EECS 151/251A Homework 9

Due Friday, April 19th, 2024

Introduction

This homework is meant to test your understanding of memories and energy/power. There are five total questions. Please check Ed first if you have any questions. Note for some questions you may need to consult with online resources, or other additional material beyond lectures.

Problem 1: SRAM vs. DRAM

Fill in the table below for the different types of memory used in digital circuits (for "Compositon" describe how the memory is created i.e. latches, capacitors, etc and for "Noise Resilience" rate as *Low*, *Medium*, or *High*):

	Registers	SRAM	DRAM	Hard Disk	Flash
Approx. Access Time (ns or s)					
Density (Mb/area)					
Volatility					
Approx. Cost/Bit ($)					
Power Usage (W)					
Composition					
Example Usage					

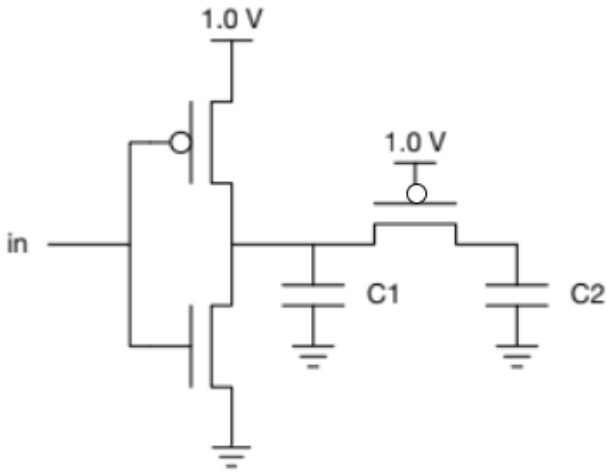
Solution:

	Registers	SRAM	DRAM	Hard Disk	Flash
Approx. Access Time (ns or s)	$< 1ns$	$> 8ns - 10s$	$> 60ns$	$5,000,000ns$ – $10,000,000ns$	$20,000ns$ – $90,000ns$
Density (Mb/area)	2.6 $Mb/\mu m^2$	$0.0199Mb/mm^2$	315 Mb/mm^2	1100000 Mb/in^2 (areal)	10000 Mb/in^2 (areal)
Volatility	Volatile	Volatile	Volatile	Non-Volatile	Non-Volatile
Approx. Cost/Bit ($)	0.8	$0.625	$0.00375	$.0000000000030625	0.00000874937
Power Usage (W)	nW	nW - mW	mW	mW - W	mW - W
Composition	D Flip-Flops	cross couple latch with access transistors, 6T, 5T, word line,2 bit lines	access transitor and capacitor, word line, single-bit line	Multiple rotating magnetic platters logically separated into sector read by magnetic heads	Transistor floating gate using tunneling. In NOR arch, transistors are in parallel; single word line and bit line. In NOR arch, transistors are in series; single word line and bit line.
Example Usage	Pipeline registers	caches	RAM in computers (desktop/laptops)	backing storage for large system	USB flash storage and space application

4

Problem 2: Energy

Consider the circuit shown below. Every transistor has the same effective resistance, R. The first node after the inverter has capacitance $C1$ and the node after the pass-transistor has capacitance $C2$. These capacitance values represent *all* the capacitance associated with those nodes. Initially, the gate terminal of the pass-transistor is connected to V_{DD}. The input to the inverter is a square wave signal with frequency f.



$$C1 = 90fF$$
$$C2 = 53fF$$
$$V_{th,n} = |V_{th,p}| = 0.32V$$

1. What is the dynamic power consumption of this circuit?
2. Now suppose we set the gate terminal of the pass-transistor to 0V. What now is the dynamic power consumption.
3. What can be done in principle to lower the power consumption of this circuit without decreasing f (reducing performance)?

Solution:

1. The capacitors are in parallel, but the access transistor is off, therefore $C_{total} = C1$. The input signal *in* switches twice in its period. We assume $\alpha = 1$. The energy expression below is for **a single cycle**.

$$E_{total} = \frac{1}{2}\alpha C_{total} V^2 f = \frac{1}{2}(2)(1)(C1)V_{DD}^2 f = 90 \text{ fJ}$$
$$P_{dynamic} = (90 \text{ fJ} \cdot f$$

2. The capacitors are in parallel and the access transistor is on, therefore $C_{total} = C1 + C2$. The pass-transistor is a PMOS which is a weak pull-down, therefore the voltage across C2 when *in* is high is $V_{th,p}$. The energy expressions below is for **a single cycle**.

$$E_{in,low} = \frac{1}{2}\alpha CV^2 = \frac{1}{2}(C1 + C2)V_{DD}^2 = 71.5 \text{ fJ}$$
$$E_{in,high} = \frac{1}{2}\alpha CV^2 f = \frac{1}{2}(C1)V_{DD}^2 + \frac{1}{2}(C2)(V_{DD} - |V_{th,p}|)^2 = 45fW + 12.2536fW = 57.2536 \text{ fJ}$$
$$P_{dynamic} = (E_{in,low} + E_{in,high})f = (128.7536 \text{ fJ})f$$

3. This question requires understanding of the terms in dynamic power. For our circuit:
   (a) The activity factor is solely determined by the input single *in*
   (b) The capacitance is fixed (the capacitors)
   (c) The frequency is solely determined by the input single *in*
   (d) Voltage is can be changed

   The term which is free to change is voltage which is good because $E \propto V^2$. However, reducing voltage to the inverter lowers drive strength and therefore the maximum frequency the circuit can run. To compensate for the reduced drive strength to replace the transistors in the inverter with multiple transistors in parallel (with shared drain) to bring the drive strength back up the original value.

Problem 3: Race to Halt

Background: *"Race to halt" is an effective energy efficiency strategy. In this scheme, a processor will run at it's maximum frequency to finish the required work as quickly as possible, then cut power to the circuit. Understandably, this strategy is effective when static power consumption is a dominant or significant component of total power consumption (although this specific strategy is implemented for CPUs, the general strategy can be applied to any circuit).*

Suppose you have a ML accelerator and you want to utilize it to run a parallelizable workload. The accelerator is composed of four matrix multiply sub-units. Functionally, all the sub-units are equivalent, capable of 9.5GFLOPs, however have different static power consumption. Sub-unit 0 is an efficiency unit which consumes 2W of static power while all the other sub-units consume 6W of static power. Regardless of the number of sub-units used, the accelerator consumes 2W of static power.

Additionally, there is a data partition unit which is by default configured to partition incoming data into blocks and send these blocks to individual sub-units (i.e. one block per sub-unit). The power consumed to partition is negligible. However, it costs 32J to reconfigure the data partition unit to not partition the data and send all incoming data to a single sub-unit. The interconnect which connects the data partition unit to the sub-units cost 0.25W per active sub-unit. Assume the interconnect is always on and consuming power regardless if data is being transmitted.

Your application requires an average of 200G floating point ops. Based upon estimates, it known that your application will consume 4W of dynamic power if ran on a single sub-unit.

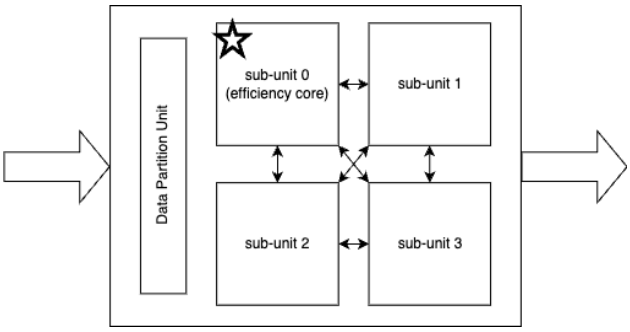


Figure 1: High-level block diagram of matrix multiplication accelerator

You would like to determine the most energy efficient way to run the application. You have the ability to control the supply voltage (V_{DD}), the clock frequency (f), and cut power to the accelerator when not in use. Assume that static power remains constant if voltage is scaled. Perform an energy analysis for three separate schemes (**assume voltage and frequency scale together**).

1. Voltage and clock frequency scaling running on single sub-unit
2. Voltage and clock frequency scaling by paralleling across multiple sub-units (**scaling factor = # of sub-units used*)
3. Implement a race-to-halt like energy savings scheme using a single sub-unit

Which approach is most energy efficient? Show your work and justify your answer.

Solution:

First, let's calculate how long would it take to compute computation of a single sub-unit. $t = \frac{200Gops}{9.5GFLOPs} = 21.0526s$.

1. If only a single sub-unit is activate, then the power consumed is expressed by: $P = 2W + 2W + 4W(k)^3 + 0.25W$ where k represents the voltage and frequency scaling factor $(P_{generalpower} + P_{sub-uint0} + P_{dynamic}k^3)$. Therefore, the energy consumed is:

$$\begin{aligned} E &= P\frac{t}{k} \\ &= (2W + 2W + 4Wk^3 + 0.25W)\frac{t}{k} \end{aligned}$$

   Let's pick a simple scaling factor of $k = \frac{1}{2}$, then $E = (4W + 4W(0.5)^3 + 0.25W)2t = 9.5tJ$. Therefore, if we scale voltage and frequency by $\frac{1}{2}$ the circuit is less energy efficient. If $k = \frac{1}{3}$, then $E = 13.19tJ$. Therefore, scaling using a single sub-unit is less efficient then not scaling at all and running on a single sub-unit.

2. Utilizing multiple sub-units requires greater static power, but reduces the execution time and the parallelization allows scaling of the voltage and frequency by the number of activate sub-units. Let N be the number of sub-units used for the application $(k = \frac{1}{N})$. The power consumed is expressed: $P = 2W + 2W + 6W(N-1) + 4W(\frac{1}{N})^2 + 0.25 * N$

N	Energy (J)
2	121.053
3	120.663
4	122.368

3. In race to halt you eliminate static power consumption. Intuitively, this is more efficient then running on a single sub-unit at the original voltage and frequency. Therefore, the energy consumed $(4W + 0.25W)t + 32J = 121.474J$.

From this analysis, the best option is to run the application across 3 sub-units, scaling the voltage and frequency due to the parallelism.

Problem 4: DRAM in the Real World

DRAM has a high density which makes it useful, but also has a long access latency. Modern DRAM modules are organized and implemented to contain structures to minimize this latency. Reducing access latency is paramount to CPUs, but also any digital circuit which relies on high bandwidth, high density storage (GPUs, FPGAs, accelerators, network cards, etc). Another primary concern is power consumption. Over the years, power saving strategies have been implemented in DRAM. Provide answer to the following prompt to learn about DRAM organization and common performance techniques used to improve performance of DRAM (feel free to consult online resources):

1. In your own words, describe what a memory bank is.
2. In your own words, describe what a memory rank is.
3. In your own words, describe what a memory module is.
4. In your own words, describe what the power-down mode does in DRAM.
5. From the answers above, it is understood that DRAM is architected as a hierarchy. Explain how this hierarchy can decrease latency and reduce power consumption.
6. Provide the primary reason you cannot repeatedly access the same row, in the same bank as clock frequency increases.
7. 1-T DRAM designs usually include a "row buffer"—a register on the periphery that is used to register an entire row. Explain how this register could be used and why it's a good idea.

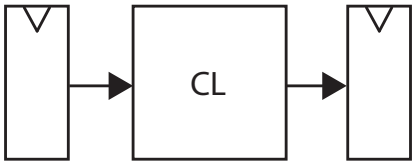
Solution:

1. A memory bank is a single memory array. In a CPU a memory bank contains data for some part of the address space.
2. A memory rank is a collection of individual DRAM chips which shared the same address lines and is composed of many memory banks.
3. A memory module is a collection of individual DRAM chips which can be composed of a single or multiple ranks. For example, each stick of RAM in a computer is a memory module.
4. Power down mode disable a memory module to reduce static power consumption. Other DRAMs have a low power mode, or idle mode which is intended to achieve a similar effect, reduce static power consumption.
5. In general the access time in DRAM is longer than SRAM. However, every read must be followed by a consecutive write to replenish the value. Therefore, every row in a bank, has some downtime following a read
6. It reduces power and increases speed (reduces response time). RAM accesses exhibit spacial locality to a high degree: it's likely that access to one word in a DRAM row is likely followed by another access to the same row. Buffering the row saves having to read the memory cells again, returning a value to the system faster and using less power. For

writing: a row is opened (copied into the row buffer) and constituent bytes/words are updated before the entire buffer is written back.

Problem 5: Energy Efficiency Improvements

The block diagram shown below represents the critical path for a circuit. This path has 0 slack. The timing specifications are as follows: $\tau_{CL} = 5ns$, and $\tau_{setup} = \tau_{clk-Q} = 1ns$.



On average, at some V_{DD} the energy for one data item passed through the combinational logic block is 3.6 J. The registers each consume 0.1 J on average for each new data word stored.

Assume you have an application where the latency for the output corresponding to the first input does not matter. In other words, your application is latency-insensitive. However, after the first output appears the application requires the circuit to produce results at a rate of 125MHz (one result per cycle).

It is possible to split the combinational logic evenly (in terms of both delay and energy) into multiple blocks.

Devise a scheme that would improve the switching energy efficiency while meeting the application requirements. Compare the switching energy per result of the original circuit and your new one.

Assume that voltage and clock frequency scale together.

Solution:

The slack is 0 which implies that the circuit is running at 142.86 MHz ($\frac{1}{\tau_{CL}+\tau_{setup}+\tau_{clk-Q}}$). As is, the circuit operating faster than required. A possible solution is to reduce the frequency to 125MHz. This would be a scaling factor of 0.84.

It is safe to assume that static power does not change with voltage scaling.

Scaling Approach:
Since switching power is $P_{\text{SW}} = \frac{1}{2}\alpha C V_{dd}^2 F$, the power ratio between version, when F and V_{dd} are scaled by k is:

$$\frac{P_{\text{SW new}}}{P_{\text{SW orig}}} = \frac{V_{\text{dd new}}^2 F_{\text{new}}}{V_{\text{dd orig}}^2 F_{\text{orig}}} = \frac{(kV_{\text{dd orig}})^2 k F_{\text{orig}}}{V_{\text{dd orig}}^2 F_{\text{orig}}} = k^3$$

However, changing the clock frequency by a factor of k changes the amount of time it takes for the operation to complete by a factor of 1/k. Since Energy = Power*Time, the net scaling in energy/operation in the logic is k^2.

For the scaling above, the energy per operation becomes $(3.6J + 2 \cdot 0.1J)0.84^2 = 2.68128J$ compared to $3.8J$ in the original which is a 29.44% energy reducation.

Pipelining Approach:
A more substantial energy efficiency improvement can be gained by pipelining the combinational logic. When pipelining, the critical path is reduced which allows the clock rate to be increased without changing V_{DD}. Just increasing the frequency due to pipeline is not beneficial because to time execute remains the same therefore the total energy consumed remains constant.

This changes the energy/op scaling factor because power is scaled by k^3 but time/op is now scaled by n/k. This leads to energy/op scaling by nk^2.

Let N be the number of pipeline stage. The total amount of time an operation takes to get through the pipeline changes to become the number of pipeline stages (N) * the delay through each pipeline stage.

Assuming the combinational logic is evenly split, the critical path when pipelined is $\frac{5ns}{N} + 2ns$, therefore the scaling factor is: $k = \frac{\frac{5ns}{N}+2ns}{8}$. The energy/operation is $(3.6J + (N+1) \cdot 0.1J)Nk^2$. Minimize the general expression to calculate how many pipeline stages you need: $N = 2.23$. Since 2.23 is not an integer, we must calculate the energy consumed for $N = 2$ and $N = 3$, and select whichever is minimum.

$$E_{N=2} = 2.46797$$
$$E_{N=3} = 2.52083$$

Therefore, select 2 pipeline stages.

Overall, the best approach is to pipeline with 2 stages and scale the voltage and frequency.