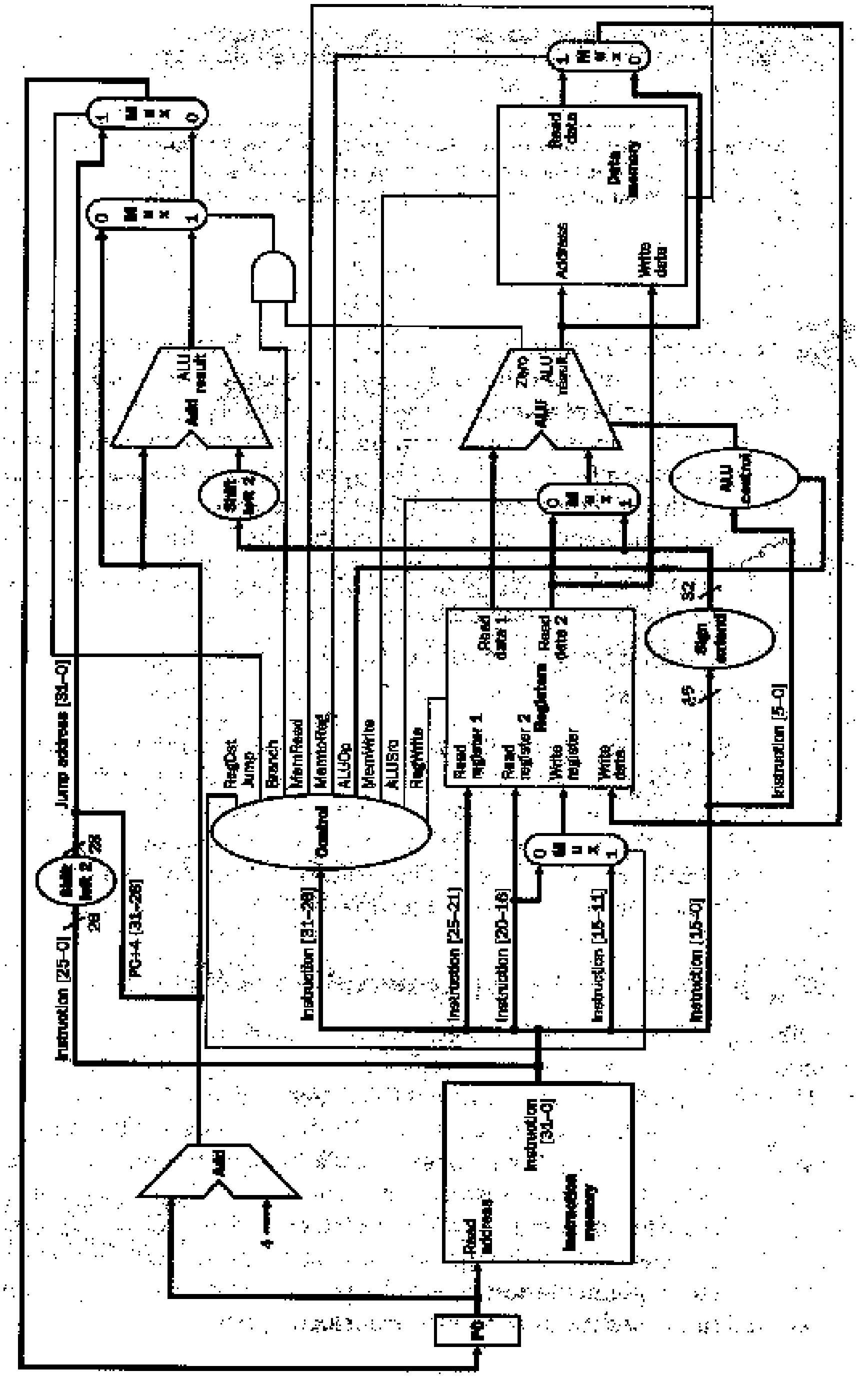
CS420/520 Homework Assignment 3: Datapath

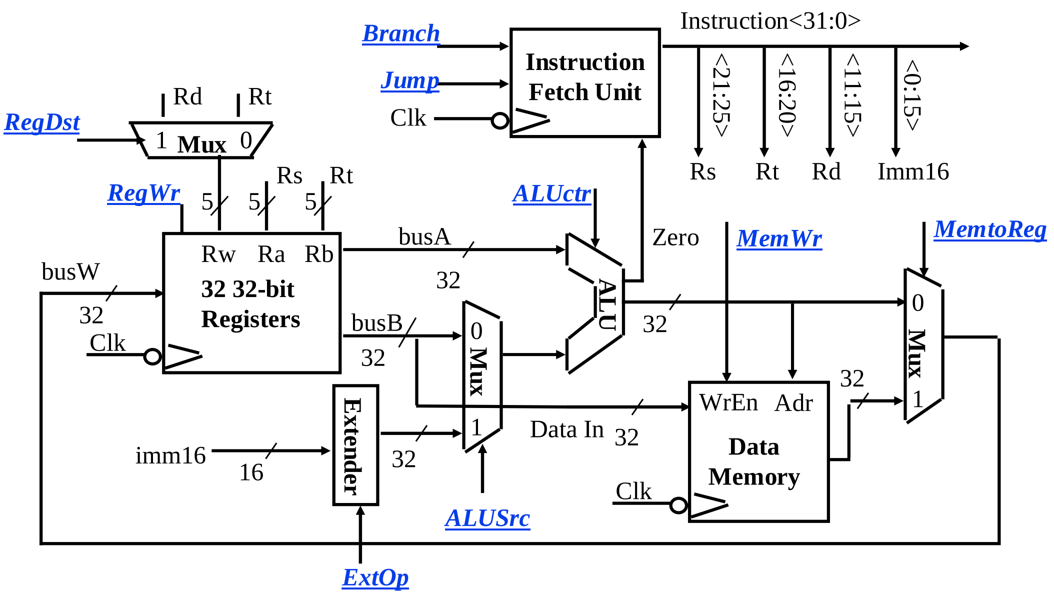
DP.1 [15] Describe the effect that a single stuck-at-0 faults (i.e., regardless of which it should be, the signal is always 0) would have on the multiplexors in the single-cycle datapath shown in the following figure. Which instructions (R-type, Imme, load, store, and branch), if any, would NOT work? Consider each of the following faults separately: RegDst = 0, ALUSrc = 0, MemtoReg = 0, Zero = 0.



DP.2 [15] This exercise is similar to Exercise DP.1. But this time consider stuck-at-1 faults (i.e., regardless of which it should be, the signal is always 1). Which instructions (R-type, Imme, Base+Offset, and PC-Relative), if any, would NOT work? Consider each of the following faults separately: RegDst = 1, ALUSrc = 1, MemtoReg = 1, Zero = 1.

DP.5 [6] We wish to add the instruction **addi** (add immediate) to the single-cycle datapath. Add any datapaths and control signals, if necessary, to the single-cycle datapath and show the necessary addition to the following control table.

A Single Cycle Datapath



CS420/520 control.4 UC. Colorado Springs Adapted from ©UCB97 & ©UCB03

Instruction	RegDst	ALUSrc	Mem-Reg	Rew Write	Mem Write	Branch
R-format	1	0	0	1	0	0
LW	0	1	1	1	0	0
SW	X	1	X	0	1	0
BEQ	X	0	X	0	0	1

The setting of the control lines is completely determined by the opcode fields of the instruction.

DP.7 [10] This exercise is similar to Exercise DP.5, except that we wish to add the instruction **bne** (branch if not equal) to the single-cycle datapath shown in the figure on the previous page. (note that **beq** should still be supported at the same time).

* you do not need to draw the whole figure again, but show the modifications and additions for bne. You may refer to beq in Power Point slides, datapath (page 40).

DP.8 [5] If we want to add some storage element to our datapath. The storage element will be updated by some instructions, but will not be updated by some other instructions. Do we always need to have a clock? Do we always need to have a Write Enable signal? In what situation that we need a clock signal, but not Write Enable signal for a storage element in some datapath?

DP.9 [6] Use the drawing of PLA Implementation of for Control (refer to Power Point slides, datapath), to show the PLA implementation for "Extop" signal (**Drawing II**). The truth table for "Extop" is on Power Point slides, datapath.

DP.10 [6] What is the key motivation for a multi-cycle datapath design? Is it possible that a single-cycle datapath has better performance than a multi-cycle datapath? If so, give a simple case study.

DP.11[8]What function units can be used more than once per instruction in a multi-cycle datapath? How they are used (in which stage, for what purpose)?

DP.12 [15] In estimating the performance of the single-cycle implementation shown in the following figure, we assume that only the major functional units have any delay (i.e., the delay of the multiplexors, control unit, PC access, sign extension unit, and wires is considered to be negligible). Assume the following delay:
ALU: 2ns, Register File: 1ns, Instruction Memory: 2ns, Data Memory:2 ns.
Adder for PC + 4: ***X***ns
Adder for branch address computation: ***Y***ns

a) What would the clock cycle time be if ***X*** = 3 and ***Y*** = 3?
b) What would the clock cycle time be if ***X*** = 5 and ***Y*** =5?
c) What would the clock cycle time be if ***X*** = 1 and ***Y*** = 8?

- You must show the procedure details for each answer you give.

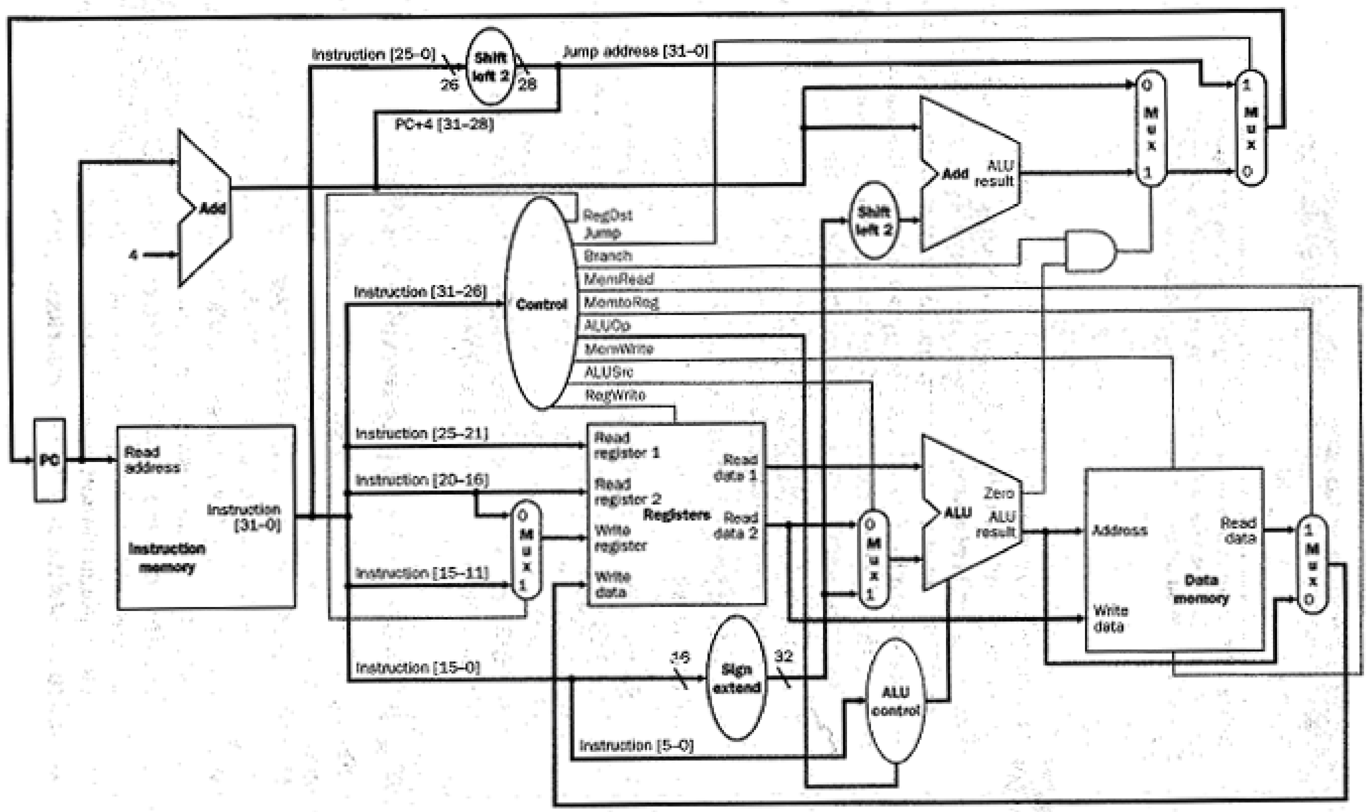


FIGURE 5.29 The simple control and datapath are extended to handle the jump instruction. An additional multiplexor (at the upper right) is used to choose between the jump target and either the branch target or the sequential instruction following this one. This multiplexor is controlled by the jump control signal. The jump target address is obtained by shifting the lower 26 bits of the jump instruction left 2 bits, effectively adding 00 as the low-order bits, and then concatenating the upper 4 bits of PC + 4 as the high-order bits, thus yielding a 32-bit address.