



Computer Architecture

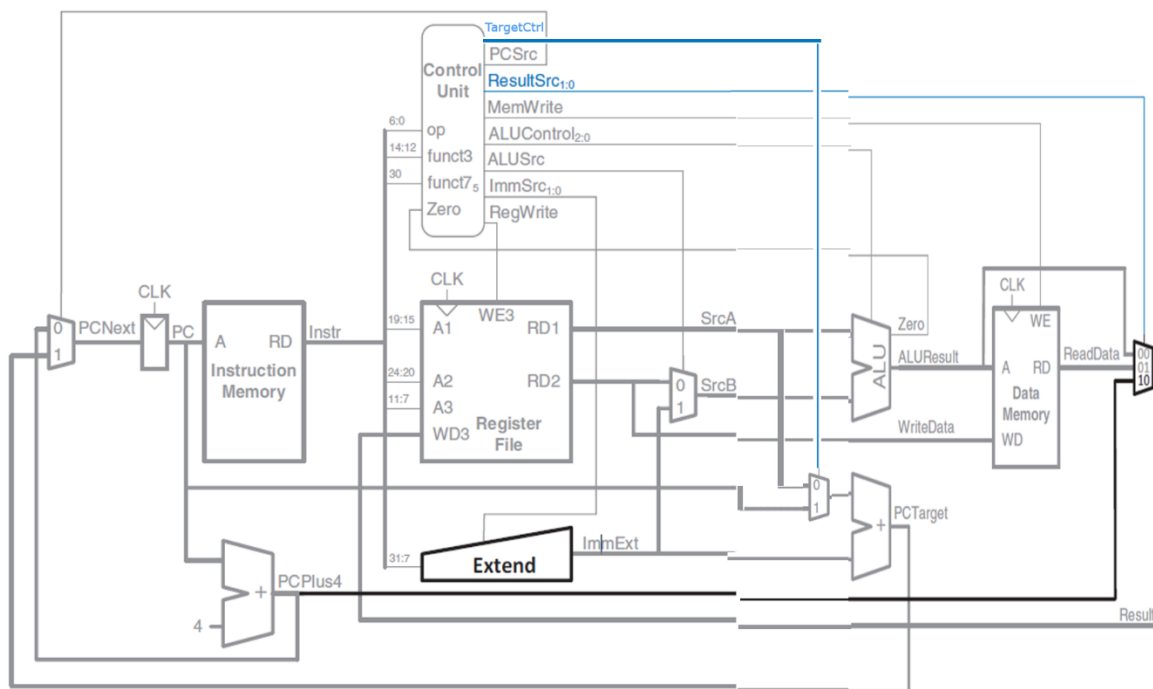
HW4 Report

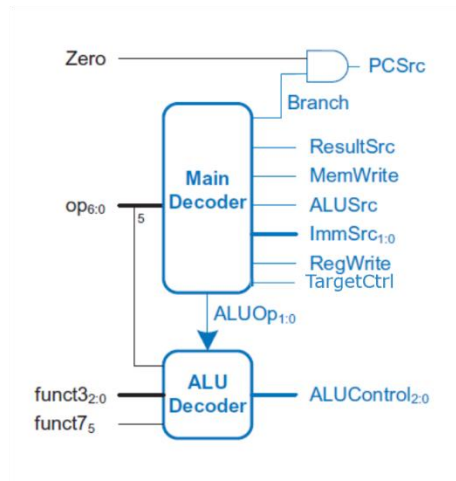
G14 (Mohammad Bahrami & Navid Raeiszadeh)

سوال اول:

الف) بله لازم است یک multiplexer به datapath اضافه شود تا بتوان از بین مقدار رجیستر داده شده SrcA و PC مقدار رجیستر را انتخاب کرد تا با offset جمع شود تا PCTarget تولید شود.

ب)





(د)

جدول ۷.۶ اصلاح شده با اضافه کردن jalr و سیگنال کنترلی TargetCtrl

Instruction	Opcode	RegWrite	ImmSrc	ALUSrc	MemWrite	ResultSrc	Branch	ALUOp	Jump	TargetCtrl
lw	0000011	1	00	1	0	01	0	00	0	1
sw	0100011	0	01	1	1	xx	0	00	0	1
R-type	0110011	1	xx	0	0	00	0	10	0	1
beq	1100011	0	10	0	0	xx	1	01	0	1
I-type ALU	0010011	1	00	1	0	00	0	10	0	1
jal	1101111	1	11	x	0	10	0	xx	1	1
jalr	1100111	1	00	x	0	10	0	xx	1	0

توضیحات تکمیلی:

1100111 (103)	000	-	I	jalr rd, rs1, imm	jump and link register	$PC = rs1 + \text{SignExt}(imm), rd = PC + 4$
---------------	-----	---	---	-------------------	------------------------	---

برای مشخص شدن مقدار PC باید رجیستر rs1 که در ورودی اول رجیستر فایل هست را با مقدار ساینده اکستند شده ی آفست جمع کرد که نوع ساین اکستند از جنس I type هاست که کد 00 را به خود اختصاص داده است.

برای اینکار از جمع کننده PCTarget استفاده میکنیم و برای انتخاب مقدار SrcA نیاز به یک مالتیپلکسر داریم که در قسمت الف شرح داده شد.

مقدار $PC + 4$ که باید در rd قرار گیرد به مالتیپلکسر بعد از DataMemory وارد میشود و در آنجا انتخاب شده و به پورت رایت رجیستر فایل رفته تا در rd نوشته شود.