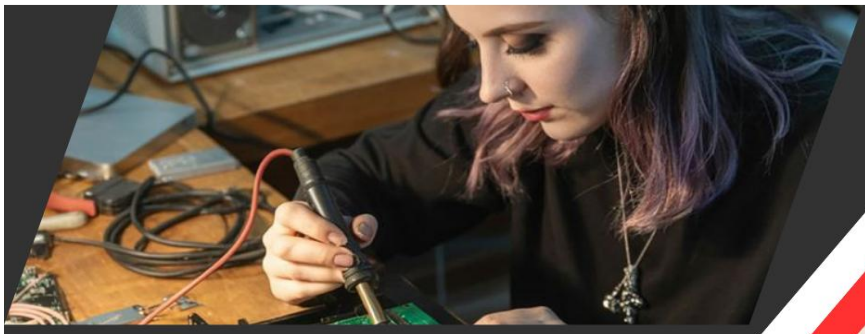
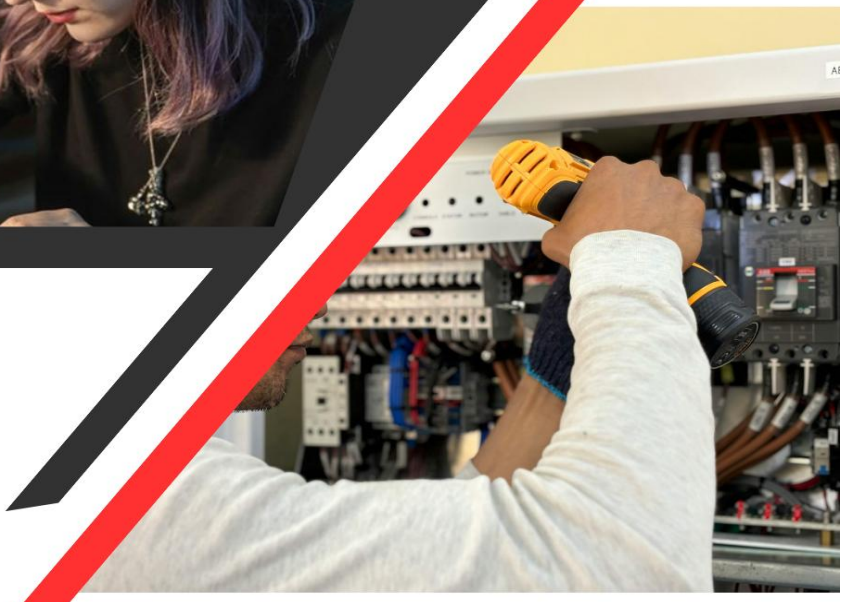
MULTISCIA
JOURNALS PUBLISHERS

FRONTIERS IN ELECTRONICS AND COMMUNICATION ENGINEERING

ISSN: (3065- 4106)





editor.fece@gmail.com
https://multisciajournals.com/journals/index.php/fece

Frontiers in Electronics and Communication Engineering
Volume 1 Issue 2 2025

Low-Power Digital CMOS Design

Armen Yuri Gasparyan*1 , Marlen Yessirkepov2
Department of ECE

Article Info

Received: 16-02-2025 *Revised:13-03-2025* *Accepted:23-03-2025* *Published:10-04-2025*

INTRODUCTION

Recent developments in VLSI fabrication have significantly raised integration levels, enabling the use of extremely complicated algorithms like discrete cosine transforms and Viterbi decoders, among others. Naturally, faster speeds have resulted from smaller integrated circuit device and feature sizes. Demand and expectations for the ongoing advancement of functionality and speed have increased. The communications sector, in particular, is growing quickly as every provider looks to improve system performance. However, since many items in this and other markets are portable and run on batteries, low power consumption is also necessary. Unfortunately, battery technology has not kept pace with VLSI's rapid progress. As a result, VLSI designers are being pushed to use new technologies in order to create complicated, high-throughput, low-power systems.

The fact that a component can only waste a certain amount of power without requiring extra cooling is another reason for low-power design [1]. These limitations are already being exceeded by high performance CMOS designs, and as a result, the costs of the final system are primarily determined by the cooling equipment's costs. By avoiding the requirement for this extra hardware, low-power integrated circuits (ICs) can provide a significant market advantage. Fortunately, low-power design is made possible by numerous features of the new integrated circuit technology. Faster devices and smaller capacitances are also provided by smaller feature sizes, which were created for more integration. Power consumption and speed are also greatly impacted by lower supply voltages, which are being developed to enable even greater integration. In order to make up for the device speed decrease brought on by lower supply voltages, the designer can utilize the extra space created by higher levels of integration.

All aspects of an integrated circuit's design, including fabrication technology, circuit optimization, logic design, control and clocking strategies, architectural partitioning and layout, and the algorithm underpinning the system, can have an impact on the circuit's overall power consumption [2-4]. This article will examine an integrated circuit's power consumption and how low-power techniques can be found at each of these levels.

SOURCES OF POWER CONSUMPTION

Digital CMOS has two different types of power consumption. While the second type is waste and results from short-circuit currents that flow straight from the power supply to ground, the first can be considered beneficial since it establishes information by charging and discharging signal lines.

Figure 1 shows the useful power dissipation for a basic CMOS inverter. The output capacitor can charge to the supply voltage when the input signal is low because the p-type transistor is ON and

the n-type is OFF. When the input is high, the n-type is switched ON and the p-type is turned off, and the output discharges to ground. Because it only happens when the output switches, the power dissipated in this way is referred to as dynamic power. Power is the product of voltage (V) and current, as we learned in school physics. In this instance, the current is the speed at which charge travels from the power rail to ground, and the voltage is the supply voltage (Vdd). By charging and discharging the output, the charge is transferred; if the output has capacitance C, the charge is (CVdd), and the current is that charge times the frequency of output switching. Dynamic power dissipation is therefore:

$$V_{dd} \cdot \sum_{i} (C_i \cdot V_{dd} \cdot A_i) \qquad (1)$$

where the summation is over each gate, and A_i is the *average* rate at which the output of gate i charges and discharges.

The power due to short-circuit current has the simple equation:

$$V_{dd} \cdot \sum_{i} I_{i\,sc} \qquad (2)$$

where the summation is over each gate and $I_{i\,sc}$ is the *average* short-circuit current flowing through gate i.

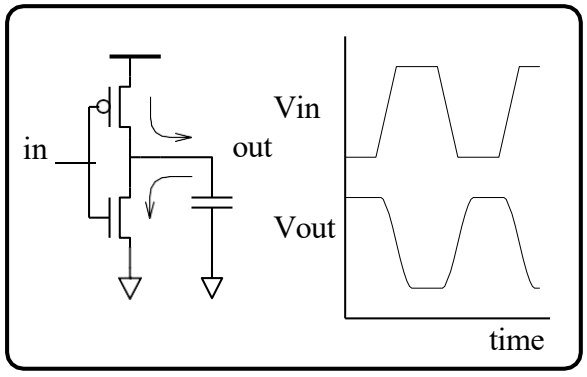


Figure 1: dynamic power dissipation

Thus to design components with low power consumption, we must consider how to reduce the values of V_{dd}, C, A and I_{sc}.

REDUCE V_{dd}

Reducing Vdd is the initial tactic, and as it shows up in equation 1 as a squared term, this seems to be the ideal place to start (albeit the relationship is complicated by the fact that A is also voltage dependant). It is obvious that the design engineer typically has little control on Vdd. Nonetheless, advancements in manufacturing are already shifting from the current 5V standard to a new level of 3.3V, and experimental procedures are considering even lower voltages. The problems with this development are worth taking into account [5].

Power supply reduction

One of the main motivations in technology development has been to increase the levels of integration by reducing feature sizes. However, as gate lengths are reduced (without reducing voltage levels) the electric field strength increases in the gate region. This leads to reliability problems as the high electric field strengths accelerate the conducting electrons to such speeds that they cause substrate current (by dislodging holes on impact in the drain area) and actually penetrate the gate oxide. The latter effect gradually alters the

characteristics of the

device leading eventually to latch-up and so to destruction. There are three approaches to enabling further feature size reduction. The first is drain engineering in which the doping profile is crafted in the channel region to reduce the degradation due to hot-electrons; the lightly doped drain (LDD) technique allows the smallest gate length[6]. The second approach is to use new circuit techniques which avoid the high electric fields across individual transistors. The third approach is to reduce the supply voltage; this solution is much the simplest for circuit designers but acceptance has been delayed as the industry wished to maintain compatibility with existing products.

The reduction in V_{dd} does not lead to a quadratic reduction in power as might be thought from equation 1 since some the other terms are dependent upon the supply voltage. To understand the actual effect, consider the activity level of each gate (A). This can be re-expressed as the product of the frequency (f) with which new inputs are presented to a whole circuit (for synchronous circuits, theVariation of the threshold voltage

From the standard transistor current equations, the speed of a circuit is a function of ($V_{gs} - V_T$) where V_{gs} is the gate-source voltage (limited by V_{dd}) and V_T is the threshold voltage. Thus it is also desirable to reduce the magnitude of the threshold voltages[7] either to minimise the reduction in speed or to allow further reduction in V_{dd} .

There are other reasons for reducing the threshold voltages. Rather than thinking about reducing one parameter at a time it is instructive to consider how to improve a technology as a whole. The work of Dennard[8] in 1974 promised that if the voltage levels (power and threshold) were scaled by the same amount as feature sizes then delay would be reduced by the same factor and power consumption (for the same circuit) by its square. This principle avoids the high electric fields which lead to the hot-electron effect because the voltage levels are reduced also; it is known as *constant electric-field scaling*. For example, a circuit designed in a technology of $V_{dd} = 5V$, $|V_T| = 1V$,
clocking frequency) and a probability for each node (pr_i) that it will change on any given cycle. The maximum possibleand gate length $= 1\mu$ could be re-implemented in one of $V_{dd} = 2.5V$,
$|V_T| = 0.5V$, and gate length = 0.5μ with
frequency of a circuit (f_{axm}) representstwice the speed and a quarter the area and power consumption.
the fastest throughput of data and this is
limited by its critical path or longest delay; thus f_{max} is inversely proportional to circuit delay. This brings us to a common measure of circuit quality: the power-delay product. By re-arranging equation 1 we have:Of course, all good things come to an end. One feature which does not scale is the roll-off rate of sub-threshold current as the following explains. In the weak inversion region (where V_{gs} is below V_T) there is no *drift* current; however, there is *diffusion* current which has the form:

$$power \,.\, delay = \frac{P}{f_{max} V_{dd}^2 \,.\, \sum (C_i \,.\, p(r3_i))_i}$$

$$I_{ds(sub-threshold)} \propto exp(V_{gs} - V_X) \qquad (4)$$

Thus variation in V_{dd} actually leads to a quadratic change in the power-delay product.where V_X is the lowest gate voltage for the weak inversion region. The important point is that this exponential roll-off rate is not effected by voltage scaling. For silicon it is in the region of 70-90mV per decade of current. Figure 2 shows how *threshold voltage* can be defined as the

junction between the axis and the linear current, as well as the roll-off of the sub-threshold current. The entire curve shifts to the left if the threshold voltage is lowered, which is accomplished by adjusting the concentrations of the substrate and channel dopant. As a result, there is a large short-circuit current and the device cannot be turned off correctly at low threshold voltages (where Vgs ⁉ 0V).

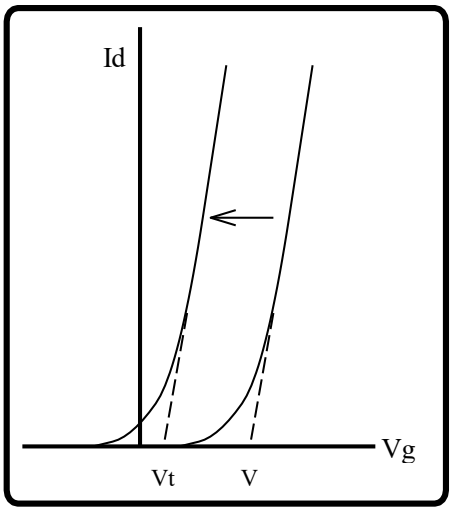


Figure 2: threshold voltage roll-off

For instance, in a "typical" 1.5 CMOS process with Vdd ⁉ 3V, a minimum sized gate has Ids ⁉ 30 A at Vgs VT and a roll-off rate of roughly 80 mV/dec. Therefore, subthreshold currents alone would cause a component of one million such devices to consume roughly 1W of power if the threshold voltage was set at 0.15V. This implies that a practical minimum of roughly half a volt must be maintained in order to prevent significant power consumption caused by sub-threshold currents at Vgs = 0V.
Another viewpoint is that Vdd could be further decreased by bringing VT down to lower values. Because of the consequent decrease in dynamic power, it might be feasible to tolerate a certain amount of short-circuit current.

consumption. This has yet to be demonstrated.

Optimal power voltage

The hot-electron effect establishes an upper limit on power supply voltage due to reliability criteria but, as suggested above, the low-power designer would prefer a lower limit. Here are two suggestions to be applied to any given technology.

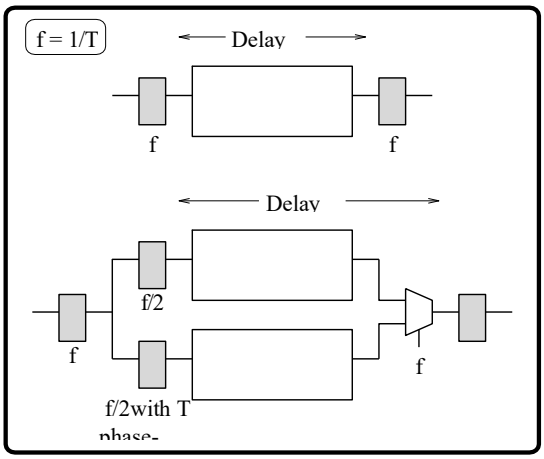
To optimise the power-delay product, it has been found that the optimal power supply voltage for a given technology is three times its threshold voltage[3, 9]. This seems intuitively reasonable in that it allows one threshold for each device type and one extra for noise margin. To avoid significant sub-threshold current, the minimum V_T should be at least 0.5V, giving an optimal power supply voltage of 1.5V.

A second approach considers a phenomenon, known as *velocity saturation*, in which the velocity of the charge carriers reaches a maximum with increasing electric field strengths. In other words, an increase in voltage does not increase the current and so does not improve the device speed. This then sets a limit above which it is unproductive to raise the supply voltage and this limit depends upon the fabrication technology and the effective channel length[5].

Compensating for lower speed

As the industry moves from the standard 5V processes to ones with lower supply voltages, design engineers need to compensate for the loss of performance if they wish to achieve the same throughput. There are two architectural approaches: first, apply the standard speed optimisation techniques only more so; second, use parallelism.

Pipelining is a standard technique for increasing the overall speed of a circuit involves adding clocked latches to combinatorial logic sequences so that a clock signal controls the data flow. The frequency (f), which is the inverse of the largest delay between any two adjacent latches, can then be used to process data. Usually, a designer will add as many latches as required to lower the critical-path delay (T) to a level that permits the specified frequency. The circuit speed will decrease and the critical-path latency will grow if the target supply voltage is subsequently decreased. A designer would then need to add or rearrange latches to make up for this and restore the proper frequency. Figure 3 shows how a designer can sustain throughput by adding additional latches in the middle of a delay channel in a circuit operating at half its initial speed. This would only be feasible, of course, if the original circuit hadn't already been "engineered" to operate at its best.



is presented to one block while the previous data is still being processed by the other. The outputs of the two blocks are selected by a multiplexor so that the valid data is latched at the original frequency. Notice that although the total capacitance of the circuit has been (approximately) doubled, the term *A* (in equation 1) has been halved because of the speed reduction: these two effects compensate for each other in the dynamic power equation.

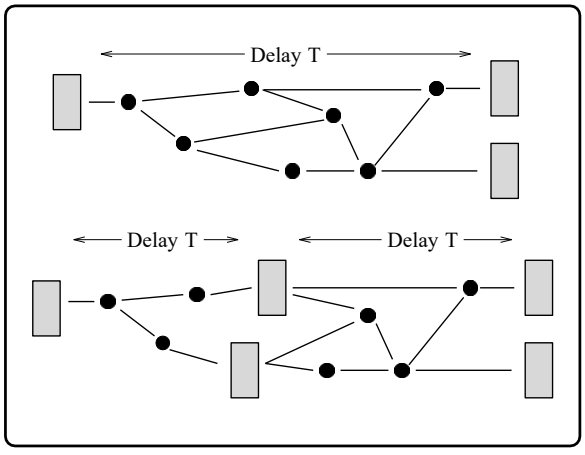


Figure 4: parallelism

Figure 3: pipelining

The idea of using parallelism is simply to have more operations being conducted at the slower speed to achieve the same overall performance. This is essentially a trade-off between circuit area and throughput. The use of parallelism is illustrated in figure 4. Here we assume that the critical path delay (T) through the combinatorial logic block has (nearly) doubled due to a reduction in the power supply voltage. To achieve the sameOf course, this strategy may sound attractive in the context of rapidly increasing levels of integration, but in terms of commercial viability it must be remembered that doubling the circuit area can have a large impact upon component cost. While many design specifications may demand this approach for the resulting speed, many will also preclude it on the grounds of cost.

In both of these example, the design has been modified to compensate for a halving in circuit speed resulting from a reduction in power supply. To illustrate with a very rough calculation, assume that this is done with a standard 5V process without changing any other fabrication parameters. If speed is taken to be proportionate to (Vdd – VT) and VT ⁋ 1V, then cutting Vdd from 5 ° 3V results in a speed reduction of half. Ignoring the comparatively minor variations in Ci Ai brought about by the additional circuitry, the i

The dynamic power consumption is lowered to its initial value of (3/5)2 ⁋ 0.36. The supply voltage could have been lowered to 2.5V and the dynamic power to just 0.25 of its initial value if the threshold voltage had also been lowered to, say, 0.5V.

Voltage swing

A final way of reducing power loss connected with the supply voltage is to re-examine equation 1. The second V_{dd} term actually refers to the voltage swing of the internal nodes. If this were reduced, then the total power consumption would also be reduced. One example of this concerns an internal bus architecture[10] which is designed for operation at about 2V with an internally generated supply for the bus itself. Modified thresholds, and special driving and sensing circuitry, allow the bus to swing less that 1V. This not only saves power in itself, but also increases the bus speed making operation at 2V more attractive.

REDUCE *C*

The second strategy is to reduce capacitance. This comes naturally with smaller feature sizes and so a circuit designer will generally wish to use the minimum geometries possible in the given technology.

Partition blocks

As a general rule, it is best to partition large blocks into smaller ones. The design on the left in Figure 5 is a large memory block: the shaded area is the address generation and bit detection circuitry, and the unshaded region is thememory array. The power calculation for

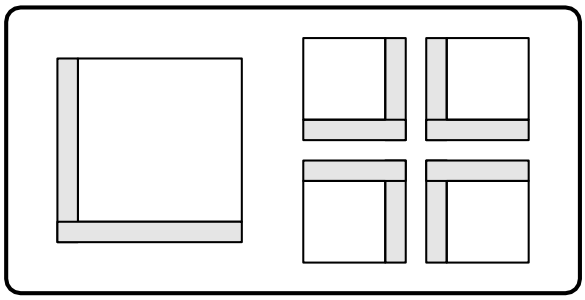


Figure 5: block partionin

each memory access is based upon the capacitances of the bit and word lines which run vertically and horizontally across the whole array. If, instead, the array is broken down into four sub-units (each with its own support circuitry) and only one unit is addressed with each access, then the product of activity and capacitance is reduce by a half.

Locality of reference

There is another architectural strategy which can significantly reduce the capacitance (or more specifically $\sum_i C_i A_i$)

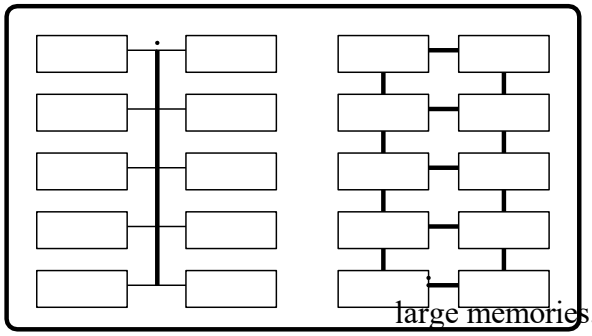
of a design; it can be summarised in the phrase *locality of reference*. This is a design philosophy in which signals are generated and used locally in terms of their physical location on the silicon surface since the further a signal has to travel, the higher is the capacitance of that connection. With signals being processed locally, there is greater opportunity for parallel execution. With parallel execution, there is greater throughput which could be traded-off for a lower supply voltage and so lower power consumption.

Designing with locality of reference is desirable for another reason related to the new fabrication technologies. Communication within a component is achieved using metal interconnect. For large feature sizes, the RC delay on such lines is relatively small compared with the transistor delays in the circuit. However, while transistor delays scale Since the resistance rises as the interconnect lines are smaller and the fringing capacitance starts to dominate the total capacitance (and does not scale), the RC delays actually increase as feature size decreases. As a result, communication delays are common in sub-micron technologies [11]. Thus, the main possible cause of delay is avoided by employing location of reference as a design style.

Architectural strategies based upon this idea may include: processing of data locally to where it is stored, communication only with physically adjacent functional units, and dedication point-to-point buses rather than shared ones. Figure 6 illustrates this idea. The architecture on the left consists of a large number of units connected by a global shared bus; this is not uncommon. Consider the communications bus alone. The architecture on the right has fourteen much smaller buses which together have roughly the same total capacitance as the single bus on the left. If only one was active, then the activity-capacitance product would be 1/14th; if they were all active, then the power consumption would be the same but up to 14 times as much information could be transferred.

Figure 6: locality of reference

It is clear that low-power systems will include the use of dedicated function- specific circuits which transform particular sections of the total algorithm onto localized areas of silicon − they will not have central processing units communicating by global buses with



large memories.

Clocks and control

In architectures with distributed processing, the question arises as to whether there should be global control and clock signals. On the one hand, there needs to be synchronisation between communicating pairs of processors; on the other hand, the global distribution network has a very large capacitance and is switched frequently. There are several possible strategies.

A new latch circuit was introduced in 1987 which allows *true single phase clocking*[12, 13] (TSPC). This implies that only one clock signal needs to be distributed where-as previously designs had relied upon having at least the complement of the clock available (either distributed in parallel or generated locally). Thus by using TSPC, a design can greatly reduce the capacitance of its largest network. TSPC has already been used to implement extremely fast and power-hungry designs (e.g. the DEC Alpha) but, as we have seen above, the speed advantage could be traded-off against power by designing for lower supply voltages.

If the problem is the widely distributed clock signal, then one solution is not to distribute it so widely. In this approach, regions of the component which are not being used have their portion of the clock network gated off. The draw back to this scheme is the need to generate and distribute the clock control signals and the added design complexity in providing a synchronous clock signal on a network which is partitioned by function rather than by equal capacitance. A variation of this approach is to disable sections of the power (rather than the clock) distribution network.

Self-timed circuits dispense with a global clock altogether. In this scheme, units individually process, store, and produce a "ready" signal for their data. A hand-shaking protocol is used to facilitate communication between units. The following unit receives the preceding one's "ready" signal, stores the data locally, and then generates a "acknowledge" signal. If additional data is available, the first unit is then free to absorb it. Without a global clock, data is processed by local units and transferred throughout the component in this way. In actuality, the power dissipated in the ready and acknowledge signals can be of the same order, even if this appears to eliminate the global clock line's high capacitance-activity. Self-timed logic isn't always better than clocked logic, thus low-power designers shouldn't presume that.

Logic design

Using logic families with low capacitance is a further strategy. Complementary pass-transistor logic [14] (CPL) is one potential family. This creates logic functions using networks of just n-type pass-transistors (no p-types). The logic functions' outputs power CMOS inverters, and all signals are produced in complementary values. The sum function's CPL implementation, shown

in Figure 7, uses 12 transistors rather of the 22 required for a traditional implementation. The lower capacitance and fewer gates are the primary sources of the power benefit. This method has been successfully used to create a 16x16-bit multiplier in a 4V, 0.5□ CMOS process. Nonetheless, a circuit designer should be aware of certain aspects of this technique. The output high logic level is VTn below the power supply as a result of the pass transistor network's threshold voltage drop. This indicates that the inverters' p-type transistors are

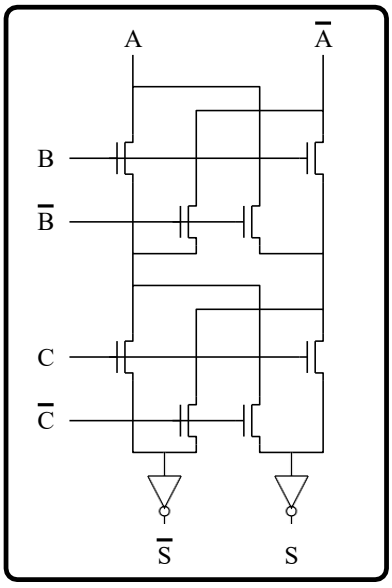


Figure 7: sum function in CPL

only switched off by ($|V_{Tp}| - |V_{Tn}|$) leading to sub-threshold currents as described in the previous section. This can be reduced by using cross-coupled p- type pull-up transistors on the complementary logic outputs (leading to increased transistor count and reduced speed) or by using a special fabrication technology with a lower threshold voltage for the pass-transistor n-types only (0V was used in 16x16-b multiplier). Thus the best results using this circuit technique for low-power depend upon also matching the fabrication process to it.

Two further design styles related to CPL have also been reported. One overcomes the problem of the threshold voltage drop by using full CMOS pass transistors[15]. This still has better speed performance than conventional CMOS and so would achieve a given throughput at a lower voltage (and so power dissipation). The second related design style uses threshold adjustment on the p- as well as the n-type transistors[16].

Buffer design

One recurrent problem is the design of circuitry to drive a relatively large capacitance (particularly external loads). The basic solution is a sequence of buffers with increasing gate widths; the design issue is what should be the size ratio (α) of each successive buffer.

With speed as the main consideration, the classical answer[17] is $\alpha = e$. With intrinsic output

capacitance of the CMOS buffer included this value is known to be layout dependent[18] and in the region of 5-6. However, if power is the main issue and the overhead in charging and discharging intermediate nodes is considered, the optimal ratio is layout and process dependent[19] and is about 11-12. The following table shows the example of different ratios for a buffer chain driving an 11pF load from an initial buffer with input capacitance of 0.1pF; "useful power" is that expended in charging the 11pF capacitance itself, and "other power" is that expended on intermediate nodes.

ratio (α)	*e*	11.5
# inverters	5	2
useful power	2.5mW	2.5mW
other power	5.4mW	1.5mW
total power	8.1mW	4.0mW
delay	5.5nS	6.5nS

Thus there is over a 50% reduction in power dissipation, and a similar reduction in layout area, at the price of only an 18% increase in delay.

REDUCE *A*

The third strategy is to reduce *A*: the average activity on each gate. Power is only expended when a node is switched; if switching is to restricted to when information changes then power is minimised. This can be summarized by the phrase *transition avoidance*. As a first observation, this argues against the use of circuit styles which involve precharging and discharging as part of logic evaluation.

Glitch avoidance

With some digital logic, there are spurious transitions (known as *glitches*) which occur due to partially resolved functions; figure 8 shows an example. If there is a unit delay through both of these gates then when the inputs both change from 1 to 0, the output will change to a 1 as the logic is resolving before returning to a a final value of 0. This wastes power. The problem is reduced in general by designing circuits so that there are equal delay paths between all of the gate inputs and the system inputs, thus equalising arrival times of changing signals. Of course, this is hard to achieve in practice and impossible if there is feedback in the circuit.

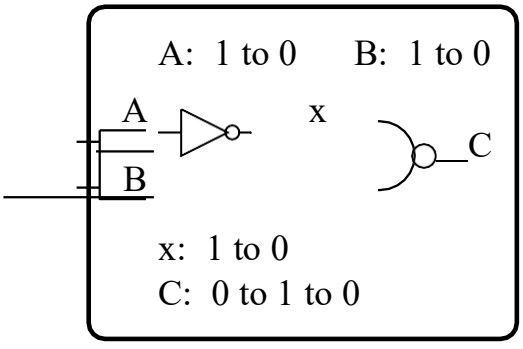


Figure 8: glitches

A more important example of power loss through spurious transitions is the ripple adder. In this logic design, each bit-adder unit passes its carry to the next unit in a carry-chain; the

value of its own input is not, however, valid until all the less significant bits have been resolved; thus each carry-bit in the chain may change (along with the corresponding sum outputs) as the valid carry signal propagates along the chain. To avoid the associated loss of power, a different adder design should be used.

Point-to-point buses

This concept of transition avoidance can be viewed at the architectural level also. Suppose there are two independent slowly-varying digital signals within a component. If these are distributed on independent data buses, then transitions only occur when information changes. If, instead, the two signals are combined by a multiplexor onto

a single bus for distribution, there is also likely to be a transition when the multiplexor is switched (i.e. when the control signal changes). Although point-to-point buses incur an area cost due to the extra interconnect routing, they save significant power by avoiding transitions which occur when mixing independent signals.

Reviewing the algorithm

The power consumption of a complex system can be greatly influenced at the algorithmic level. Normally, component power consumption corresponds to the usual algorithmic performance criterion of speed since algorithmic speed is a function of the number of operations and this translates onto the component as the amount of switching. Thus the programmer's desire to reduce the number of steps in a computation will naturally reduce the power consumption of its implementation.

The idea of locality of reference may be mapped directly into algorithmic design through the use of certain programming languages. Concurrent object-orientated languages (e.g. ADA and VHDL) allow the creation of software modules which mostly use locally declared memory and which allow communication between modules as an alternative to parameter passing by function call. It could be most effective for low-power system design if the underlying algorithms were developed in such a language from the very eginning.

REDUCE I_{sc}

A designer needs to consider short-circuit current in two ways: first, how to minimise what is unavoidable, second how to avoid what is unnecessary.

Resistive networks

Firstly, some logic styles deliberately use resistive networks formed from transistors to establish the value of the output signal (e.g. pseudo-NMOS). These styles cannot be used for low- power design. Secondly, some strategies for avoiding power loss involve generating multiple voltage levels using resistive networks either on-chip or at the system level. This static power loss must be carefully included in the evaluation of such strategies.

Switching current

However, even conventional static CMOS has a source of short-circuit currents. Consider figure 9. As the input to a CMOS inverter changes, there is a period during which both transistors are switched ON that is when the input voltage is between ($V_{dd} - V_{Tp}$) and V_{Tn}. During this period, there is a short-circuit current and so power dissipation. This is clearly dependent upon the rise time of the input signal. For poorly designed circuits, this power loss can be about 20% of the total power dissipation. A simple rule-of-thumb for designers

is to size the transistors so that the delay in the output signal is the same as that of the input; with this strategy, the short-circuit power loss is reduced to 1-2% of the dynamic power dissipation[19].

Glitch propagation

The example concerning spurious transitions in figure 8 above was explained in terms of unit time delays. In fact the problem is compounded in that the output glitch propagates on to other

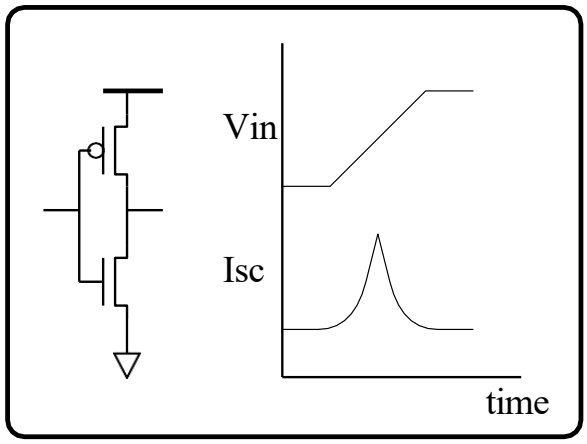


*Figure 9: short-circuit current on switching*tages.

In practice this signal often takes the form of a slowly varying voltage which hovers in the centre of its range causing short-circuit currents in the next gate. This is another source of power dissipation and a further reason to avoid logic glitches.

APPROACHES TO LOW-POWER DESIGN

The first decision the IC designer needs to make is the choice of fabrication technology. Low supply voltage is good. Small feature size is very good because of the low capacitance and increased device speed due to the short channel length. Fortunately, the first is tending to follow the second because of the hot- electron effect. The simple rule-of- thumb is to use the process with the smallest feature size that the project can afford.

If there is access to a fabrication technology with multiple (and specifiable) threshold voltages, then this might be chosen to support the CPL design style or one of its derivatives.

The next decision is the design partitioning. A designer should avoid architectures based on central processing units, and always review the specified function. The aim should be to partition the function into small independent units (avoiding high capacitance interconnects)operating in parallel (raising throughput). The method is to apply the principle of locality of reference even if this means returning to the algorithmic development level.

The final decision must be to re-evaluate the standard techniques for logic and circuit design. Designing for low power means that many of the old stand-bys are redundant and that new approaches must be developed. It requires a strong but subtle change in emphasis.

For instance, in one sense the designer must abandon the imperative for speed which in itself leads to high power consumption. On the other hand, if a reduced voltage level is the main mechanism for power reduction, then all the old tricks for enhancing speed may be needed to compensate for the reduced drive capability.

A second change in emphasis is that area is no longer as limited a resource as it used to be. Thus, with low power as the main criterion, techniques which require extra area are not unattractive. In particular, resource sharing is less important particularly when it leads to additional switching.

The fascinating opportunity is that since power has become the main design cost, the designer can now explore radical options in algorithmic, architectural, logic and circuit design. The challenge is here, the fun is just beginning.

Dr Gerard M Blair is a lecturer in VLSI Design in The Department of Electrical Engineering, The University of Edinburgh, The King's Buildings, Edinburgh EN9 3JL; email: gerard@ee.ed.ac.uk

References

1. Minoru Nagata, "Circuits and Devices into a Half Micrometer and Beyond: Limitations, Innovations, and Challenges," IEEE Journal of Solid-State Circuits, vol. 27, no. 4, pp. 465-472, April 1992.
2. "Low-Power CMOS Digital Design" by Anathna P. Chandrakasan, Samuel Sheng, and Robert W. Brodersen, IEEE Journal of Solid-State Circuits, vol. 27, no. 4, pp. 473-484, April 1992.
3. "Low-Voltage ULSI Design" by Katsuki Shimohigashi and Koichi Seki, IEEE Journal of Solid-State Circuits, vol. SC-28, no. 4, pp. 408-413, April 1993.
4. Richard F. Lyon, "Cost, Power, and Parallelism in Speech Signal Processing," IEEE Custom integrated Circuits Conference, pp. 15.1.1–15.1.9, 1993.
5. "Power-Supply Voltage Impact on Circuit Performance for Half and Lower Submicron CMOS LSI," by Masakazu Kakumu and Masaaki Kinugawa, IEEE Transactions on Electron Devices, vol. 37, no. 8, pp. 1902-1908, August 1990.
6. "Comparison of Drain Structures in n-Channel MOSFETs" by Hiroaki Mikoshiba, Tadahiko Horiuchi, and Kuniyuki Hamano, IEEE Transactions on Electron Devices, vol. ED-33, no. 1, pp. 140-144, January 1986.
7. "Trading Speed for Low Power by Choice of Supply and Threshold Voltages," by Dake Liu and Christer Svensson, IEEE Journal of Solid-State Circuits, vol. 28, no. 1, pp. 10-17, January 1993.
8. "Design of Ion-Implanted MOSFETs with Very Small Physical Dimensions" by Robert H. Dennard, Fritz H. Gaensslen, Hwa-Nien Yu, V. Leo Rideout, Ernest Bas- sous, and Andre R. LeBlanc was published in the IEEE Journal of Solid-State Circuits in October 1974 (Vol. SC-9, no. 5, pp. 256-267).
9. J. Burr and A. Peterson, "Ultra Low Power CMOS Technology," 3rd NASA Symposium on VLSI Design, October 1991, pp. 4.2.1-4.2.13.
10. Yoshinobu Nakagome, Kiyoo Itoh, Masanori Isoda, Kan Takeuchi, and Masakazu Aoki, "Sub-1-V Swing Internal Bus Architecture for Future Low-Power ULSIs," IEEE Journal of Solid-State Circuits, vol. 28, no. 4, pp. 414-419, April 1993.
11. Arnold Reisman, "Device, Circuit, and Technology Scaling to Micron and Submicron Dimensions," IEEE Proceedings, vol. 71, no. 5, pp. 550-565, May 1983.
12. "A true single phase clock dynamic CMOS circuit technique" by Y Ji-ren, I Karlsson, and C Svensson was published in the IEEE Journal of Solid-State Circuits in 1987 (Vol. SC-22, pp. 899–901).
13. "A Unified Single-Phase Clocking Scheme for VLSI Systems" by Morteza Afghahi and Christer Svensson, IEEE Journal of Solid-State Circuits, vol. SC-25, pp. 225-233, February 1990.
14. "A 3.8-ns CMOS 16x16-b Multiplier Using Complementary Pass-Transistor Logic" by Kazuo Yano, Toshiaki Yamanaka, Tashiaki Nishida, Masayoshi Saito, Katsuhiro Shimohigashi, and Akihiro Shimizu was published in the IEEE Journal of Solid-State Circuits in April 1990 (Vol. 25, no. 2, pp. 388-395).

1. Makoto Suzuki, Norio Ohkubo, Toshinobu Shinbo, Toshiaki Yamanaka, Akihiro Shimizu, Katsuro Sasaki, and Yoshinobu Nakagome, ‘‘A 1.5-ns 332-m CMOS ALU in Double Pass-Transistor Logic’’, *IEEE Journal of Solid-State Circuits*, vol. SC-28, no. 11, pp. 1145-1151, Nov 1993.

2. Menahem Lowy, Chi-Yuan Chin, and Jerome J Tiemann, ‘‘Low Power Consumption

Communication Systems'', *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, pp. 807-810, May 1991.

3. Carver Mead and Lynn Conway, *Introduction to VLSI systems,* Addison-Wesley, 1980. ISBN 0-201-04358
4. Nils Hedenstierna and Kjell O Jeppson, ''CMOS Circuit Speed and Buffer Opti- mization'', *IEEE Transactions on Computer-Aided Design*, vol. CAD-6, no. 2, pp. 270-281, March 1987.
5. Harry J M Veendrick, ''Short-Circuit Dissipation of Static CMOS Circuitry and its Impact on the Design of Buffer Circuits'', *IEEE Journal of Solid-State Circuits*, vol. SC-19, no. 4, pp. 468-473, Aug 1984.