

Announcement: There will be one more homework assignment on caching to be due Monday, May 7.

Reading: Patterson and Hennessy sections 6.1-6.6 (plus material from chapter 3 as necessary). You can omit the details of the pipeline control but pay attention to how **data** is transmitted and processed within the pipelined processor. This determines what hazards can be removed and what hazards require the pipeline to stall.

Mano sec. 11-2, 11-3, 11-4, 11-5, and 11-6 for more details on topics discribed in the lectures.

Problem 1: Implement a single bit 8-to-1 multiplexor with a combination of boolean gates (to decode the select input) and three-state buffers (Mano p. 100ff) to drive the output wire. Draw the circuit diagram. Explain what **guarantees** one and only one of the 8 tri-state buffer outputs will be in a “connected” or low impedance state at one time. (Attention physics students: impedance means complex electrical resistance, measured in ohms, and is often used interchangeably with resistance.)

Problem 2: Draw Unified Modeling Language (UML) Sequence Diagrams to explain the sequences described in Mano listed below. Follow the links on <http://www.cs.albany.edu/~sdc/CSI404> to the UML Sequence Diagram example and usage guidelines.

- a. Mano figures 11-3 and 11-4.
- b. Mano figures 11-5 and 11-6.
- c. One Direct Memory Access (DMA) transfer from an I/O device to memory. Include the device driver software run by the CPU commanding the device to do the transfer, the device using the bus for the transfer, the interactions with memory, the device signaling the completion of the transfer with an interrupt (i.e., show that the CPU is affected by the interrupt), and the device driver’s interrupt service routine being called.

Problem 3: The **speedup due to pipelining** of the UC. Berkeley dorm laundry is defined as

$$\text{Speedup} = \frac{\text{Time for all the jobs without pipelining}}{\text{Time for all the jobs with pipelining}}$$

(thus it is good if Speedup > 1.) Calculate **numerically** for each case below the speedup when the number of students who use the laundry (one after the other) is: 1; 2; 4; 6; 10; 20; 200; 20,000,000.

Problem 4: Calculate the (approximate) speedup for a large number of instruction executions of the MIPS 5 stage pipeline when:

- a. You assume there are no stalls.
- b. You assume that 10% of the instruction executions are memory transfer (load or store) instructions that cause 1 stall clock step each and 15% of the the instruction executions are control transfers that cause 2 stall clock step for each.

Hint: Figure out the average number of clock steps between instruction completions. For example, if there would be 1 stall for every 3 instruction executions, this average would be 1.0 (for the normal single step to complete each instruction) plus 0.333 (approximately one out of three instructions make the computer take an extra clock step) which equals (1.333).

(Assignment continued on the reverse side.)

Problem 5: Use the following code fragment:

```
loop: lw    $1, 0($2)
      addi  $1, $1, 15
      sw    $1, 0($2)
      addi  $2, $2, 4
      sub   $4, $3, $2
      bne  $4, $0, loop
      nop
```

- a. Show the timing (showing any stalls) of this instruction sequence as it progresses down the MIPS pipeline assuming that no forwarding hardware is used. Assume the branch is implemented by flushing the pipeline. Do assume that the value read from a register in the register file is the same as the (new) value written to the same register during the same clock step.
- b. Now show the timing when forwarding hardware is used.