

OpType	Type	Syntax	Binary	Remark
Arithmetic	R	add \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 100000	
	R	addu \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 100001	
	R	sub \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 100010	
	R	subu \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 100011	
	I	addi \$t,\$s,C	001000 sssss ttttt CCCCC CCCCC CCCCC	
	I	addiu \$t,\$s,C	001001 sssss ttttt CCCCC CCCCC CCCCC	
	R	mult \$s,\$t	000000 sssss ttttt ddddd 00000 011000	(HI,LO) = (64-bit) \$s * \$t
	R	multu \$s,\$t	000000 sssss ttttt ddddd 00000 011001	(HI,LO) = (64-bit) \$s * \$t
	R	div \$s,\$t	000000 sssss ttttt ddddd 00000 011010	LO = \$s / \$t, HI = \$s % \$t
	R	divu \$s,\$t	000000 sssss ttttt ddddd 00000 011011	LO = \$s / \$t, HI = \$s % \$t
Logical	R	and \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 100100	
	I	andi \$t,\$s,C	001100 sssss ttttt CCCCC CCCCC CCCCC	
	R	or \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 100101	
	I	ori \$t,\$s,C	001101 sssss ttttt CCCCC CCCCC CCCCC	
	R	xor \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 100110	
	R	nor \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 100111	
	R	slt \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 101010	
	R	sltu \$d,\$s,\$t	000000 sssss ttttt ddddd 00000 101011	
	I	slti \$t,\$s,C	001010 sssss ttttt CCCCC CCCCC CCCCC	
Bitwise Shift	R	sll \$d,\$t,shamt	000000 sssss ttttt ddddd 00000 000000	\$d = \$t << shamt
	R	srl \$d,\$t,shamt	000000 sssss ttttt ddddd 00000 000010	\$d = {16'b0, \$t >> shamt}
	R	sra \$d,\$t,shamt	000000 sssss ttttt ddddd 00000 000011	\$d = {{16{t[31]}}, \$t >> shamt}
	R	sllv \$d,\$t,\$s	000000 sssss ttttt ddddd 00000 000100	\$d = \$t << \$s
	R	srlv \$d,\$t,\$s	000000 sssss ttttt ddddd 00000 000110	\$d = {16'b0, \$t >> \$s}
	R	srav \$d,\$t,\$s	000000 sssss ttttt ddddd 00000 000111	\$d = {{16{t[31]}}, \$t >> \$s}
Data Transfer	I	lw \$t,C(\$s)	100011 sssss ttttt CCCCC CCCCC CCCCC	
	I	lh \$t,C(\$s)	100001 sssss ttttt CCCCC CCCCC CCCCC	
	I	lhu \$t,C(\$s)	100101 sssss ttttt CCCCC CCCCC CCCCC	
	I	lb \$t,C(\$s)	100000 sssss ttttt CCCCC CCCCC CCCCC	
	I	lbu \$t,C(\$s)	100100 sssss ttttt CCCCC CCCCC CCCCC	
	I	sw \$t,C(\$s)	101011 sssss ttttt CCCCC CCCCC CCCCC	
	I	sh \$t,C(\$s)	101001 sssss ttttt CCCCC CCCCC CCCCC	
	I	sb \$t,C(\$s)	101000 sssss ttttt CCCCC CCCCC CCCCC	
	I	lui \$t,C	001111 00000 ttttt CCCCC CCCCC CCCCC	
	R	mfhi \$d	000000 00000 00000 ddddd 00000 010000	
	R	mflo \$d	000000 00000 00000 ddddd 00000 010010	
Conditional branch	I	beq \$s,\$t,C	000100 sssss ttttt CCCCC CCCCC CCCCC	
	I	bne \$s,\$t,C	000101 sssss ttttt CCCCC CCCCC CCCCC	
Unconditional jump	J	j C	000010 CCCCC CCCCC CCCCC CCCCC CCCCC	addr = {PC_plus_4[31:28], C << 2}
	R	jr \$s	000000 sssss 00000 00000 00000 001000	
	J	jal C	000011 CCCCC CCCCC CCCCC CCCCC CCCCC	addr = {PC_plus_4[31:28], C << 2}