Slide 1

Administrivia

- Reminder: Quiz 4 Wednesday.
- Homework 5 on Web. Due next Monday.

Slide 2

Minute Essay From Last Lecture

- Questions:
  - In figure 5.13, the multiplexor at the far right has two inputs. One comes from the "read data" output of the data memory and is 32 bits. Where does the other input come from, and how many bits is it? Where does the output go, and how many bits is it?
  - In figure 5.13, there are three separate things that can perform addition (two adders and a full ALU). Do we need all three? Why or why not?
- Answers?

Slide 3

Generating Control Signals — ALU Control (Recap)

- First use instruction opcode to generate `ALUOp` (details deferred for now).
- Then design CL block to generate ALU control inputs from `ALUOp` and instruction's function field, as sketched last time, details in Appendix C.

Slide 4

Using Instruction's Fields

- Some input to functional units comes directly from the instruction:
  - Read register 1.
  - Read register 2.
  - Write register.
  - Sign extender.
  - Block to generate ALU control input.
- For each of these, figure out which bits we need. This gives figure 5.17.

Slide 5

Generating Control Signals, Continued

- From figure 5.17, we need the following control signals (meanings described in figure 5.18):

  `ALUOp`, `RegDst`, `RegWrite`, `ALUSrc`, `PCSrc`, `MemRead`, `MemWrite`, `MemtoReg`.

- All can be generated from instruction opcode, except `PCSrc`. For that, we generate a `Branch` signal from the opcode and combine it with the ALU's `Zero` output.
- Adding a CL block to do this gives figure 5.19. What this CL block is supposed to do is given in figure 5.20.

Slide 6

Minute Essay

- Fill in the following table (using figure 5.19):

	add	lw	sw	beq
RegWrite	?	?	?	?
MemRead	?	?	?	?
MemWrite	?	?	?	?