

Configurare registre MIPS16 Pipeline – varianta 1

Se introduc pe coloane semnalele de date și control mapate la registre, de sus în jos, începând de la biții cei mai semnificativi ai registrului către cei mai puțin semnificativi. Se introduc în paranteză biții din registru alocați pentru fiecare semnal în parte. În dreptul numelui registrelor din primul rând se introduce în paranteză poziția bitului cel mai semnificativ (<msb>) din care reiese dimensiunea totală alocată registrului.

REG_IF_ID(31 – 0)	REG_ID_EX(83– 0>)	REG_EX_MEM(58 – 0)	REG_MEM_WB(36 – 0)
Instr(31 – 16)	RegDst(83)	br_ne(58)	RegWrite(36)
PC + 1(15 – 0)(PCinc)	ALUSrc(82)	br_gez(57)	MemtoReg(35)
	Branch(81)	Branch(56)	ALUResOut(34-19)
	br_gez(80)	MemWrite(55)	MemData(18-3)
	br_ne(79)	MemtoReg(54)	rWA(2-0)
	ALUOp(78-77)	RegWrite(53)	
	MemWrite(76)	Zero(52)	
	MemtoReg(75)	gez(51)	
	RegWrite(74)	BranchAddress(50-35)	
	RD1(73-58)	ALURes(34-19)	
	RD2(57-42)	rWA(18-16)	
	Ext_Imm(41-26)	RD2(15-0)	
	func(25-23)		
	sa(22)		
	rd(21-19)		
	rt(18-16)		
	PC+1(15-0)		

URL: <https://drive.google.com/file/d/1Yw18tfjFjo-v897vVpd8KNGnpx5U8i9n/view?usp=sharing>

Configurare registre MIPS16 Pipeline – varianta 2

Se introduc pe coloane numele utilizate în codul VHDL pentru semnalele de date și control implementate ca registre, pe categorii. Se introduce în paranteză dimensiunea în biți.

IF/ID	ID/EX	EX/MEM	MEM/WB
Instruction_IF_ID(16)	RegDst_ID_EX(1)	Branch_EX_MEM(1)	RegWr_MEM_WB(1)
PC_1_IF_ID(16)	<nume_semnal>(<dimensiune>)
	...		

URL: <https://drive.google.com/file/d/1Yw18tfjFjo-v897vVpd8KNGnpx5U8i9n/view?usp=sharing>