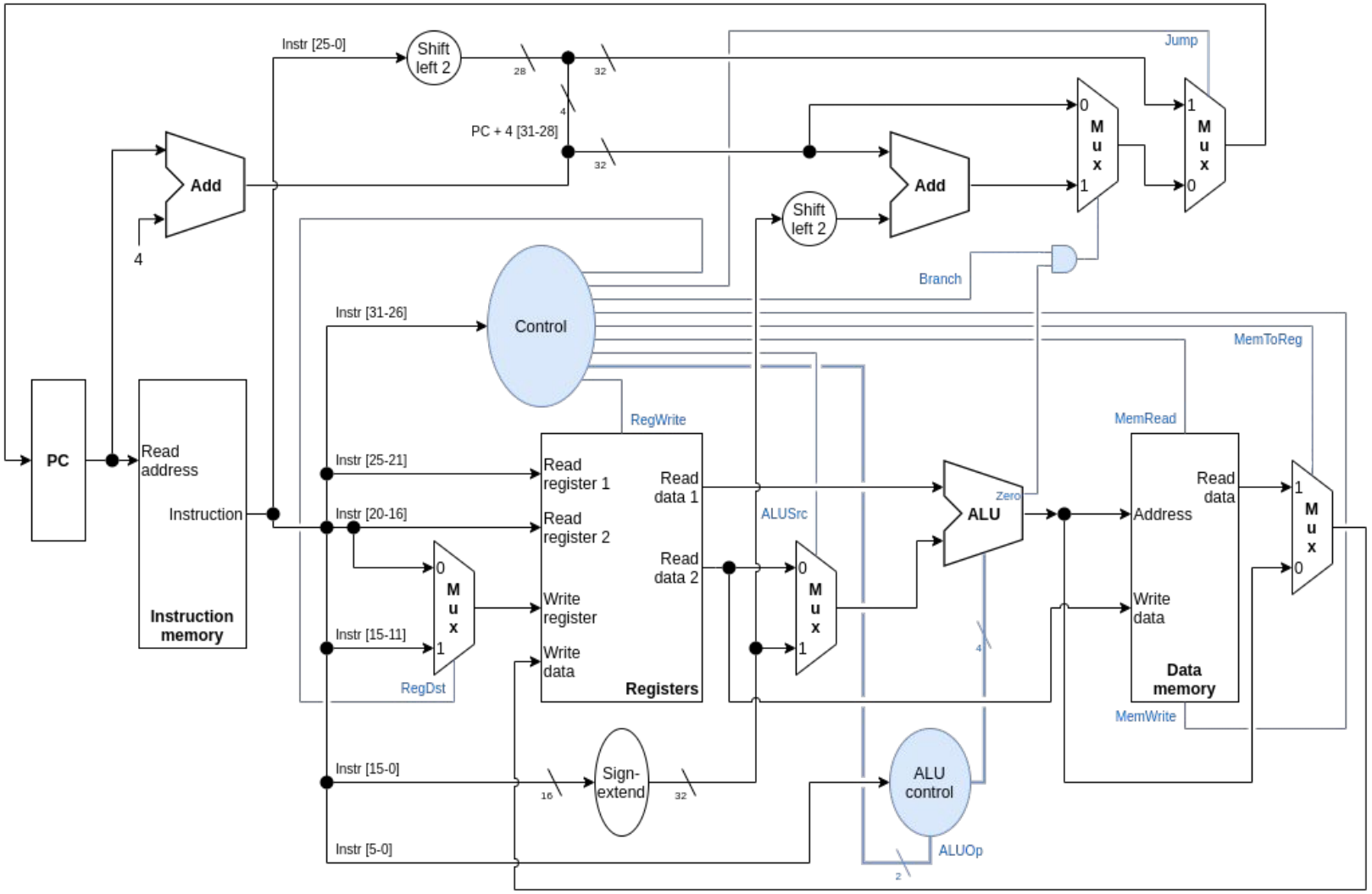
Architettura MIPS

SAPIENZA
UNIVERSITÀ DI ROMA

Modifica architettura ed istruzioni

SAPIENZA
UNIVERSITÀ DI ROMA

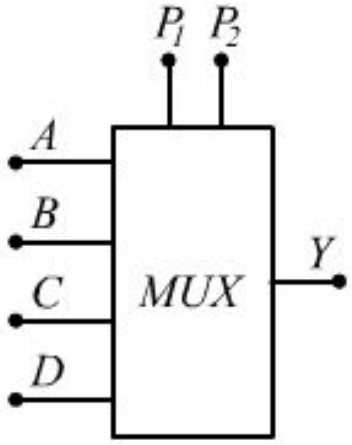
Architettura MIPS



Architettura MIPS - facciamo chiarezza!

MUX

sceglie un segnale da far passare tra quelli in input.
la scelta è effettuata sulla base dei segnali di controllo.



P_1	P_2	Y
0	0	A
0	1	B
1	0	C
1	1	D

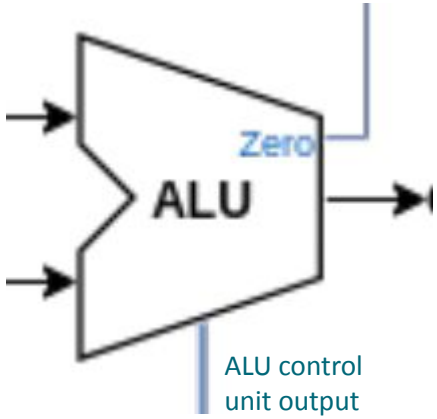
Architettura MIPS - facciamo chiarezza!

ALU

Effettua operazioni aritmetiche sugli input dando in output il risultato.

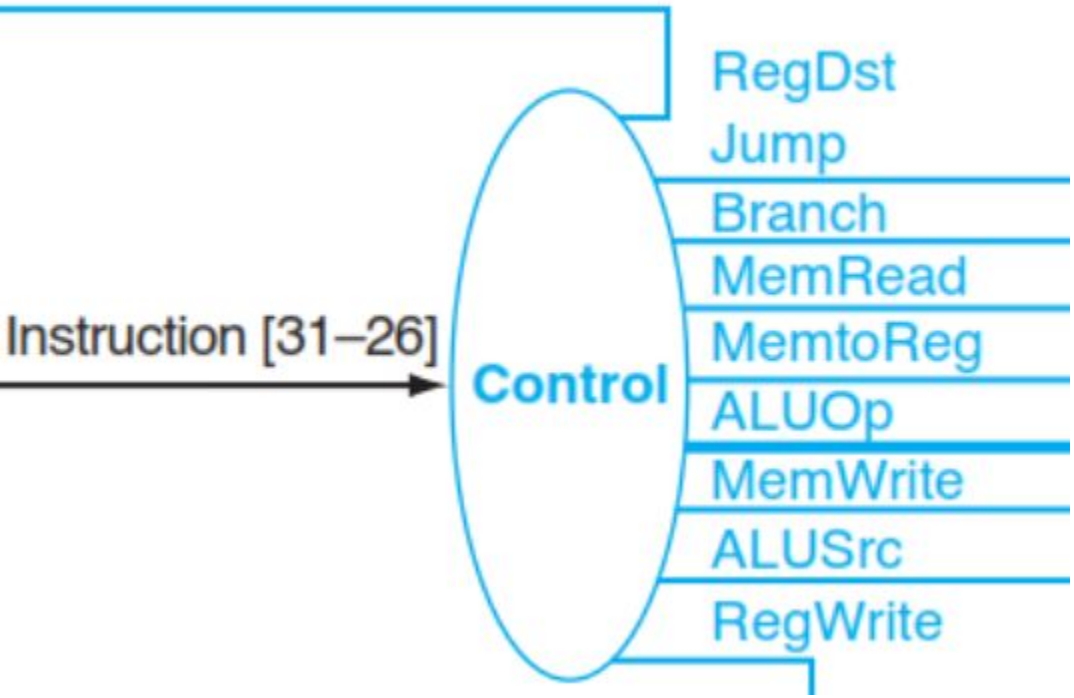
Le operazioni sono scelte dalla ALUOp.

Il segnale *zero* vale 1 se il risultato dell'operazione è 0, altrimenti vale 0.



Architettura MIPS - facciamo chiarezza!

CONTROL UNIT



- Preso in input un instruction code imposta i flag al valore corretto
- i flag servono per controllare il segnale attraverso i MUX e per effettuare varie operazioni

SAPIENZA
UNIVERSITÀ DI ROMA

CONTROL UNIT

Il segnale all'interno della CPU è sempre presente, quindi i flag servono a evitare comportamenti indesiderati (e avere quelli desiderati).

Esempio: se volessi eseguire **add $t0, $t0, $t1** . la CU deve impostare il flag MemWrite a 0 per evitare che sia scritto qualcosa in memoria.

CONTROL UNIT

Non tutti i flag sono utili per tutte le istruzioni. Per alcune operazioni un flag potrebbe avere qualsiasi valore e non apportare alcun problema.

Provate a osservare l'architettura e capire il perché.

Istruzione	RegDst	ALUSrc	MemtoReg	RegWrite	MemRead	MemWrite	Branch	ALUOp1	ALUOp0
Tipo R	1	0	0	1	0	0	0	1	0
`lw`	0	1	1	1	1	0	0	0	0
`sw`	X	1	X	0	0	1	0	0	0
`beq`	X	0	X	0	0	0	1	0	1

SAPIENZA
UNIVERSITÀ DI ROMA

Esercizio - JRR

Aggiungere alla CPU l'istruzione **jrr rs** (Jump Relative to Register)

di tipo R, che salta all'indirizzo (relativo al PC) contenuto nel registro **rs**

Ovvero che esegue come prossima istruzione quella che si trova all'indirizzo

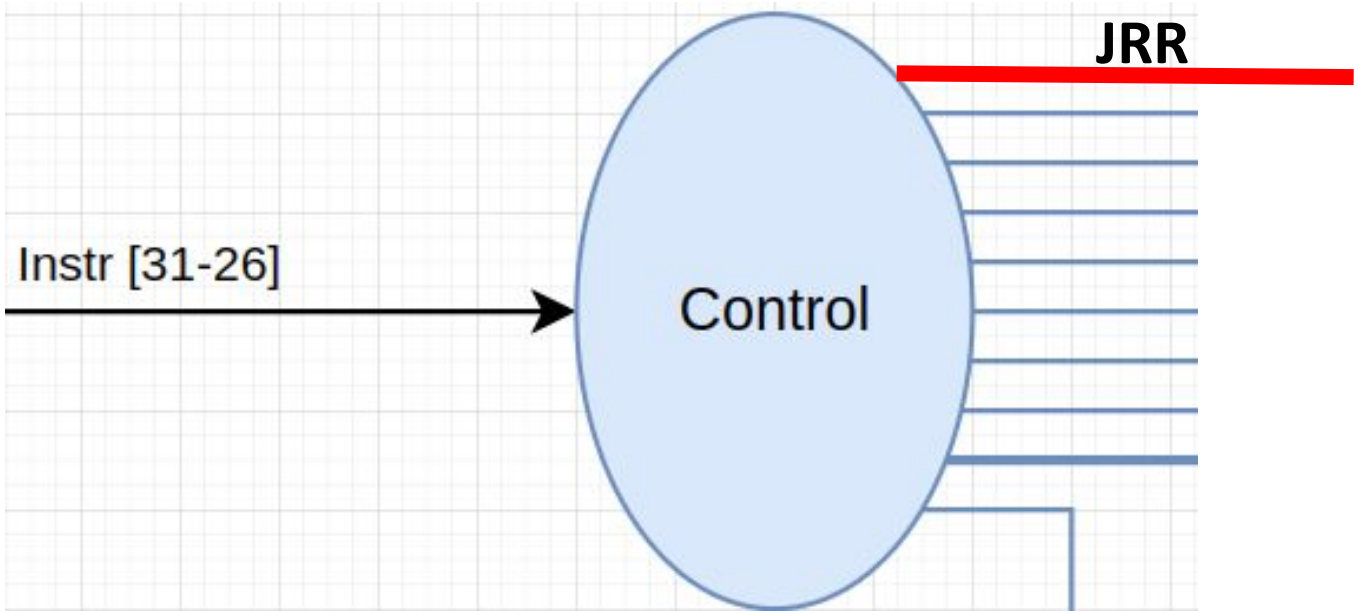
PC+4+Registri[rs]

a) Modificare lo schema per realizzare l'istruzione

b) Indicare tutti i segnali di controllo che la CU deve generare

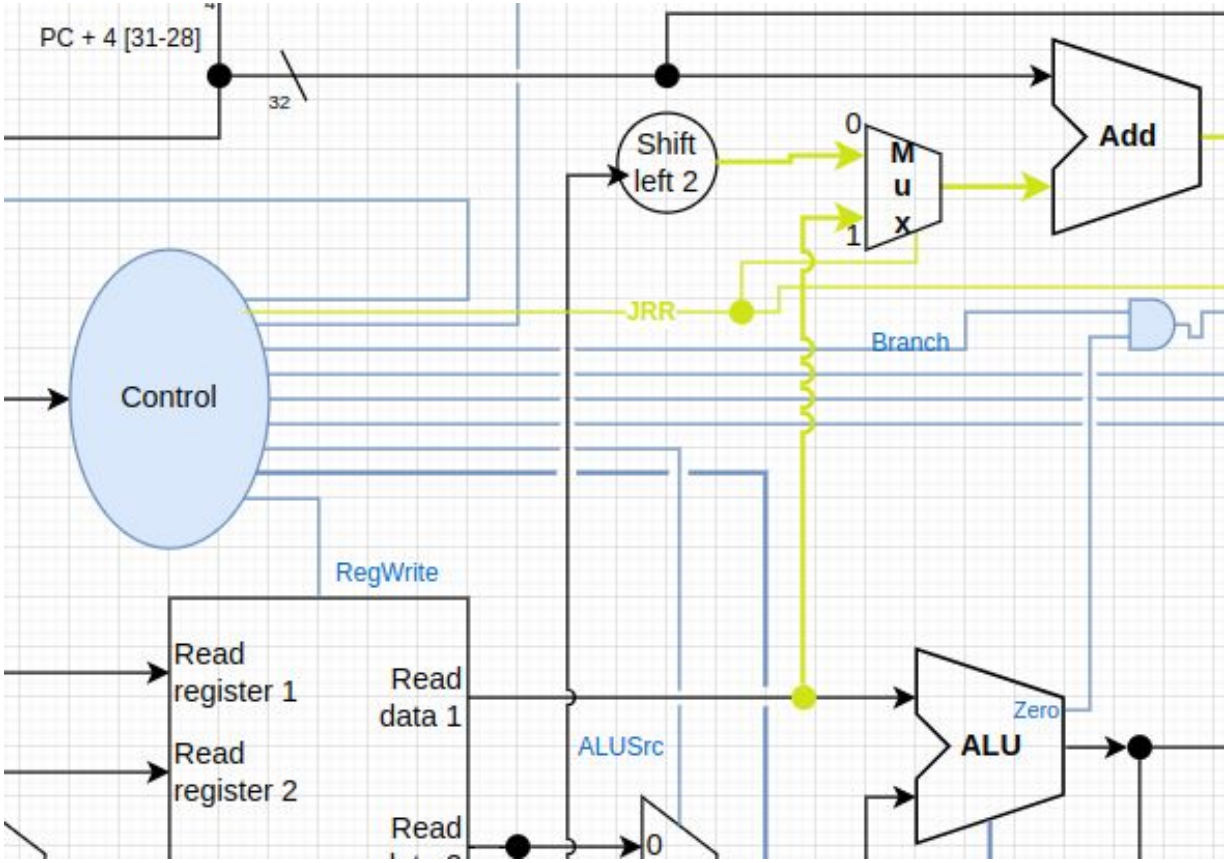
JRR - come ragionare?

Nel momento in cui stiamo progettando una nuova istruzione dobbiamo aggiungere un altro flag nella CU



JRR - come ragionare?

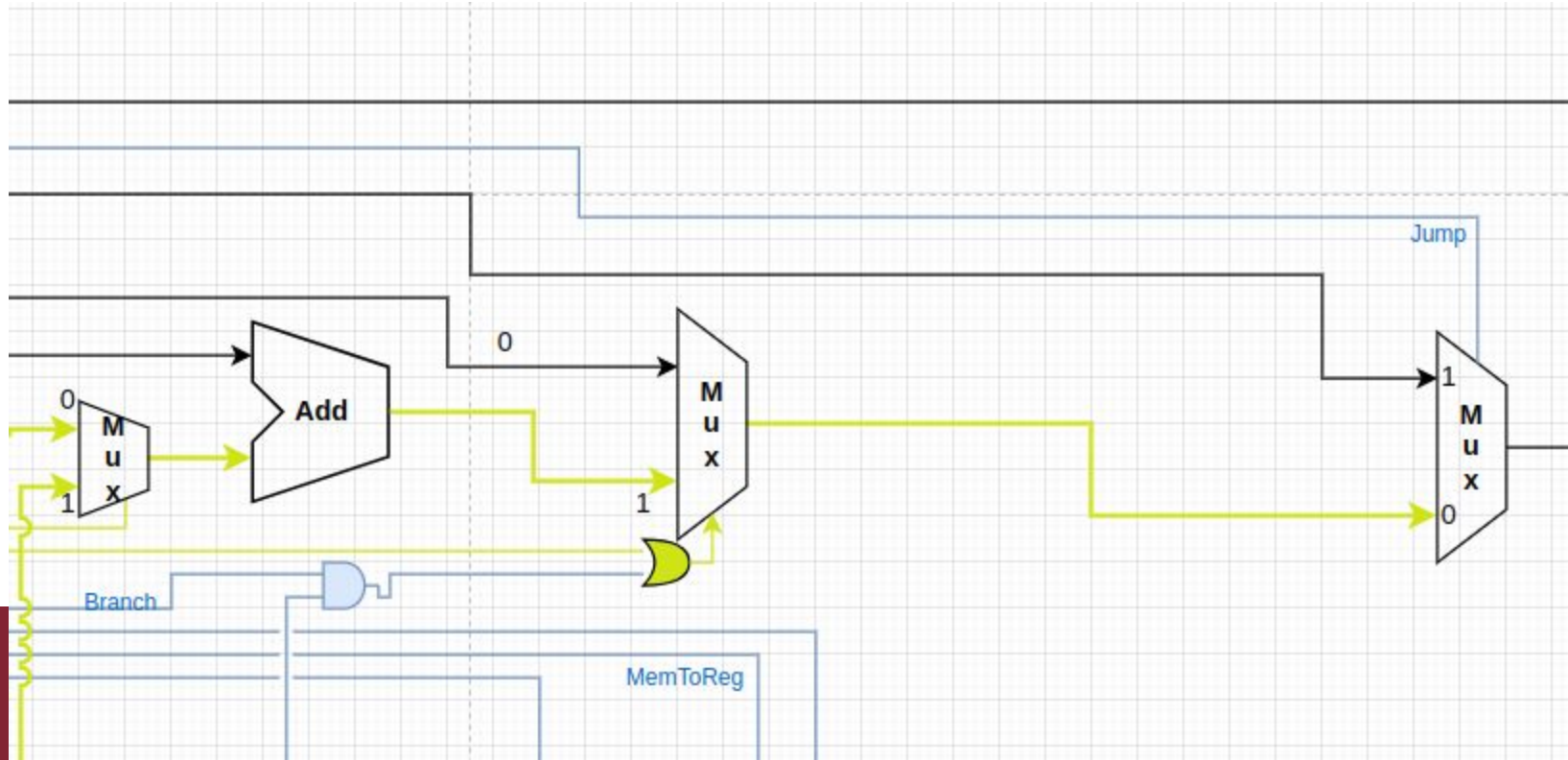
- Calcoliamo dove e se effettuare il branch.
- Per farlo biforchiamo l'output del contenuto del registro 1 e lo mandiamo all'adder
- Utilizziamo il MUX per scegliere quale indirizzo calcolare tramite il *flag* JRR



SAPIENZA
UNIVERSITÀ DI ROMA

JRR - come ragionare?

Con la nuova istruzione abbiamo una nuova condizione per effettuare un branch; perciò prenderemo il risultato di una **branch o JRR** solo nel caso uno dei 2 flag può essere eseguito.



SAPIENZA
UNIVERSITÀ DI ROMA

Cosa modificare dell'architettura?

- Aggiungere **flag alla CU**.
- Aggiungere **MUX** all'architettura per scegliere i dati corretti da passare.
- Aggiungere **porte logiche** per mantenere la correttezza e la consistenza dell'architettura.
- Si possono separare le connessioni per "copiare" il segnale e indirizzarlo dove necessario.

Ovviamente potrebbero essere necessari altri step nel caso l'istruzione da aggiungere sia più complessa o semplicemente li richieda.

SAPIENZA
UNIVERSITÀ DI ROMA

Esercizio 2 - aggiungere l'istruzione "StoreAtRegister"

Modifica dell'Architettura MIPS per Aggiungere l'Istruzione "StoreAtRegister":

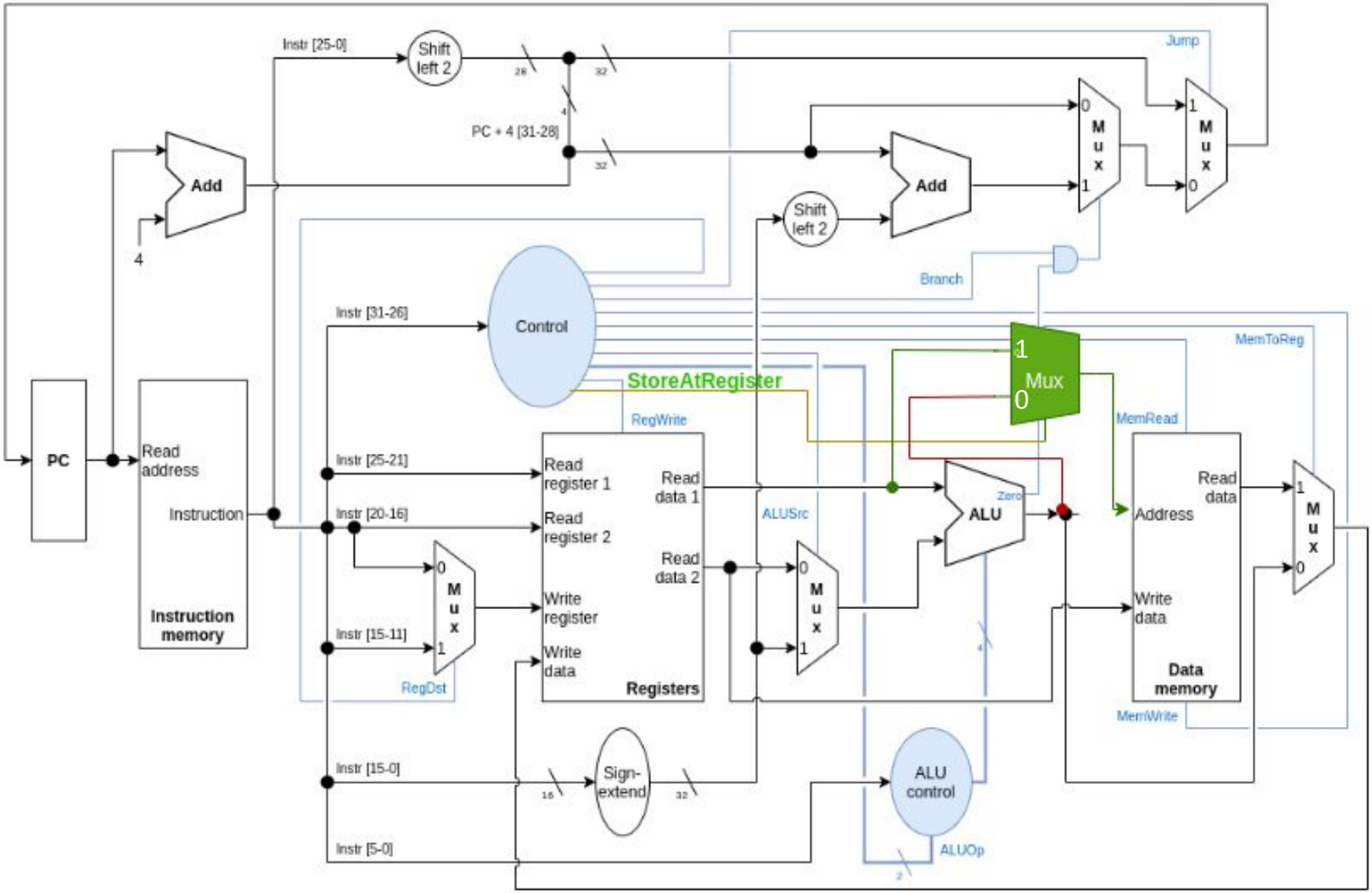
Modifica l'architettura MIPS per aggiungere l'istruzione "StoreAtRegister", che salva un dato contenuto nel registro 2 all'indirizzo in memoria specificato nel registro 1.

Procedimento:

- Aggiungere nuovo flag alla CU.
- Aggiungere logica di controllo.

SAPIENZA
UNIVERSITÀ DI ROMA

Esercizio 2 - soluzione



SAPIENZA
UNIVERSITÀ DI ROMA

Esercizio 3 - esame 15-06-2020

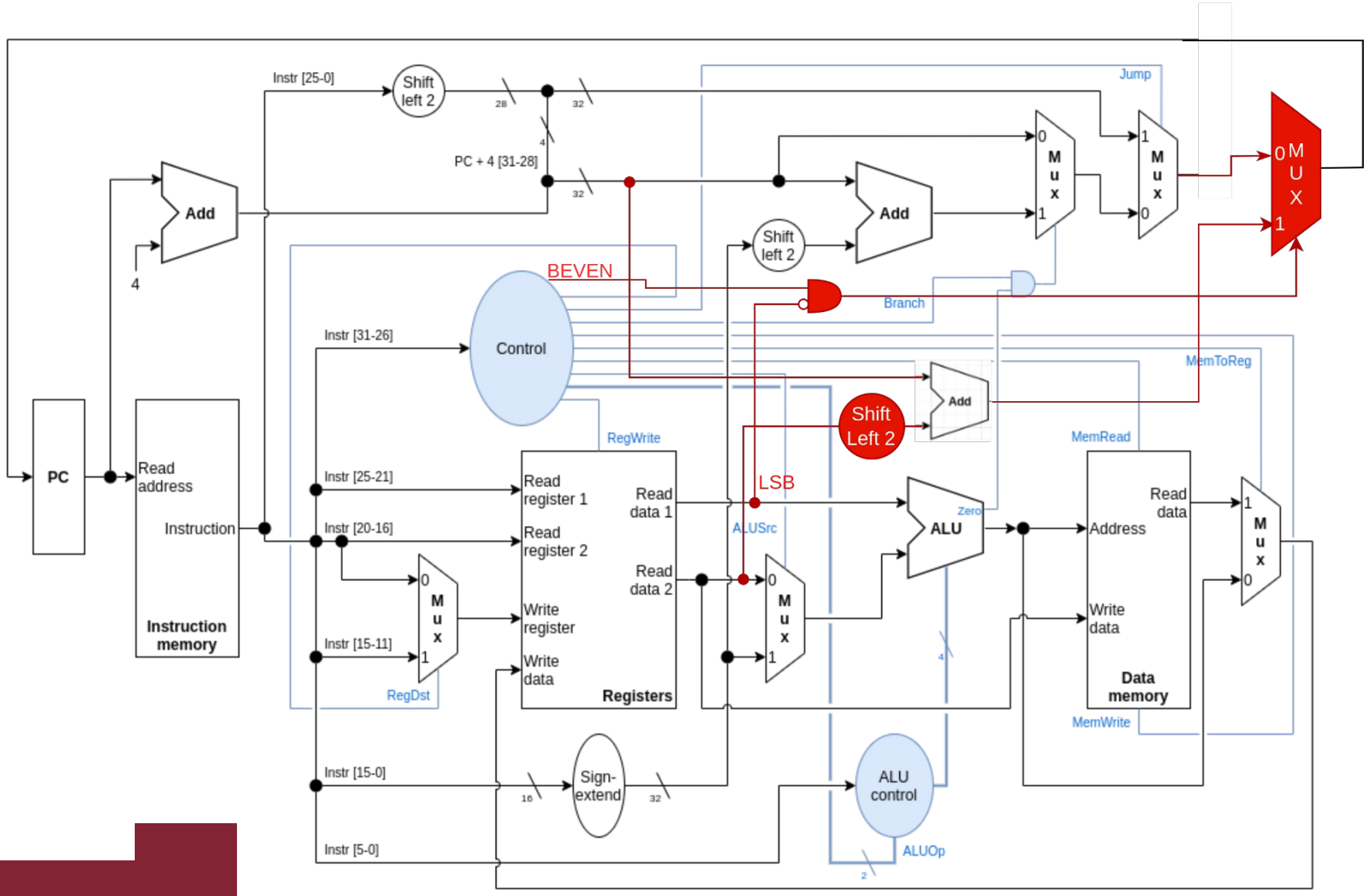
Si vuole aggiungere l'istruzione di tipo R **beven rs, rt (branch if even)** che esegua un salto relativo al PC (pre-incrementato di 4) con l'offset contenuto nel registro rt se e solo se il valore del registro **rs è pari**.

Si assuma che rt corrisponda al numero di istruzioni da saltare.

Hint: potete prendere il LSB separando il filo.

SAPIENZA
UNIVERSITÀ DI ROMA

Soluzione BEVEN



Beven - spiegazione

- Aggiunta nuova istruzione nella CU
- Il valore del secondo registro viene moltiplicato per 4 (SLL 2), perché le istruzioni sono grandi una word
- L' ALU calcola l'indirizzo su cui effettuare il branch e lo passa al MUX
- Con la porta AND controlliamo se le condizioni per eseguire BEVEN sono soddisfatte (beven == 1 && LSB r1 == 0), quindi indichiamo al MUX quale segnale risultante far passare (PC+4, branch, jump)

SAPIENZA
UNIVERSITÀ DI ROMA

Esercizio 4 - MAX

L'istruzione "MAX" prende due registri come input e restituisce il valore massimo tra i due.

Ad esempio, se i registri $t0 e $t1 contengono i valori 5 e 10 rispettivamente, l'istruzione "MAX $t2, $t0, $t1" dovrebbe impostare il registro $t2 al valore 10.

HINT:

Il testo precedente è invisibile, per leggerlo bisogna evidenziarlo con il cursore o fare copia incolla.

SAPIENZA
UNIVERSITÀ DI ROMA

Soluzione MAX