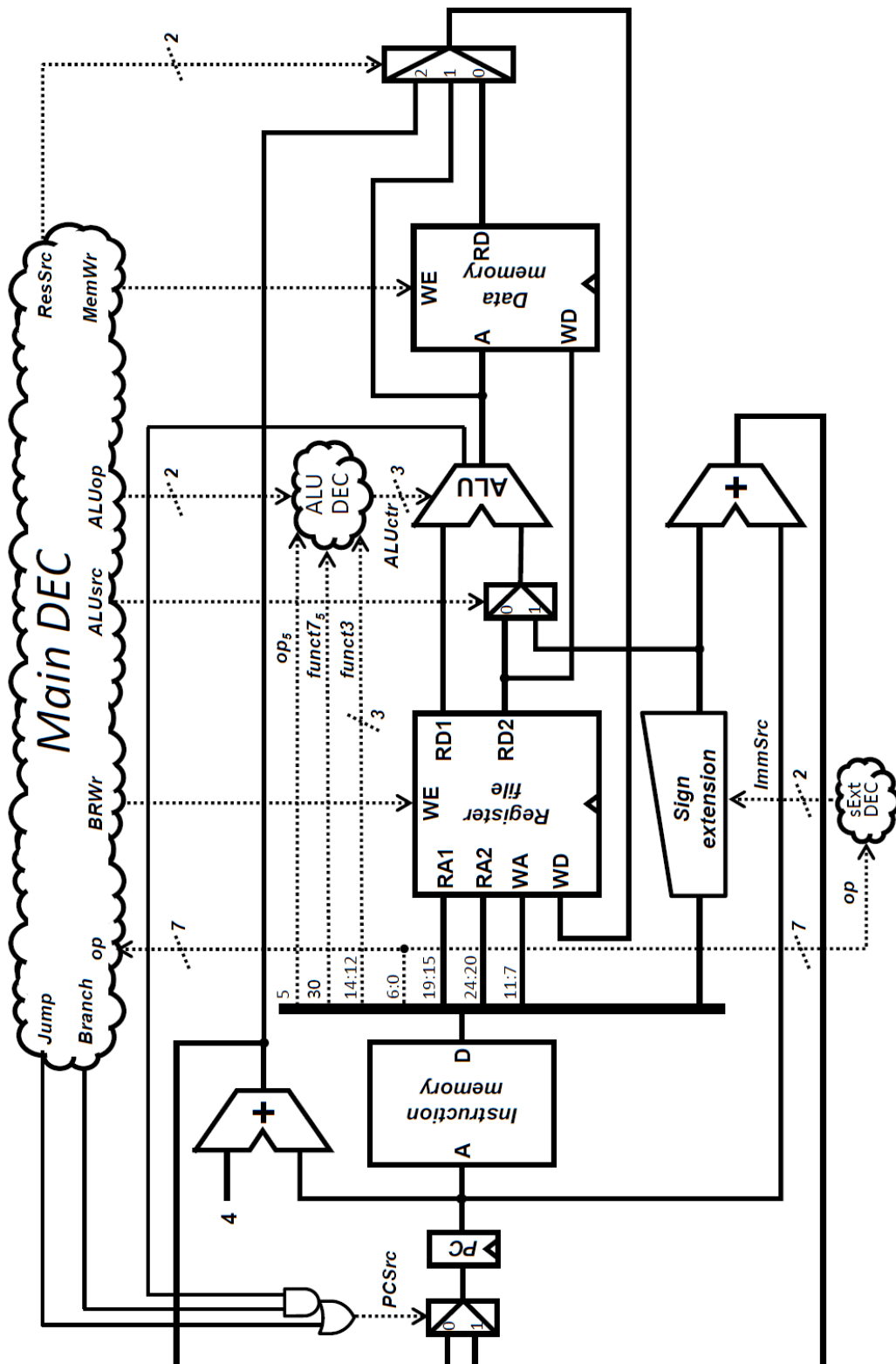
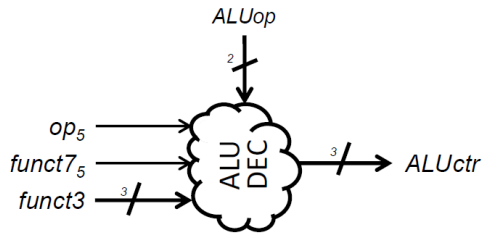


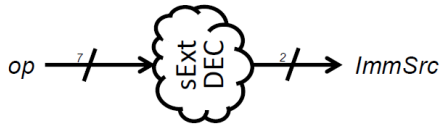


INTRODUCTION TO COMPUTERS II SINGLE-CYCLE MICROARCHITECTURE

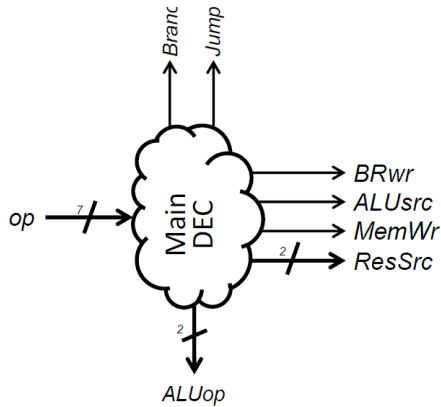




ALUop	op ₅	funct7 ₅	funct3	ALUctr
00 ^(add)	X	X	XXX	000 ^(A + B)
01 ^(subtract)	X	X	XXX	001 ^(A - B)
10 ^(operate)	0	X	000 ^(addi)	000 ^(A + B)
10 ^(operate)	1	0	000 ^(add)	000 ^(A + B)
10 ^(operate)	1	1	000 ^(sub)	001 ^(A - B)
10 ^(operate)	X	X	010 ^(slt/slti)	101 ^(A < B)
10 ^(operate)	X	X	110 ^(or/ori)	011 ^(A B)
10 ^(operate)	X	X	111 ^(and/andi)	010 ^(A & B)



Op	ImmSrc
0000011 ^(lw)	00 ^(I-type)
0100011 ^(sw)	01 ^(S-type)
0010011 ^(l-type)	00 ^(I-type)
0110011 ^(R-type)	-
1100011 ^(beq)	10 ^(B-type)
1101111 ^(jal)	11 ^(J-type)



op	Branch	Jump	BRwr	ALUsrc	ALUop	MemWr	ResSrc
0000011 ^(lw)	0	0	1	1	00 ^(add)	0	00
0100011 ^(sw)	0	0	0	1	00 ^(add)	1	-
0010011 ^(l-type)	0	0	1	1	10 ^(operate)	0	01
0110011 ^(R-type)	0	0	1	0	10 ^(operate)	0	01
1100011 ^(beq)	1	0	0	0	01 ^(subtract)	0	-
1101111 ^(jal)	0	1	1	-	-	0	10