

Cele 4 instructiuni aditionale

SLT (TIP R)

Binar : 000000 sssss ttttt ddddd 00000 000110

Asamblare : slt \$d , \$s , \$t;

RTL : $PC \leftarrow PC + 4$; if $\$s < \t then $\$d \leftarrow 1$ else $\$d \leftarrow 0$;

Descriere : Dacă $\$s < \t , \$d este inițializat cu 1, altfel cu 0;

Resurse :

Resurse necesare pe faze de execuție	
IF	PC, Memoria de instrucțiuni, sumator
ID/OF	Bloc de registre, <i>Unitatea principală de control</i>
EX	ALU, <i>Unitatea de control ALU</i>
MEM	No operation
WB	Bloc de registre

XOR (TIP R)

Binar : 000000 sssss ttttt ddddd 00000 000111

Asamblare : xor \$d, \$s, \$t;

RTL : $\$d \leftarrow \$s \wedge \$t$; $PC \leftarrow PC + 4$;

Descriere : SAU-Exclusiv logic între două registre, memorează rezultatul în alt registru

Resurse:

Resurse necesare pe faze de execuție	
IF	PC, Memoria de instrucțiuni, sumator
ID/OF	Bloc de registre, <i>Unitatea principală de control</i>
EX	ALU, <i>Unitatea de control ALU</i>
MEM	No operation
WB	Bloc de registre

ORI (TIP I)

Binar : 000101 sssss tttt iiiiiiiiiiiiii

Asamblare : ori \$t, \$s, imm;

RTL : $\$t \leftarrow \$s \mid ZE(\text{imm}); PC \leftarrow PC + 4;$

Descriere : SAU logic între un registru și o valoare imediată, memorează rezultatul în alt registru;

Resurse:

Resurse necesare	
IF	PC, Memoria de instrucțiuni, sumator
ID/OF	Bloc de registre, Unit. princ. de control, Extensie
EX	ALU, Unitatea de control ALU
MEM	No operation
WB	Bloc de registre

BGTZ (TIP I)

Binar : 000110 sssss 00000 oooooooooooooooooo

Asamblare : bgtz \$s, offset;

RTL : If \$s > 0 then PC <- (PC + 4) + (SE(offset) << 2) else PC <- PC + 4;

Descriere : Salt condiționat dacă un registru este mai mare ca 0;

Resurse :

Resurse necesare	
IF	PC, Memoria de instrucțiuni, sumator
ID/OF	Bloc de registre, Unit. princ. de control, Extensie
EX	ALU, Unitatea de control ALU, sumator, circuit deplasare, MUX
MEM	No operation
WB	No operation