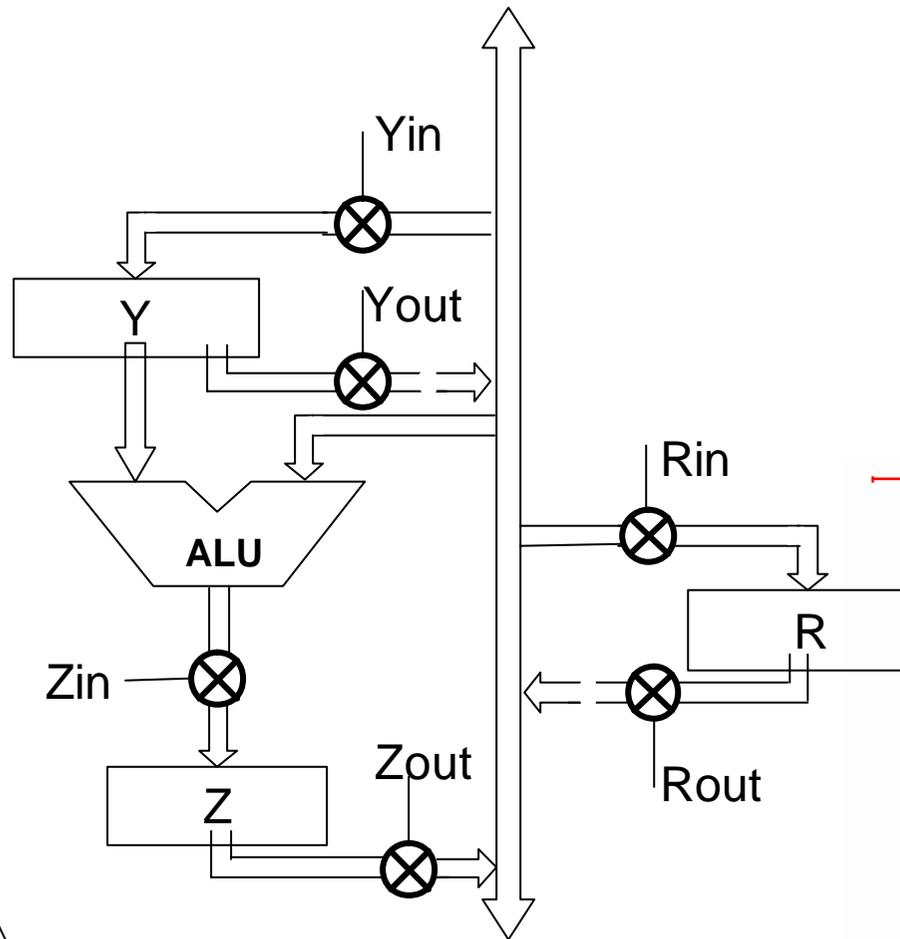


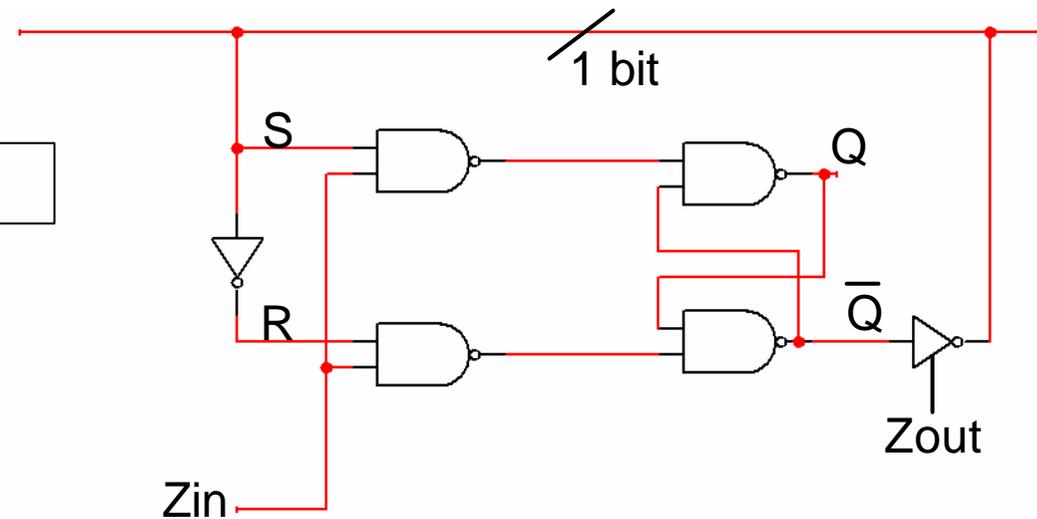
Trasferimento tra registri



I registri sono dotati di porte che abilitano il trasferimento dei dati da/verso il registro.

Le abilitazioni sono fornite da segnali provenienti dall'unità di controllo.

Le porte di uscita sono realizzate in tecnica *three-state*.



Esecuzione di trasferimento tra registri

- Per trasferire dati tra registri è necessario fornire le opportune abilitazioni.

Esempio:

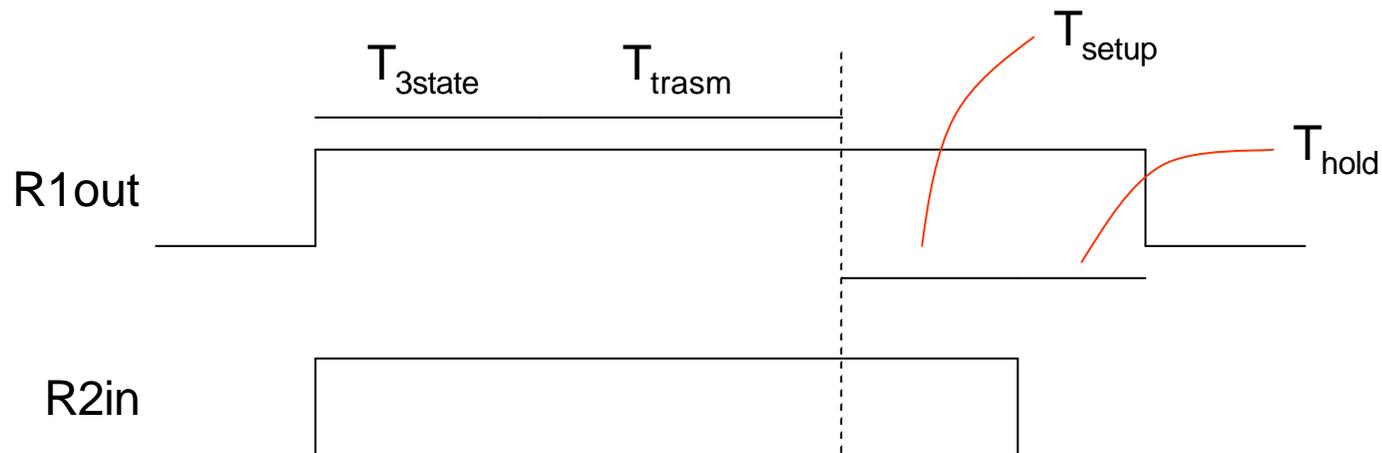
[R1] -> R2 \Rightarrow **R1out, R2in**

R1out (R1out=1)

abilita la porta di uscita di R1, ponendo il contenuto del registro sul bus comune.

R2in (R2in=1)

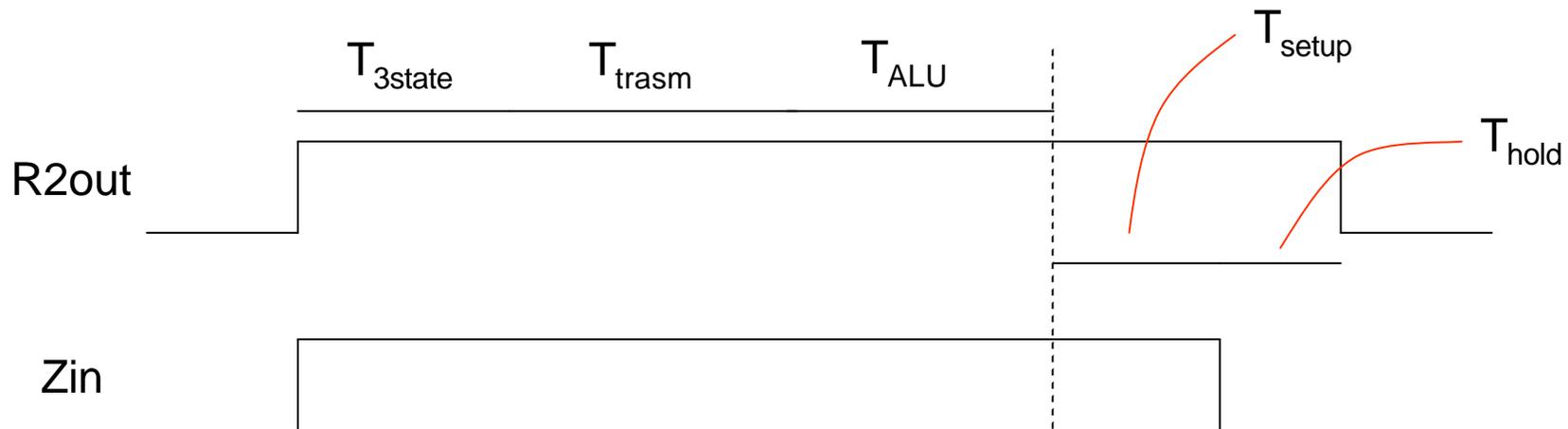
abilita la porta d'ingresso di R2, trasferendo nel registro il dato presente sul bus comune



Esecuzione di operazioni

- Per realizzare operazioni che coinvolgono l'ALU, sono necessari anche i controlli verso l'unità aritmetica.

Esempio:



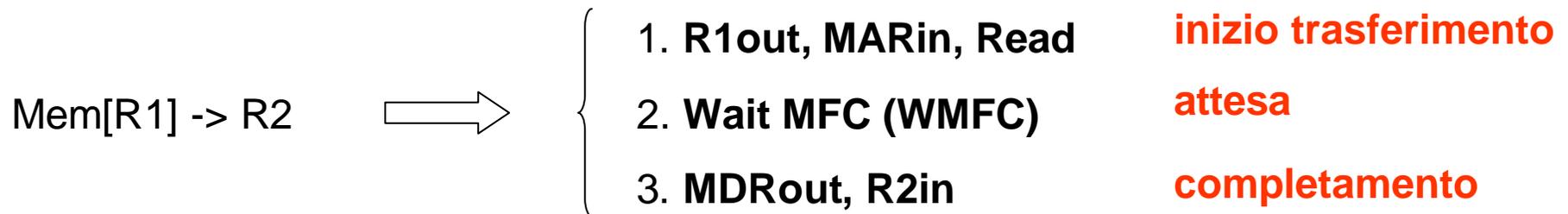
Trasferimento dati registri « memoria

- Il trasferimento dati tra un registro e la memoria centrale avviene tramite i registri MAR e MDR.
- Richiede la gestione di segnali di controllo verso il modulo di memoria e provenienti dal modulo di memoria.
- Due tipi di segnali di controllo:

Read/Write CPU ® memoria precisa il tipo di accesso da realizzare

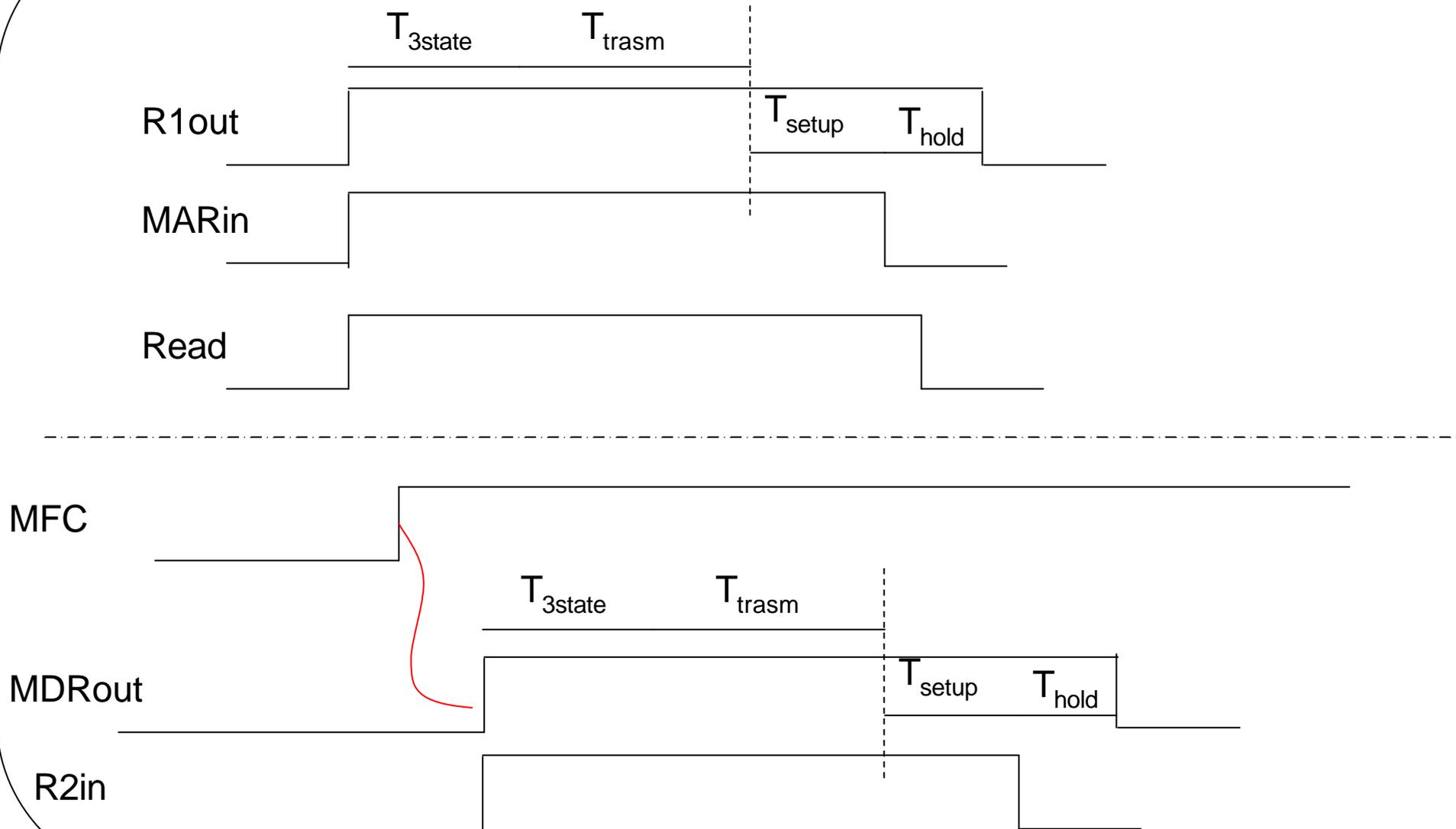
MFC CPU ⇐ memoria segnala la conclusione dell'operazione

Esempio:



 **trasferimento asincrono**

Tempificazione



Esecuzione di un'istruzione (1/2)

ADD (R3),R1

Instruction fetch:

Mem[PC] -> IR

[PC]+1 -> PC

Operand Assembly:

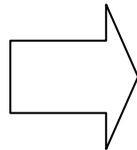
Mem[R3] -> Y

[R1] -> bus

Execution

[Y]+[bus] -> Z

[Z] -> R1



1. PCout, MARin, Read, Clear Y, Set Carry-in, Add, Zin
2. Zout, Pcin, WMFC
3. MDRout, IRin
4. R3out, MARin, Read
5. R1out, Yin, WMFC
6. MDRout, Add, Zin
7. Zout, R1in, End

Esecuzione di un'istruzione (2/2)

BMI disp

Instruction fetch:

Mem[PC] -> IR

[PC]+1 -> PC

Operand Assembly:

[PC] -> Y

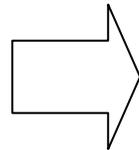
[disp_field] -> bus

Execution

if (N==1)

[Y]+[bus] -> Z

[Z] -> PC



1. PCout, MARin, Read, Clear Y, Set Carry-in, Add, Zin
2. Zout, Pcin, WMFC
3. MDRout, IRin
4. PCout, Yin, if (N==0) End
5. disp_field out, Add, Zin
6. Zout,PCin, End