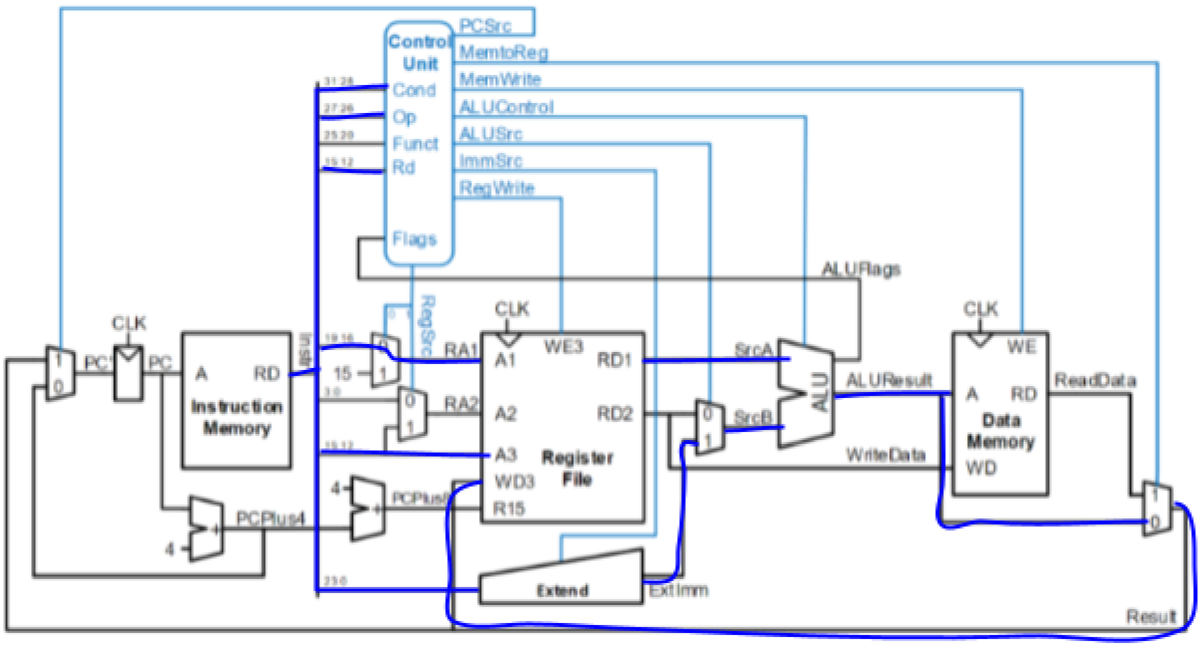
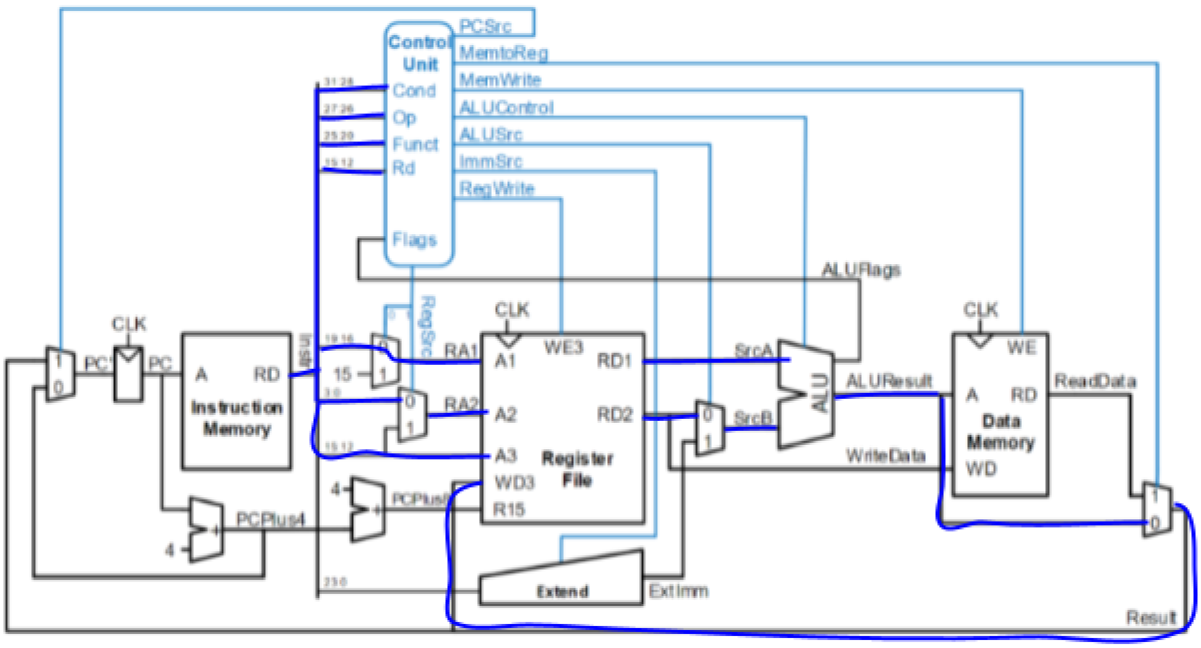
EECSE 469 Computer Architecture I
Brayden Lam, Christopher Andrade
HW 3

1. With the single-cycle CPU shown in lecture, highlight the paths used by the following instructions. Also, fill out the corresponding control signal table:
    a. AND R10, R9, #3



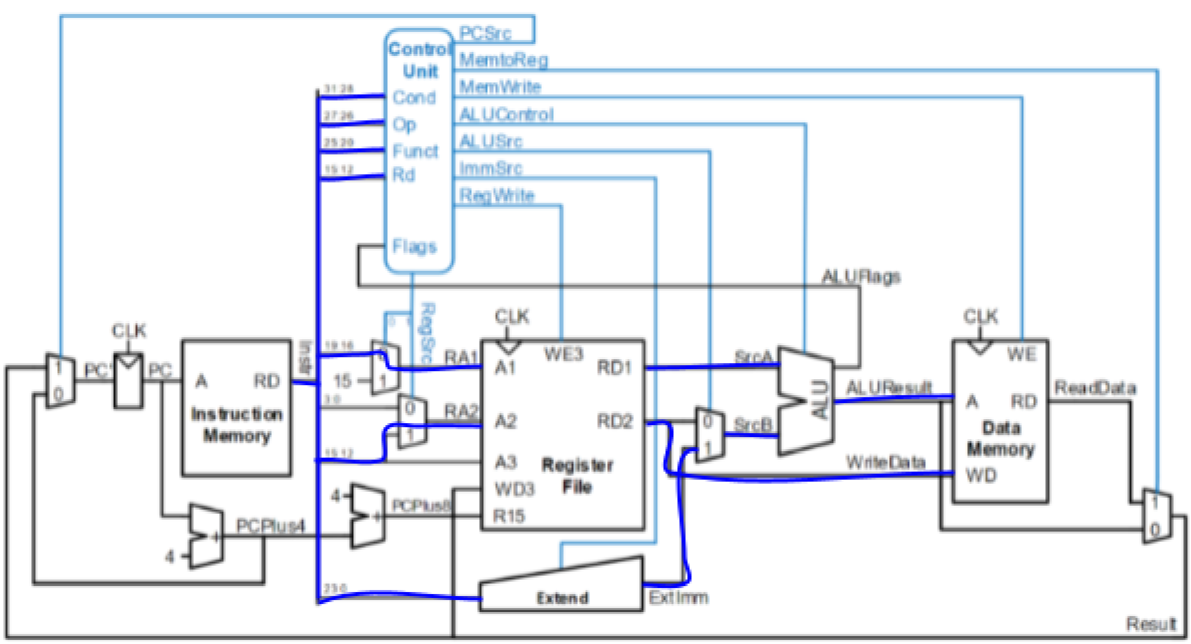
PCSrc	MemtoReg	MemWrite	ALUControl	ALUSrc	ImmSrc	RegWrite	RegSrc [1:0]
0	0	0	10	1	0	1	X0

b. ADD R1, R3, R4



PCSrc	MemtoReg	MemWrite	ALUControl	ALUSrc	ImmSrc	RegWrite	RegSrc [1:0]
0	0	0	00	0	X	1	00

c. STR R4, [R11, #0]



PCSrc	MemtoReg	MemWrite	ALUControl	ALUSrc	ImmSrc	RegWrite	RegSrc [1:0]
0	X	1	00	1	1	0	10

2. We would like to implement a new custom ARM instruction, SOIRIP Rd, Rn, Rm (Subtract Only If Result Is Positive). This is like a normal subtract instruction, except if the result of the subtraction is negative, write a 0 back into the destination register. If the result is positive, write the computed result back into the destination register. The bit encodings for Rd, Rn, and Rm are the same as SUB with I=0, and Instruction[4] = 0. Your custom instruction may overwrite SUB (and SUBS).

a. Examples: Let R1 = 4, R2 = 5, R3 = 6

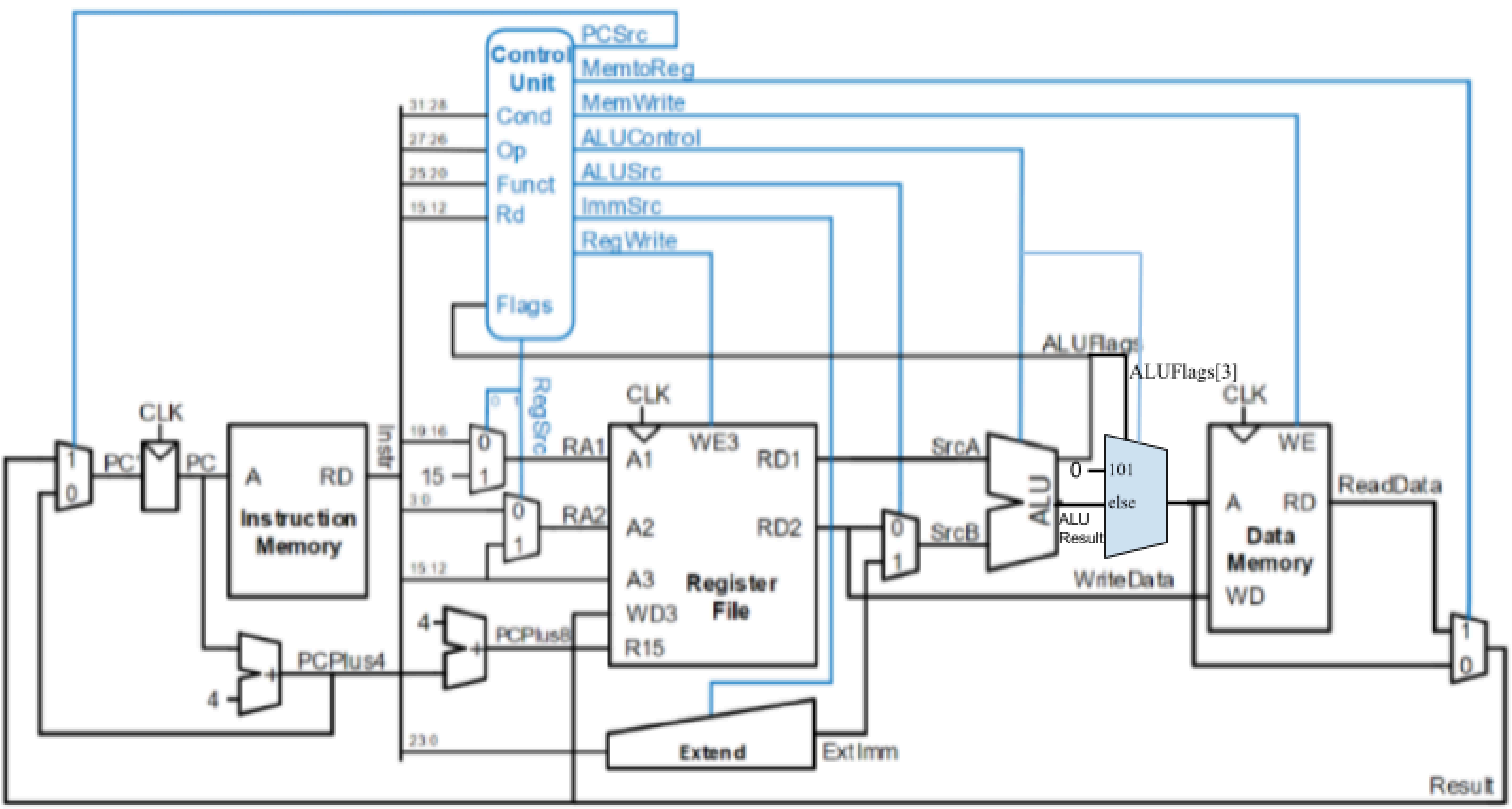
i. SOIRIP R5, R1, R2

This would result in R5 = R1 - R2, which comes out to be -1, so the value 0 would be stored in R5.

ii. SOIRIP R6, R3, R2

This would result in R6 = R3 - R2, which comes out to be 1, so the value 1 would be stored in R6.

Modify the datapath shown in class (provided below), adding additional control signals and hardware as necessary.



3. Give a diagram of the datapath (including any necessary control signals) needed to implement **ONLY** the custom instruction "LDRM Rd, [Rn, Rm]". LDRM behaves like LDR, but performs Rn*Rm during load address computation. Rd, Rn, and Rm are encoded in the same bits as the LDR instruction with no immediate. You can use the provided multiplier block as a component.

   Your datapath should be as simple as possible. You can use the processor from class as a reference, but you should not include anything that is not needed.

   a. Example: LDRM R3, [R1, R2] results in R3 <- data memory[R1*R2]
   b. Multiplier Unit: