

MIPS Instruction Set (core)

<i>instruction</i>	<i>example</i>	<i>meaning</i>
arithmetic		
add	add \$1,\$2,\$3	$\$1 = \$2 + \$3$
subtract	sub \$1,\$2,\$3	$\$1 = \$2 - \$3$
add immediate	addi \$1,\$2,100	$\$1 = \$2 + 100$
add unsigned	addu \$1,\$2,\$3	$\$1 = \$2 + \$3$
subtract unsigned	subu \$1,\$2,\$3	$\$1 = \$2 - \$3$
add immediate unsigned	addiu \$1,\$2,100	$\$1 = \$2 + 100$
set if less than	slt \$1, \$2, \$3	if ($\$2 < \3), $\$1 = 1$ else $\$1 = 0$
set if less than immediate	slti \$1, \$2, 100	if ($\$2 < 100$), $\$1 = 1$ else $\$1 = 0$
set if less than unsigned	sltu \$1, \$2, \$3	if ($\$2 < \3), $\$1 = 1$ else $\$1 = 0$
set if < immediate unsigned	sltui \$1, \$2, 100	if ($\$2 < 100$), $\$1 = 1$ else $\$1 = 0$
multiply	mult \$2,\$3	Hi, Lo = $\$2 * \3 , 64-bit signed product
multiply unsigned	multu \$2,\$3	Hi, Lo = $\$2 * \3 , 64-bit unsigned product
divide	div \$2,\$3	Lo = $\$2 / \3 , Hi = $\$2 \bmod \3
divide unsigned	divu \$2,\$3	Lo = $\$2 / \3 , Hi = $\$2 \bmod \3 , unsigned
transfer		
move from Hi	mfhi \$1	$\$1 = \text{Hi}$
move from Lo	mflo \$1	$\$1 = \text{Lo}$
load upper immediate	lui \$1,100	$\$1 = 100 \times 2^{16}$
logic		
and	and \$1,\$2,\$3	$\$1 = \$2 \& \$3$
or	or \$1,\$2,\$3	$\$1 = \$2 \$3$
and immediate	andi \$1,\$2,100	$\$1 = \$2 \& 100$
or immediate	ori \$1,\$2,100	$\$1 = \$2 100$
nor	nor \$1,\$2,\$3	$\$1 = \text{not}(\$2 \$3)$
xor	xor \$1, \$2, \$3	$\$1 = \$2 \oplus \$3$
xor immediate	xori \$1, \$2, 255	$\$1 = \$2 \oplus 255$
shift		
shift left logical	sll \$1,\$2,5	$\$1 = \$2 \ll 5$ (logical)
shift left logical variable	sllv \$1,\$2,\$3	$\$1 = \$2 \ll \$3$ (logical), variable shift amt
shift right logical	srl \$1,\$2,5	$\$1 = \$2 \gg 5$ (logical)
shift right logical variable	srlv \$1,\$2,\$3	$\$1 = \$2 \gg \$3$ (logical), variable shift amt
shift right arithmetic	sra \$1,\$2,5	$\$1 = \$2 \gg 5$ (arithmetic)
shift right arithmetic variable	srav \$1,\$2,\$3	$\$1 = \$2 \gg \$3$ (arithmetic), variable shift amt
memory		
load word	lw \$1, 1000(\$2)	$\$1 = \text{memory} [\$2+1000]$
store word	sw \$1, 1000(\$2)	$\text{memory} [\$2+1000] = \1
load byte	lb \$1, 1002(\$2)	$\$1 = \text{memory} [\$2+1002]$ in least sig. byte
load byte unsigned	lbu \$1, 1002(\$2)	$\$1 = \text{memory} [\$2+1002]$ in least sig. byte
store byte	sb \$1, 1002(\$2)	$\text{memory} [\$2+1002] = \1 (byte modified only)
branch		
branch if equal	beq \$1,\$2,100	if ($\$1 = \2), $\text{PC} = \text{PC} + 4 + (100*4)$
branch if not equal	bne \$1,\$2,100	if ($\$1 \neq \2), $\text{PC} = \text{PC} + 4 + (100*4)$
jump		
jump	j 10000	$\text{PC} = 10000*4$
jump register	jr \$31	$\text{PC} = \$31$
jump and link	jal 10000	$\$31 = \text{PC} + 4$; $\text{PC} = 10000*4$

Last updated 28 October 2005