the properties of interconnection lines on various substrates is reported in [16]. Saraswat and Mohammadi have made some predictions concerning the scaling behavior of interconnect lines in [13]. Their formulation for capacitance, however, appears somewhat simplistic in that they assume a parallel plate model by which they are able to predict a linear reduction in capacitance. We will discuss this issue later in this section.

Any transmission line is characterized by four impedance properties - the series resistance and inductance, and the shunt conductance and capacitance per unit length. Since the fabrication of integrated circuits generally utilizes heavy layers of insulation per unit length, the conductance component can be neglected safely. The resistive component depends on the material used for the lines; while aluminum and polysilicon have conventionally been used, metals like tungsten and titanium (and their silicides) are under investigation. For current linewidths these have a line resistivity of about 500 Ω/cm. This resistivity increases as lines are made smaller; while the value increases as the square of the scaling factor (a figure >1, by which all dimensions are divided), it is important to recognize the fact that the absolute resistance for some fixed line-length (say λ_{gate}) is a more meaningful metric for comparison. In some sense, this gives us an idea of the resistance required to achieve the same degree of *functionality* , in this case the connection of two minimum-sized gates (gate-sizes presumably scale at the same rate as interconnect lines). This figure scales up linearly as dimensions are scaled.

Capacitances are harder to deal with, since they depend on the material used for the substrate and the insulation rather than the line itself. Further, interactions between adjacent lines introduce additional capacitances that are not easy to account for. Finally, the simple parallel plate model of [13] will not really hold since "fringing effects" become particularly dominant at small linewidths. The parallel plate model predicts that capacitance per unit line-length remains essentially constant, while the capacitance per λ_{wire} scales down linearly. By taking a more rigorous approach, Yuan *et al.* [16] show that even at present (1-5 μm) linewidths, the latter figure does not diminish quite as rapidly for silicon substrates. Gallium arsenide and sapphire substrates are already into this zone of sublinear capacitance scaling. Current values for capacitance per λ_{wire} are in the region of 0.05 - 0.1 femtofarads per λ_{wire}, for various substrates. [16] also indicates that these figures rise by as much as a 100% for gallium arsenide and sapphire substrated when multiple lines are considered (due to inter-line capacitances). We therefore take the 0.1 fF value for current technology.

The results mentioned above show that the RC product (per λ_{wire}) will not remain constant, due to the fact that the increase in resistance will not quite be counterbalanced by the decrease in capacitance. We are not considering such figures as the delay on the longest line of the largest chip that can be fabricated in a technology (as do Saraswat and Mohammadi), the significance of which is not entirely obvious (large die sizes are not necessarily a part of the scaling process; further, it is not clear that such long lines are inevitable). It appears that with metal lines in current technology, we have the ability to transmit signals at delays below one nanosecond per cm. It also appears that the most important concern for designers of the future is to reduce the series resistance of lines; "tall" lines (small width and large height) could alleviate this problem, but are hard to etch.

5. On the Modeling of Seven Technologies

In this section we will look into the properties of computational devices manufactured by seven technologies in some detail.

5.1. NMOS

We choose to begin our description of technologies with NMOS, because of its widespread popularization by Mead and Conway. Some of the salient features of NMOS logic are noted below, following which we give a listing of model parameters and their future trends.

Logic functions can be realized by two major methods in NMOS: *active gates,* and *pass-transistor networks.* Attempts have been made to realize logic by means of more intricate MOS networks [12]; but these attempts have not been very successful and have in fact been discouraged by advocates of structured design techniques. We will therefore study computations involving just these two NMOS elements.

In our model, we depict gates by vertices in the hypergraph. Modeling pass-transistors explicitly has proven very complicated, and we choose instead to absorb them into the gates following them. Since good circuit design practice precludes long chains of pass-transistors (to maintain logic levels) we impose the constraint that no more than two pass-transistors can be connected serially without level restoration by buffers/gates.

We anticipate reductions of λ_{gate} and λ_{wire} to about a tenth of their current values by the beginning of the next decade. We also note that modeling the temporal behavior of transistors driving large loads (wire/fan-out) is a complex problem; we have therefore adopted the conservative technique of separating the gate- and wire-delay components. A good treatment of the limit λ_{max} is given in [2]. The figures reported there are 10 *mm* for present technology and 50 *mm* in the future.

The work of Hoeneisen and Mead [8] and of Hart *et al.* [7] suggests that the ultimate limits on MOS speed will be reached at around a tenth of a nanosecond; this limit stems from the fundamental physical properties of MOS devices.

In NMOS it is possible to connect the outputs of several gates (sharing a common pullup) so that the output of the composite structure (at the lower end of the common pullup) is pulled down if any one of the gates (voters) conducts. In keeping with popular convention, we speak of this as a wire-or configuration (the exact logic function realized is, of course, dependent on which logic convention - positive or negative - is assumed). We restrict the number of voters in a wire-or configuration to a hundred; this is because a pull-down has a reverse-junction leakage current even when "OFF". The presence of a large number of pull-down voters increases this leakage component; when all of them are "OFF", the trickle current may be large enough to lead to the deterioration of the vote on the edge. In scaled transistors, the ratio of OFF-resistance to ON-resistance is even lower, and we expect the maximum possible number of wire-or voters to decrease to about ten.

For similar reasons we restrict o_{max} to be 10; i_{max} is restricted to 10^3 because of the limitation on λ_{max}.

The major phenomenon of interest to us is that an NMOS gate dissipates power when "ON" i.e. when the pulldown network is conducting. In this condition, the power dissipation is governed by the size of the pull-up alone, since the pull-down resistance is small in comparison. Over the next decade, we expect $\epsilon_{1\text{-}0}$ to drop from its present value in the femtojoule range to the order of a hundredth of a femtojoule. At this point we again run into fundamental limits, and intricate circuit techniques must be resorted to for lowering the energy dissipations of NMOS circuits; these may include the use of coding techniques, intermittently operational circuits (between bouts of activity, the circuit is "rested" to cool down) and other tricks. The importance of such techniques is supported by the studies of [8], which indicate that in large-scale circuit configurations the power dissipation constraint could reach a critical stage before the physical limits of individual devices become a problem.

5.2. CMOS and CMOS-SOS

The devices used in CMOS are essentially the same as those in NMOS, and the same fundamental limits apply here as well. Logic is realized through a combination of complementary gates and pass transistors. We shall follow the same convention for pass-transistors here as we did in the case of NMOS.

The study of CMOS limits is a little more subtle than NMOS, since the dominant effects are circuit-dependent (as opposed to device-dependent) to a greater extent. [8] suggests, for instance, that speed of operation would be constrained by power dissipation rather than by device physics. This is because almost all the dissipation in CMOS is during state transitions; this imposes a maximum rate of occurence of transitions within a circuit.

We thus give a 2 ns figure for gate delay with present technology; this could drop by a factor of ten over the next decade. Clearly, wire-oring is not permitted, at least in vanilla CMOS. The parameters o_{max} and i_{max} are restricted to ten and one thousand, as with NMOS.

Depending on whether we consider ordinary CMOS or CMOS fabricated by the SOS technique, energy dissipation is dominated by ϵ_{wire} or ϵ_{sink}. This is because CMOS-SOS has very low interconnect capacitance making ϵ_{wire} small. The dominant form of energy dissipation in each style is currently of the order of a femtojoule; this could scale down to about 0.05 fJ if the dissipation density does not lead to thermal failure. The problem, however, is not as acute as with I^2L, where the dissipation is continuous; the fact that CMOS dissipation occurs mainly during transitions could conceivably be exploited using circuit design and system timing schemes which exploit the speed of individual gates to the fullest extent possible without causing them to change state too often.

5.3. Integrated Injection Logic

Integrated Injection Logic is perhaps the most promising bipolar technology for VLSI, mainly by virtue of its low power dissipation. In addition, this logic family is dense and is particularly suitable for gate arrays. A typical gate consists of a four-output inverter whose outputs can be wire-ored with the outputs of other inverters.

While a number of studies give results on the intrinsic limitations of the bipolar components used for I^2L, little has been said about the behavior of this logic family in circuit configurations. We rely primarily on the work of Evans [4] and of Hart *et al.* [7] for the constant factor estimates given here.

With current linewidths in the one micron range, the parameter λ_{gate} is of the order of ten microns. Devices with τ_{gate} as low as 2 *ns* are within the realm of currently available technology, but this is already close to physical limits imposed by collector-base capacitances. It is thus reasonable to conclude that in the case of I^2L, the benefits of scaling will consist mainly of lower injection current requirements for minimum-delay devices. $\epsilon_{standby}$, currently of the order of a hundredth of a picojoule, can be expected to fall to about one femtojoule over the next decade. There could be a problem regarding the density of power dissipation on the chip, since I^2L is a very dense family.

The problem of reverse leakage through wire-ored transistors is not as serious as for NMOS, and we permit up to a hundred sources for a hyperedge. The number of sinks, however, clearly cannot exceed one.

5.4. Josephson Junction - Current Injection Logic

IBM has fabricated circuits out of current injecting Josephson junction devices [5]. Projections of circuit performance under device and circuit scaling are given in [9]. Our description of current injection logic here is based mainly on these reports.

There are two basic gates, a two-input AND and a two-input OR; inversion is accomplished by means of a "clocked inverter" [5]. Gheewala reports figures on "four-input gates" as well, but these appear to be made up of two-input gate structures. Moreover, they do not differ significantly in performance from a tree of two-input ANDs. A feature unique to this technology is that gates are latching; in order to perform a new computation with a gate, it is necessary to turn off the power supply briefly. Circuits using this technology thus utilize power supplies that are cycled periodically (at approximately one nanosecond intervals in current practice).

Current linewidths in this technology are of the order of two microns; but the presence of a number of inductances and resistances in the gates raises gate size to about 4000 μm^2, yielding a figure of about 60 μm for λ_{gate}. Signals propagate along interconnect lines that are essentially transmission lines. However, we impose the restriction that a signal transmitted during one power supply cycle should be received during the same cycle at the "far end". This is a somewhat conservative approach to synchronization, and limits the maximum length of interconnect transmission lines to about a tenth of a meter or $10^5 \mu m$.

Gheewala has split delays in this technology into three components: (i) gate delay, (ii) crossing delay, and (iii) propagation delay. While gate and propagation delay have clear-cut analogues in other technologies, the term crossing delay requires some explanation. Since information transfer is in the form of a current rather than a voltage, the output of a gate drives the input of another if the latter is a part of the output current loop of the former. A current pulse traveling on this loop suffers a delay in "crossing" the gate being driven; this corresponds to fanout delay in technologies like NMOS and CMOS.

Current values for these parameters are in the range of 10 *ps* for τ_{gate} and for τ_{fanout}, and .01 *ps*/μm for τ_{wire}. Power dissipation is not a characteristic of any single state assumed by the gates, but takes on a steady-state form instead. Gheewala reports values in the range of 1-10 μW for power dissipation, which yields figures in the range of 0.01–0.1 *fJ* for $\epsilon_{standby}$: we adopt the upper limit for our estimate.

Ko and Van Duzer suggest [9] that in this technology, area and delay cannot be minimized simultaneously. Their work indicates, however, that gate delay can be brought down to about 5 *ps* by means of various optimizations, while propagation delay can be cut by a factor of four. Crossing delay seems harder to reduce, and it is likely that a heavy price will be paid for fanout in current injection logic of the future.

5.5. Josephson Junction - Current Steering Logic

Current-steering logic is an alternate form of Josephson junction circuitry, developed for use in conjunction with single-flux-quantum memory devices [6]. Current-steered superconducting loops form the basis for logic implementation; this has a useful property we will discuss below. Guéret *et al.* have demonstrated the feasibility of a complete family of logic gates consisting of a two-input AND, a two-input OR and an inverter. It is worth noting that all gates provide both true and complemented outputs, so that in effect we have NANDs and NORs as well; this is because current is switched between two loops each of which could drive other gates (somewhat like ECL).

From the figures reported in [6], we can arrive at an estimate of 500 *ps* for τ_{gate} and τ_{fanout}. Values for λ_{gate} are of the order of 100 μm, with 10 μm lines. Guéret *et al.* state that energy dissipation occurs only during switching events; the magnitude depends on the loop inductance and hence the wire length, so that this is a case of ϵ_{wire} dissipation. In addition, however, we expect an $\epsilon_{standby}$ component in the current source(s) driving the circuit.

The figure for ϵ_{wire} dissipation is not easy to estimate as in other technologies, since fanout plays a very complicated role in the process. Large fanouts call for long output loops, the inductance of which determines the dissipation. However, the inductance can be prevented from growing linearly with the fanout by increasing the width of the line. We thus give a somewhat conservative estimate of 0.02 fJ/λ_{wire} for present technology.

The loop structure restricts o_{max} to one.

5.6. Gallium Arsenide Logic Circuitry

Recent advances in gallium arsenide technology have paved the way for high performance combinational logic. The two classes of GaAs circuit technology that exhibit the most promise are self-aligned MESFET circuits and High Electron Mobility Transistor (HEMT) circuitry [1]. A more comprehensive review of the various kinds of GaAs circuitry can be found in [3]. Our figures here are mainly from these sources.

Current linewidths are of the order of a micron, as with other technologies. The most complex GaAs chip known is the 8 x 8 multiplier of Lee *et al.* [10], with over a thousand gates and measuring 2.7mm by 2.25mm. A gate density of 33000 gates/cm^2 was reported then, while Abe *et al.* exhibit a somewhat higher density in [1]. From these figures it is reasonable to assume a figure of 10 μm for λ_{gate} with present technology.

Gate delays approaching 10 ps have been reorted in recent GaAs literature [10].

6. Conclusion

Our discussion of the model would be incomplete without a list of the problems encountered in using it. We therefore conclude this report with a summary of situations that are hard to model, and the limitations of our model as a consequence of these situations.

Practical memory devices are not made of static latch circuits unless speed is of the essence. In fact, information storage is often realized by means of very much simpler circuitry - such as the single-transistor cell in dynamic MOS memories. The inability of our model to deal with such phenomena forces us to use expensive static circuits for storage in any constructions using this model.

The second major shortcoming of the model is in the treatment of switch devices, like pass-transistors in NMOS. In practice, the availability of these devices often reduces circuit area significantly.

It should be stressed that ours is a strictly "upper bound" model, and generally overestimates the area/time/energy metric. Our goal in developing this model was to to be able to form estimates that are within an order of magnitude of the exact value. Some of these conservative assumptions were made necessary by our objective of catering to several technologies, given the diversity of the underlying physical phenomena.

A major application of this model would thus consist in the evaluation of different circuit solutions for a given problem. Asymptotic predictions of circuit complexity and performance (under different metrics) can be made, subject to technological limits; for instance, it may be possible to predict that the area of the chip solving a certain problem grows as the square of the size of the input, as long as the area does not exceed α_{max} and the longest wire in the circuit does not exceed λ_{max}.

Within the costraints of these limitations, our model permits some useful comparisons and estimates; in particular, our provision for the evaluation of energy consumption is perhaps the most general classification of energy dissipation modes in VLSI to date. Our experience in using the model has been that it avoids the minute details that complicate circuit simulation models, while retaining a more realistic picture of reality than existing graph-theoretic models.

References

1. Masayuki Abe, Takashi Mimura, Naoki Yokoyama, and Hajime Ishikawa, "New Technology Towards GaAs LSI/VLSI for Computer Applications," *IEEE Trans. on Electron Devices*, vol. ED-29, no. 7, pp. 1088-1093, July 1982.

2. G. Bilardi, M. Pracchi, and F. P. Preparata, "A Critique of Network Speed in VLSI Models of Computation," *IEEE Journal of Solid-State Circuits*, vol. SC-17, no. 4, pp. 696-702, August 1982.

3. Richard C. Eden, Bryant M. Welch, Ricardo Zucca, and Stephen I. Long, "The Prospects for Ultrahigh-Speed VLSI GaAs Digital Logic," *IEEE Journal of Solid-State Circuits*, vol. SC-14, no. 2, pp. 221-239, April 1982.

4. S. A. Evans, "Scaling I^2L for VLSI," *IEEE Journal of Solid-State Circuits*, vol. SC-14, no. 2, pp. 318-326, April 1979.

5. T. R. Gheewala, "Design of 2.5-Micrometer Josephson Current Injection Logic (CIL)," *IBM J. Res. Develop.*, vol. 24, no. 2, pp. 130-142, March 1980.

6. P. Guéret, A. Moser, and P. Wolf, "Investigations for a Josephson Computer Main Memory with Single-Flux-Quantum Cells," *IBM J. Res. Develop.*, vol. 24, no. 2, pp. 155-166, March 1980.

7. Paul A. H. Hart, Toon Van 'T Hof, and Francois M. Klassen, "Device Down Scaling and Expected Circuit Performance," *IEEE Journal of Solid-State Circuits*, vol. SC-14, no. 2, pp. 343-351, April 1979.

8. B. Hoeneisen and C. A. Mead, "Fundamental Limitations in Microelectronics - I. MOS Technology, II. Bipolar Technology," *Solid-State Electronics*, vol. 15, pp. 819-829, 891-897, 1972.

9. H. Ko and T. Van Duzer, "Miniaturization of Josephson Current Injection (CIL) Logic Circuits," in *International Conference on Computer Design, VLSI in Computers*, IEEE Computer Society, November 1983.

10. F. S. Lee, R. C. Eden, S. I. Long, B. M. Welch, and R. Zucca, "High speed LSI GaAs integrated circuits," in *Proc IEEE International Conference on Circuits and Computers*, pp. 697-700, Oct 1980.

11. Thomas Lengauer and Kurt Mehlhorn, "On the Complexity of VLSI Computations," in *VLSI Systems and Computations*, ed. H. T. Kung, Bob Sproull, Guy Steele, pp. 89-99, Computer Science Press, October 1981.

12. Tso-Kai Liu, "Synthesis of Multilevel Feed-Forward MOS Networks," *IEE Trans. Comput.*, vol. C-26, no. 6, pp. 581-588, June 1977.

13. Krishna C. Saraswat and Farrokh Mohammadi, "Effect of Scaling of Interconnections on the Time Delay of VLSI Circuits," *IEEE Journal of Solid-State Circuits*, vol. SC-17, no. 2, pp. 275-280, April 1982.

14. J. Savage, "Planar Circuit Complexity and the Performance of VLSI Algorithms," in *VLSI Systems and Computations*, ed. H. T. Kung, Bob Sproull, Guy Steele, pp. 61-68, Computer Science Press, October 1981.

15. C. D. Thompson, "A Complexity Theory for VLSI," Ph. D. Dissertation, CMU-CS-80-140, Computer Science Dept., Carnegie-Mellon University, August 1980.

16. Han-Tzong Yuan, Yung-Tao Lin, and Shang-Yi Chiang, "Properties of Interconnection on Silicon, Sapphire, and Semi-Insulating Gallium Arsenide Substrates," *IEEE Journal of Solid-State Circuits*, vol. SC-17, no. 2, pp. 269-274, April 1982.