

# INFORMAÇÃO ESSENCIAL PARA COMPREENDER A VERSÃO *MULTI-CYCLE* DO *DATAPATH* (SIMPLIFICADO) DO MIPS

TOMÁS OLIVEIRA E SILVA, DEZEMBRO DE 2010

## FORMATO DAS INSTRUÇÕES

	6 bits	5 bits	5 bits	5 bits	5 bits	6 bits
<b>Tipo R</b>	opcode (0)	rs	rt	rd	shamt	funct
	bits 31:26	bits 25:21	bits 20:16	bits 15:11	bits 10:6	bits 5:0

Exemplo: add \$rd,\$rs,\$rt

	6 bits	5 bits	5 bits	16 bits
<b>Tipo I</b>	opcode	rs	rt	Imm
	bits 31:26	bits 25:21	bits 20:16	bits 15:0

Exemplos: addi \$rt,\$rs,Imm  
 lw \$rt,Imm(\$rs)  
 sw \$rt,Imm(\$rs)  
 beq \$rs,\$rt,Label # o valor imediato é dado por  $(\text{Label} - (\text{PC} + 4))/4$

	6 bits	26 bits
<b>Tipo J</b>	opcode	Imm
	bits 31:26	bits 25:0

Exemplos: j Label # o valor imediato é dado pelos 26 bits menos significativos de Label/4  
 jal Label # os quatro bits mais significativos de PC + 4 e de Label têm de ser iguais

## SIGNIFICADO DO SINAL ALUOP

valor (em binário)	00	01	10	11
operação efectuada pela ALU	add	sub	definida pelo campo funct	stli

## FASES DE EXECUÇÃO

Nome/Tipo da instrução	Instruction Fetch (IF)	Instruction Decode (ID)	Execute (EX)	Memory Access (MEM)	Write Back (WB)
Tipo R	Fase 1	Fase 2	Fase 3		Fase 4
	Lê instrução da memória; Soma 4 ao PC	Decodifica a instrução; Lê registos; Calcula BTA	Efectua a operação definida pelo campo funct		Escreve registo
sw	Fase 1	Fase 2	Fase 3	Fase 4	
	Idem	Idem	Calcula endereço	Escreve na memória	
lw	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5
	Idem	Idem	Calcula endereço	Lê da memória	Escreve registo
bne	Fase 1	Fase 2	Fase 3		
	Idem	Idem	Compara (subtrai); Se o resultado não for zero transfere BTA para PC		

Nota: BTA é a sigla de *Branch Target Address*.