

# MIPS Instruction Reference

## Arithmetic and Logical Instructions

Instruction	Opcode/Function	Syntax	Operation
add	100000	f \$d, \$s, \$t	$\$d = \$s + \$t$
addu	100001	f \$d, \$s, \$t	$\$d = \$s + \$t$
addi	001000	f \$d, \$s, i	$\$d = \$s + \text{SE}(i)$
addiu	001001	f \$d, \$s, i	$\$d = \$s + \text{SE}(i)$
and	100100	f \$d, \$s, \$t	$\$d = \$s \& \$t$
andi	001100	f \$d, \$s, i	$\$t = \$s \& \text{ZE}(i)$
div	011010	f \$s, \$t	$\text{lo} = \$s / \$t; \text{hi} = \$s \% \$t$
divu	011011	f \$s, \$t	$\text{lo} = \$s / \$t; \text{hi} = \$s \% \$t$
mult	011000	f \$s, \$t	$\text{hi:lo} = \$s * \$t$
multu	011001	f \$s, \$t	$\text{hi:lo} = \$s * \$t$
nor	100111	f \$d, \$s, \$t	$\$d = \sim(\$s   \$t)$
or	100101	f \$d, \$s, \$t	$\$d = \$s   \$t$
ori	001101	f \$d, \$s, i	$\$t = \$s   \text{ZE}(i)$
sll	000000	f \$d, \$t, a	$\$d = \$t \ll a$
sllv	000100	f \$d, \$t, \$s	$\$d = \$t \ll \$s$
sra	000011	f \$d, \$t, a	$\$d = \$t \gg a$
srav	000111	f \$d, \$t, \$s	$\$d = \$t \gg \$s$
srl	000010	f \$d, \$t, a	$\$d = \$t \ggg a$
srlv	000110	f \$d, \$t, \$s	$\$d = \$t \ggg \$s$
sub	100010	f \$d, \$s, \$t	$\$d = \$s - \$t$
subu	100011	f \$d, \$s, \$t	$\$d = \$s - \$t$
xor	100110	f \$d, \$s, \$t	$\$d = \$s \wedge \$t$
xori	001110	f \$d, \$s, i	$\$d = \$s \wedge \text{ZE}(i)$

## Constant-Manipulating Instructions

Instruction	Opcode/Function	Syntax	Operation
lhi	011001	o \$t, immed32	HH (\$t) = i
llo	011000	o \$t, immed32	LH (\$t) = i

## Comparison Instructions

Instruction	Opcode/Function	Syntax	Operation
slt	101010	f \$d, \$s, \$t	\$d = (\$s < \$t)
sltu	101001	f \$d, \$s, \$t	\$d = (\$s < \$t)
slti	001010	f \$d, \$s, i	\$t = (\$s < SE(i))
sltiu	001001	f \$d, \$s, i	\$t = (\$s < SE(i))

## Branch Instructions

Instruction	Opcode/Function	Syntax	Operation
beq	000100	o \$s, \$t, label	if (\$s == \$t) pc += i << 2
bgtz	000111	o \$s, label	if (\$s > 0) pc += i << 2
blez	000110	o \$s, label	if (\$s <= 0) pc += i << 2
bne	000101	o \$s, \$t, label	if (\$s != \$t) pc += i << 2

## Jump Instructions

Instruction	Opcode/Function	Syntax	Operation
j	000010	o label	pc += i << 2
jal	000011	o label	\$31 = pc; pc += i << 2
jalr	001001	o labelR	\$31 = pc; pc = \$s
jr	001000	o labelR	pc = \$s

## Load Instructions

Instruction	Opcode/Function	Syntax	Operation
lb	100000	o \$t, i (\$s)	\$t = SE (MEM [\$s + i]:1)
lbu	100100	o \$t, i (\$s)	\$t = ZE (MEM [\$s + i]:1)
lh	100001	o \$t, i (\$s)	\$t = SE (MEM [\$s + i]:2)
lhu	100101	o \$t, i (\$s)	\$t = ZE (MEM [\$s + i]:2)
lw	100011	o \$t, i (\$s)	\$t = MEM [\$s + i]:4

## Store Instructions

Instruction	Opcode/Function	Syntax	Operation
sb	101000	o \$t, i (\$s)	MEM [\$s + i]:1 = LB (\$t)
sh	101001	o \$t, i (\$s)	MEM [\$s + i]:2 = LH (\$t)

sw	101011	o \$t, i (\$s)	MEM [\$s + i]:4 = \$t
----	--------	----------------	-----------------------

## Data Movement Instructions

Instruction	Opcode/Function	Syntax	Operation
mfhi	010000	f \$d	\$d = hi
mflo	010010	f \$d	\$d = lo
mthi	010001	f \$s	hi = \$s
mtlo	010011	f \$s	lo = \$s

## Exception and Interrupt Instructions

Instruction	Opcode/Function	Syntax	Operation
trap	011010	o i	Dependent on OS; different values for immed26 specify different operations.