MIPS Instruction “Subset” Reference

Name	Mnemonic	Syntax	Operation	Format	Opcode/Funct
Load Word	lw	lw $rt, i($rs)	$rt = Mem[$rs + i]	I	23_{16}
Store Word	sw	sw $rt, i($rs)	Mem[$rs + i] = $rt	I	$2b_{16}$
Add	add	add $rd, $rs, $rt	$rd = $rs + $rt	R	$0/20_{16}$
Sub	sub	sub $rd, $rs, $rt	$rd = $rs - $rt	R	$0/22_{16}$
And	and	and $rd, $rs, $rt	$rd = $rs & $rt	R	$0/24_{16}$
Or	or	or $rd, $rs, $rt	$rd = $rs | $rt	R	$0/25_{16}$
Set Less Than	slt	slt $rd, $rs, $rt	$rd = ($rs < $rt)?1:0	R	$0/2a_{16}$
Branch On Equal	beq	beq $rs, $rt, label	if($rs == $rt) pc+=addr<<2	I	5_{16}
Jump	j	j label	pc = addr<<2	J	2_{16}

REGISTER NAME, NUMBER, USE, CALL CONVENTION

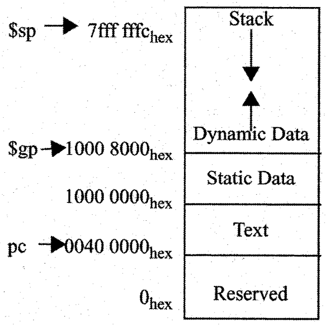
NAME	NUMBER	USE	PRESERVED ACROSS A CALL?
$zero	0	The Constant Value 0	N.A.
$at	1	Assembler Temporary	No
$v0-$v1	2-3	Values for Function Results and Expression Evaluation	No
$a0-$a3	4-7	Arguments	No
$t0-$t7	8-15	Temporaries	No
$s0-$s7	16-23	Saved Temporaries	Yes
$t8-$t9	24-25	Temporaries	No
$k0-$k1	26-27	Reserved for OS Kernel	No
$gp	28	Global Pointer	Yes
$sp	29	Stack Pointer	Yes
$fp	30	Frame Pointer	Yes
$ra	31	Return Address	Yes

(1) May cause overflow exception
(2) SignExtImm = { 16{immediate[15]}, immediate }
(3) ZeroExtImm = { 16{1b’0}, immediate }
(4) BranchAddr = { 14{immediate[15]}, immediate, 2’b0 }
(5) JumpAddr = { PC+4[31:28], address, 2’b0 }
(6) Operands considered unsigned numbers (vs. 2’s comp.)
(7) Atomic test&set pair; R[rt] = 1 if pair atomic, 0 if not atomic

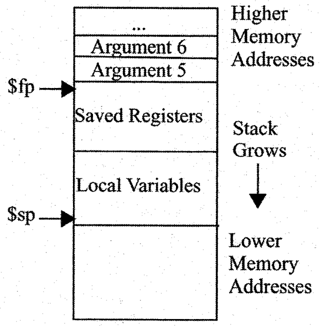
BASIC INSTRUCTION FORMATS

R	opcode	rs	rt	rd	shamt	funct
	31 26	25 21	20 16	15 11	10 6	5 0
I	opcode	rs	rt	immediate		
	31 26	25 21	20 16	15 0		
J	opcode	address				
	31 26	25 0				

MEMORY ALLOCATION



STACK FRAME



DATA ALIGNMENT

Double Word							
Word				Word			
Halfword		Halfword		Halfword		Halfword	
Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte
0	1	2	3	4	5	6	7

Value of three least significant bits of byte address (Big Endian)