Homework 4 Computer Organization

Due: Tuesday, 10/10/06

Chapter 3 Exercise: 36, 40 from the text, AND

Additional Problems:

1) Draw a 4-bit register that is able to perform the following operations:
   - parallel read/output of all bits
   - parallel write/input of all bits
   - circular shift right one bit position (value shifted out of least-significant bit is shifted into the most-significant bit)
   - logical shift right one bit position (value of 0 shifted into the most-significant bit)
   - arithmetic shift right (sign-extend the most-significant bit)

Hint: For each D-flip flop, use the output of a MUX as the D-input. You can draw block-diagrams for flip-flops and MUXs without showing their internal gate implementations.

2) Using the discussion in the book and the register file handout, draw a complete (not just a one-bit slice) register file that has:
   - 3 registers
   - 3-bits per register
   - one write-port
   - two read-ports

You can draw block-diagrams for flip-flops, decoders, and MUXs without showing their gate implementations, but you should show all the flip-flops, decoders, MUXs, and connecting wires.

3) How well does this register-file design scale? Suppose that we are implementing a 128 M x 8 (128M registers, each with 8 bits) register file with one write-port and one read-port.

a) How many and what type of decoder(s) would be needed?

b) How many total gates (assume 9-input limit on AND & OR gates) would be needed to implement this (these) decoder(s)?

c) How many and what type of MUX(s) would be needed?

d) How many total gates (assume 9-input limit on AND & OR gates) would be needed to implement this (these) MUX(s)?

e) Assuming D flip-flops to store each bit (5 gates/flip-flop). What % of the total gates is used to implement the D flip-flops?

4) Redo the previous question using the 128 M x 8 square-memory implementation similar to the "Implementation of Large Memory Chips" class handout.