IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

A Survey on Low Power Memory Design Techniques

Z. Mahesh Kumar[*1], R. Manjula[2]
[*1,2]VIT University, Vellore (T.N.), India
mahesh.cse349@gmail.com

Abstract

Low power is an important factor when designing chips as well as memories. That is driven by the increasing complexity and operating speeds of microprocessors and the demands of portable electronic equipment. Many techniques have been developed for getting low power. This term paper report includes a summary of conventional low power circuit design techniques, as well as a discussion on low power memory. Those discussed will be techniques for reducing power in memory, including intelligent and OS Controlled refresh in DRAMs, multi-divided arrays and power/performance ratios, and a survey of low power SRAM and DRAM. The paper will also discuss power requirements of microprocessors, as one aspect of IRAM is adding a microprocessor on-board DRAM.

Keywords: Power reduction, refresh in DRAMs, SRAM caches.

Introduction

One of the main aspects of power consumption is it puts an upper limit on the number of gates that can reliably integrated on a single package of any technology. As technology advanced, chips grew and it is possible to integrate more functions into one chip. Just as for TTL (Transistor-Transistor logic), newer technology called CMOS (Complementary Symmetry Metal Oxide Semiconductor) came to replace NMOS (N-type metal oxide semi conductor) because CMOS consumes less power.

Because of advances in technology and fabrication, the integration densities and the rate at which chips operate have increased drastically, it causes power consumption is important thing. In addition, the new requirements set by device portability, reliability and costs helped to reduce power consumption in CMOS circuits.

Comparing the low-end processor's performance to that of the 1Gb DRAM device we can observe that the processor consumes slightly more power than the DRAM array. The power considerations of a small processor (not aimed at high-performance computing) included on-chip with a DRAM device are not likely to be a critical issue. Any serious processing, on the level of a high-end Alpha, however, would easily dominate over the power of any on-chip DRAM. This is further evidence for Horowitz's prediction that IRAM might be best targeted for embedded applications market.

Intelligent RAM, or IRAM, merges processing and memory into a single chip to lower memory latency, increase memory bandwidth, and improve energy efficiency as well as to allow more flexible selection of memory size and organization. In addition, IRAM promises savings in power and board area.

The big power-sink of an IRAM type device, however, is likely to come from the interconnection between the DRAM and processor blocks. These lines would be needed to fill cache lines, feed vector units. The number of these lines, and the distance they may have to travel (halfway across the chip) make them a problematic power drain. The interconnect problem is fast becoming a concern for modern logic designers.

Methodology of Low Power Circuit Design

Importance of Low power CMOS circuit design:

The number of transistors that can be integrated on a single chip would grow exponentially with time. The increase in the number of transistors per package allowed more functions to be integrated and increase the total logic density of the chip. As the time passes the size of the process (micro meters) is also reducing. IC logic complexity increasing day to day.(Figure 1) Memory integration density also increasing every year.(Figure 2). Both figures show the number of transistors per chip is increasing.

According to the constant field scaling theory, power dissipation scales as k2 and power

density i.e power dissipated per unit area remains constant while speed increases as k.

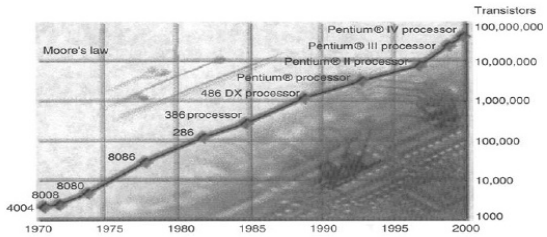


Fig 1 : IC Logic Complexity

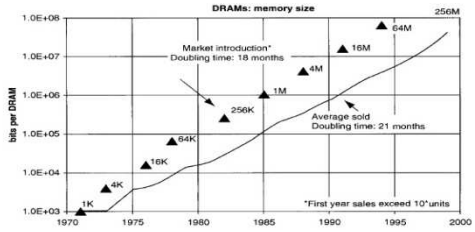


Fig 2 : Memory Integration density as a function of time

Sources of power consumption in CMOS:

Two types of power consumptions exist for digital CMOS. The first dynamic power component, useful one because it establishes information by charging and discharging signal lines. The second type consists of short circuit and static power components, is waste and comes from short circuit and leakage currents that flow directly from the power supply to the ground.

1.Dynamic Power Dissipation:

The dynamic power dissipation is the power required for the circuit to perform its anticipated tasks, simply power needed for charging and discharging all nodes in a CMOS circuit. The power is only consumed when the circuit input signals change. In CMOS circuits dynamic power dominates the total power dissipation. Such characteristic is greatly affected by

current process or the deep sub-micron processes (DSM). When the input signal falls PMOS transistor switches on and the NMOS transistor switches off. When the input signal rises NMOS transistor switches on and the PMOS transistor switches off.

If C_L is the total capacitance charged per cycle, then the dynamic power dissipation is as follows:

$$Pdynamic = CL * Vdd * Vdd * f * \alpha$$

V_{dd} is the Supply voltage level, f is the frequency of operation, and α is the switching activity of the capacitive node C_L on each clock cycle.

2. Short Circuit power dissipation :

The dynamic power dissipation equation is derived usually by assuming that the inputs have zero size and fall times. But in reality such assumption is not valid and input signals have non-zero rise and fall times. Hence a direct current path exists between V_{dd} and ground for a short period of time during input switching. NMOS and PMOS both never on simultaneously.

Short circuit power dissipation is as follows.

$$Pshort - circuit = Isc * Vdd$$

3. Static power dissipation :

Ideally the static power consumption of static CMOS circuits is assumed to be zero. (PMOS and NMOS devices are never on simultaneously steady state operation). Leakage currents come from the variety of effects in the transistor.

Weak inversion current (sub-threshold current) is carried through the channel when the gate voltage is below the threshold and it flows between source and drain of a MOS transistor. It increases exponentially with the reduction of device threshold voltage, making it critical for low voltage and low voltage circuit design.

I_{static} is the total current leaking in the steady state, static power dissipation is as follows

$P_{static}=I_{static}*V_{dd}$

Thus the total amount of power consumed by CMOS circuit is

$$\begin{aligned} Ptotal &= Pdynamic + P(short - circuit) \\ &\quad + Pstatic \\ &= (CL * Vdd * Vdd * f * \alpha) \\ &\quad +(Isc * Vdd) + (Istatic * Vdd) \end{aligned}$$

Techniques for Power Reduction

Power is the number of joules (energy) dissipated over a certain amount of time.

The average power consumed by a circuit is defined as :

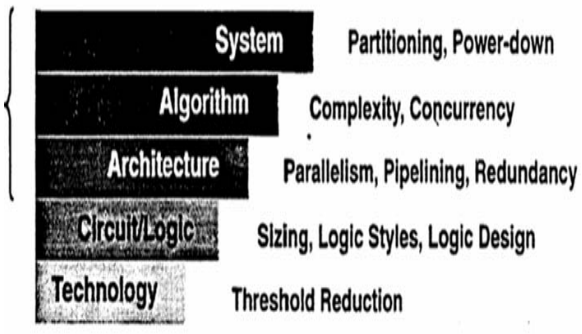
$$P_{av} = \frac{1}{T}\int_0^T P(t)dt = \frac{V_{\text{Supply}}}{T}\int_0^T i(t)dt = I_{av} \times V_{\text{Supply}},$$

Where P is average power, I is average current and $Vsupply$ is the supply voltage.

In general, power minimization targets maximum instantaneous power or average power. Power problem can be faced in two perspectives, design and technology. From a design perspective, circuit designers can choose from a number of options to reduce the power dissipation ranging from high levels of abstraction (architectural level) to lowest levels of abstraction (physical or technology level) . In each level of abstraction there exist a

number of techniques that can be employed to effectively reduce power dissipation of CMOS circuits.

The stages during the design involve system, architectural, logic, circuit, physical levels.



System level

At the highest level of abstraction, system is viewed as an integrated entity that consists of a hardware infrastructure executing software programs. Addressing the power dissipation at this level at this level has the greatest influence on power dissipation, as much as 400% power savings could be achieved.

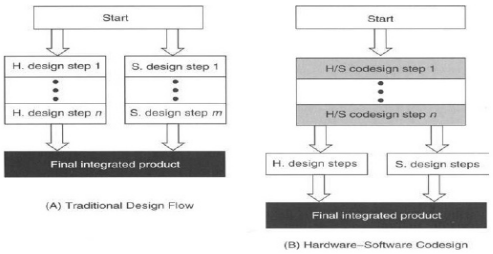
1.Instruction level optimization: In this power optimization is achieved by selecting a minimum power instruction mix for executing application software.

2. Hardware software co-design: Traditionally system is divided into hardware and software sections that are designed independently except for some common standards required for compatibility concerns. With systems growing larger and power consumption becoming of great importance, consider whole system design process and partition the various tasks of the system between hardware and software.

3. Memory design techniques: These techniques mainly focused on reducing the power consumed by memory. Increasing memory blocks on chip can reduce overall power consumption of a processor because the power dissipation of an external memory access is an order of magnitude higher than that of an on-chip access.

4. Dynamic power management: These techniques allow systems or system blocks placed in low power sleep mode when systems are in-active. Shutdown in-active blocks to reduce power consumption.

5. Variable voltage techniques: This is the most effective way of reducing power consumption. This is to lower the supply voltage with variable voltage techniques. With the availability of limitless number



of transistors, parallelism can be employed at the chip level, providing significant power savings by reducing the supply voltage level.

Architectural level

In this level, digital system consists of the structural view of the datapath and the logical view of the control unit of a circuit.

1.Parallelism and pipelining exploitation: One effective technique to reduce power consumption is to reduce supply voltage level by employing pipelining and parallelism which has more effect in reducing the overall power consumption.

2.Block disabling technique and clock gating: One can disable the blocks that are not in use for some particular clock cycles with the objective of limiting the power consumption.

3.InterCommunication and interconnect optimization: Some of the methods used to reduce the power dissipation in communication networks inside a chip include data encoding techniques and low swing signaling. In data encoding or compression method, reducing power dissipation by reducing the switching activities on switching buses.

Logic gate level

In this low power design is to target a library in which the components are designed to be low power.

1. Constrained optimization: Power optimization can take the form of constrained optimization problem where performance degradation is acceptable to a given bound. Thus power minimization requires optimal exploitation of the slack on the performance metrics.

2. Path equalization, lower v_{dd}: The path other than critical path is faster and one can lower the supply voltage for these paths, until the delay becomes similar to critical path.

3.path equalization -resizing: For path equalization, the alternative to lowering the supply voltage is to use gate re-sizing. Sizing here again focuses on non-critical paths or the fast paths where the gates along these fast paths are downsized to reduce their input

capacitances. Reducing these capacitances yields reducing in power consumption.

4.Glitch avoidance and Local transformations: Path equalization techniques help in reducing the glitches (un equal paths to outputs), consequently resulting in less power dissipation. Other logic level power minimization techniques involve local transformations including refactoring, remapping, phase assignment and pin swapping.

Circuit level

Only local optimizations are possible at this level that is low power techniques are applied to primitive components that assume some specific input and output characteristics such as input rise and fall times and output load capacitance. The techniques at this level are library cell design, transistor sizing and circuit design style.

1.Library cell design: Most critical cells in a design are timing elements i.e flipflops and latches. Flipflop design on low power focuses on minimizing the clock load and reducing the internal power when the clock is toggled.

2.Transistor sizing: It is quite useful to design the components with different sizes for a wide range of gate loads. Transistor sizing at the circuit level complements design techniques at the gate level, where the logic level sizing can help the synthesis tool in selecting the component with optimum sizing for low power consumption.

3.Circuit design style: Majority of digital VLSI systems are designed using static CMOS (SCMOS) circuit style. The main reason for using SCMOS in designing VLSI systems are robustness (low sensitivity to noise) and low power with no static consumption.

Physical level

Techniques at this level are technology dependent and are produced at the fabrication stage. Scaling the MOS transistor feature size causes the power to scale down by the same scaling factor and the power density remains constant. In the joint supply voltage-threshold voltage reduction, the supply voltage can be reduced to decrease the power consumption.

By Summarizing low power CMOS circuit design techniques at a relatively high level, the power consumption reduces. The first techniques developed are mostly common sense design practices such as lowering the power-supply to the chip rather than having a 5 volt supply and internal voltage regulation. The main techniques are voltage scaling, transistor sizing, and adiabatic circuits, as well as technology scaling, transition reduction, and parallelism.

1. Voltage Scaling :

Voltage scaling is the easiest and most effective way of controlling power. Adjustments to the operating voltage affect the delay in a linear manner, while having a quadratic effect on the power. The most common technique is to architecturally increase the performance of a system, and then lower the voltage for a reduction in the power consumption.

2. Transistor Sizing:

This technique directly trades speed for power. Decreasing the size of the transistor lowers power requirements and decreases the current drive, thereby making the gates slower. Increasing the size of non-critical path transistors can decrease power while not affecting the delay. This is difficult to implement in synthesis tools, however, is not a widely-used technique.

3.Adiabatic Circuits:

Adiabatic circuits are also known as charge recovery circuits. They resonate the load capacitance with an inductor in order to recover some of the energy used to change the capacitor's voltage. This is not a widely used technique because it introduces substantial delay.

4.Technology Scaling:

Technology shrinks cause the capacitance of nets to decrease. This decrease in capacitance results in not only the performance of a design increasing, but also in a reduction of the power requirements. This is not a good technique with the passage of time. While maintaining constant performance, the power dissipation of a circuit is related to x^4 where x is the ratio of the process shrink.

5. Transistor reduction:

In static CMOS design, a transition on a bit line is the fundamental event that uses power. Gating clocks to functional blocks is one common and effective method for reducing unnecessary switching. It is also theoretically possible to synthesize circuits so as to reduce the number of spurious transitions, but this is difficult and hard to achieve in practice.

6.Parallelism:

Parallelism can be used in a system to increase overall performance. The voltage of the system can then be reduced, lowering the performance to original levels, and lowering the power consumed even further. There is an overhead incurred with adding parallelism (control, inefficiency) so this is not always a win-win situation. (For example, the overhead of super-scalar operation makes it poor for power-reduction).

Techniques for Lower Power Using Refresh in DRAMs

The Motorola 4Mx1 low-power CMOS DRAM part has a 128ms refresh cycle. With a 110ns cycle time and 1024 banks, the device is required to be refreshing approximately 0.7% of the time. This percentage is small enough that it is likely to be overwhelmed by regular data accesses to the device. However, with the advent of the 1Gb DRAM device, the numbers change. The 1 GB DRAM from ISSCC has a 128ms refresh interval, requiring 1GB/sec of refresh bandwidth. This is equivalent to the peak internal 1GB/sec bandwidth for data transfer. This indicates that at peak theoretical operation, 50% of the power is going towards refresh. In periods of non-peak operation, the power consumed will be dominated by refresh. This ratio indicates that the alway-refresh line of thinking may not be ideal.

- **Intelligent Refresh**

For the DRAM cell, the refresh operation functionality is accomplished by a read or write operation. This means that if a cell has been recently read (or written to) then it does not need to be refreshed. This has the attraction of a more consistent power behavior. The device will take less power to refresh if accesses are being made to it, so this technique would be most effective during periods of great use. Implementation of this idea is probably not technically feasible because of the overhead needed to remember which lines have been recently accessed. Some clever algorithm, however, similar to a clock-paging algorithm (one bit used to appoximate LRU) may be applicable.

One simular situation where this concept might be useful is for systems with cache and DRAM on the same chip. If a word line is known to be in the cache, then it does not need to be refreshed. However, because of the relative size of on-chip caches and the size of DRAM, this technique is not likely to make much difference. (A 1Mb cache would only be able to 'prevent' .1% of the refreshes in a 1Gb DRAM.)

- **OS Controlled Refresh**

With memory sizes increasing and increasing (both system memory and single-chip memory), it is more and more likely that physical memory is not utilized at any given time. As it is not necessary to refresh unused memory, a considerable amount of power can be saved by intelligently controlling which pages get refreshed. The OS of a system knows which pages are used and unused, so given the opportunity it could disable refresh on selected pages.

Traditionally, a system only worries about swapping out pages when the memory space is full. Under a OS controlled refresh scheme, the OS could start to swap out pages to save power. The performance/benefit tradeoff of such actions is difficult to analyze because no current operating systems do this. This technique, however, would only help reduce the average power dissipation, not the maximum. This means that it can only be used for conserving battery life, but for not preventing a chip meltdown.

A Survey of Power Reduction in SRAM Caches

Techniques to Reduce Power in Wide Fast Memories

CMOS memories have an access path that can be examined in two parts: the address to the local word line select, and from the local word line to the sense amps. Driving the word line bus and sensing the data consumes the most power in this process.
Power consumption could be reduced by limiting the energy consumed by each bit line. This energy is conserved by limiting the swing of the bit line by controlling the local word line drive strength. This circuit technique adds an overhead of two extra columns and rows to implement a reference cell and reference bit line used in the drive strength regulation. The swing on the data lines is also limited.

One other optimization used was to only pre-charge selected blocks that were to be accessed, instead of pre-charging the whole array.

Supply Voltage (V)	Power (mW)	Gate Delay (ns)
1.5	5.2	2.63
3.0	75.0	0.62
5.0	66.0	0.38

6 ns 1.5 V 4 Mb BiCMOS SRAM

One of the problems in designing with Bipolar CMOS is that it is extremely difficult to scale. The fixed .8 V threshold voltage prevents scaling the voltage down as much as in other processes. So the reason for designing a BiCMOS process is for Speed. However, the speed benefit is not so much as to exclude consideration of other technologies. The speed-up as around a factor of 2, and BiCMOS designs usually require more area.
The 4 Mb BiCMOS SRAM presented at the 1996 ISSCC conference was a 1.5 V, 6 ns SRAM. This low-power SRAM was achieved using several low-power techniques.

1. Since the voltage is reduced, the speed benefit could be potentially lost. This can be fixed by using a boost voltage to accelerate the speed of the gates used in address decoding.

2. The standard method of reading and writing low power SRAM involved a word-boost technique on all cells in the array. This SRAM boosts only 1 of the 16,000 word lines.
3. The chip includes a stepped-down sense amp.
4. The chip includes an optimized boostvoltage generator.

Process: .3 um 4-poly 2-metal p-sub triple well BiCMOS
Supply: 1.5-3.3 V
Access: 6 ns
Power: 180 mW at 1.5 V

Conclusion

As far as implementing low power IRAMs, it looks like we should target using a smaller RISC microprocessor (possibly with a vector extension). This will prevent the microprocessor from dominating the power consumption in the IRAM. As far as reducing power in DRAM, we can sub-divide the memory array into blocks and share the row and column decoders. If we only activate the blocks we need, we can save power in this manner. We can also perform intelligent refreshes, such as refreshing only blocks that have been written to, instead of an entire array. There are also many circuit tricks that can be perpetrated on the DRAM or SRAM cores in order to optimize for low power. A main concern of power consumption in IRAM will be the interconnect situation. Application specific designs will further determine what sort of connection grid will be required.

References

[1] Bharadwaj Amrutur. "Techniques to Reduce Power in Fast Wide Memories" 1994 IEEE Symposium on Low Power Electronics. October 1994.
[2] Terry Biggs. "A 1 Watt 68040-Compatible Microprocessor."1994 IEEE Symposium on Low Power Electronics. October 1994.
[3] Tom Burd. "An interview with Tom Burd" Tom Burd's Ph.D. thesis is power efficient computing.
[4] Paul Gronowski. "A 433MHz 64b Quad-Issue RISC Microprocessor." 1996 IEEE ISSCC Digest of Technical Papers. Feb. 8-10, 1996.
[5] Mark Horowitz. "Low Power Digital Design." 1994 IEEE Symposium on Low Power Electronics. October 1994.
[6] Kiyoo Itoh. "Trends in Low-Power RAM Circuit Technologies."1994 IEEE Symposium on Low Power Electronics. October 1994.
[7] Shigeru Kuhara. "A 6 ns 1.5 V 4 Mb BiCMOS SRAM." 1996 IEEE ISSCC Digest of Technical Papers. Feb. 8-10, 1996.
[8] James Montanaro, et. al. "A 160Mhz 32b 0.5W CMOS RISC Microprocessor" 1996 IEEE ISSCC Digest of Technical Papers. Feb. 8-10, 1996.