Milestone Due: Tuesday, November 19, 10:00PM

Due: Tuesday, November 26, 10:00PM

Design Problem: Design a synchronous register file with one read and one write port.

- Target technology is the High Performance 22nm process (/home1/e/ese370/ptm/22nm_HP.pm)
- Design should be synchronous with one cycle of delay between presenting a read address and the read data value returning.

Reads to the same address as the write in a cycle return the value being written.

- We will concretely focus on a 16 word register file holding 16b-wide data words.
- Inputs: WAddr<3:0>, RAddr<3:0>, WData<15:0>, Wen, Ren, Clock
- Outputs: RData<15:0>

Design Metrics:

- **area**: Sum the total transistor width for the entire design.
- **memory cell area**: Sum the total transistor width for the repeated memory cell in the memory core.
- **minimum cycle time**: What is the smallest cycle time for which the design will operate? Reads, writes, and simultaneous reads/writes all occur within this cycle. Since this is the complete cycle time, it includes precharge time and any clocking overheads.
- **write energy**: Measure the energy writing 0xFFFF into a cell that previously held 0x0000 (and vice-versa). Report the larger value.
- **read energy**: Measure the energy for a read of a word with 0x0000 and a read of a word with 0xFFFF. Report the larger value.
- **standby energy**: Measure the energy of a cycle on which no reads or writes occur.
**Design:** Your primary objective is delay minimization. Your secondary objective is minimum memory cell area.

- \( V_{dd} \leq 1.0V \).
- Memory cells should be static.
- You select decoder design, driver design, output sensing/buffer, control timing.
- You may **not** use resistors in the design.

**Recommendations:**

- Even before you worry about delay minimization, you need to assure that you can achieve correct operation.
- Pay careful attention to the timing of controls for writes. What do you need to guarantee to make sure you only write into the intended word cells on a write cycle? What test cases will you need to validate these properties?
- This design will likely have around 2000 transistors.
- Use hierarchy in your schematics. You should have a single memory cell design that you instantiate. You should also organize cells into word rows. You may also want a separate organization into columns for reasons noted below.
- A common trick for regular design like this is to extract the critical path to avoid having to simulate the entire design during sizing and delay optimization. Can you build a design that has a single row and column rather than all row\( \times \)column memory cells? The parasitic loading of other cells is important, so you want to capture that when you extract your critical path (hence the need to capture a row and a column; similar reasoning should also apply to address decoders).
- Use unit testing. Make sure your memory cell, decoder, buffers work in isolation before you integrate larger structures.
- To generate a sequence of logic values, you can use a VPWL (Piece-Wise Linear) source in electric. This produces a SPICE PWL. Please see the ngspice manual for more information.
- We are not necessarily looking for innovation in the core memory cell circuit, but it will need to be sized for the technology and adapted for your usage.
- Cell sizing may be asymmetric to address the different needs of reads and writes.
**Milestone:** (Due 11/19) Turn in as a single PDF.

- Estimate bit line capacitance for target design point. Report and use in the simulations below to demonstrate cell operation.
- Include schematics for column driver and **sized** memory cell. You may use PWL controls and ideal voltage references for this phase of the design. PWL controls and ideal voltage references are temporary scaffolding; you will implement the controls and references in transistors for the final design.
- Identify the set of cases you should test in order to validate correct operation of your memory cell on both writes and reads.
- Demonstrate that you can write to the memory cell. Include schematics for setup and simulation waveforms as necessary.
- Demonstrate that you can read from the memory cell without losing the value. What test cases do you need to demonstrate this? Include schematics for setup and simulation waveforms as necessary.
- Identify the constraints on write timing in the full register file design (not just the cell) to guarantee correct operation, and identify how you will verify this (what test setup? what test cases?).

You want to make sure you know how the core needs to be controlled before building logic to generate the controls, and you want to an operational memory core.
Report: Your report should be a single, stand-alone document (including stand-alone from your milestone report) and should include:

- Schematics for the design – make sure transistor sizing annotations are easily readable in the diagram you include in the report.
- For both the read and the write operations show:
  - Waveform showing timing of key signals in the operation. Identify the delay of each component that makes up the critical path.
- Text description of the memory operation and design choices. Identify and describe the operation of each of the sub-components. Your description should make it easy for us to understand how and why your design works.
  - Explain the timing requirements of the memory for correct operation and how they are met.
  - Explain rationale for sizing of memory cell—why are the transistors sized as you did, and how did you arrive at this conclusion? Show simulations as necessary.
- Description of how you optimized the delay of each sub-component and the ensemble, including the tradeoffs you needed to make. Show experimental simulation results as necessary.
- Description of how you validated correctness of the design. This will include a description of your test cases.
- Summary of the design metrics to two significant figures. Include supporting evidence in the form of equations and/or simulation results.
- Identify how your design will scale to larger memory capacities and word sizes. This should include an equation for the delay of your design as a function of the number of words \( d \) and the word width \( w \).
- Please include a statement on your final submission:

  I, your-name-here, certify that I have complied with the University of Pennsylvania’s Code of Academic Integrity in completing this project.
Voltage Midpoint Reference Generator

Here’s a voltage reference generator that may be useful for your memory design. The inverter with input and output tied together settles to the midpoint of the transfer curve where the input and output are equal. The second inverter is driven to the same point. It serves to isolate the output from the reference generated by the first one. The pass gate allows you to connect or disconnect this reference voltage to a node.

You may need to resize the driving buffer and pass transistor. Sizes shown are just to give concrete examples.