

Format of an Assembly command line

31:24	23:20	19:16	15:12	11:0
Opcode	rd	rs	rt	immediate

Registers table

Register Number	Register Name	Purpose
0	\$zero	Constant zero
1	\$imm	Sign extended immediate
2	\$v0	Result value
3	\$a0	Argument register
4	\$a1	Argument register
5	\$t0	Temporary register
6	\$t1	Temporary register
7	\$t2	Temporary register
8	\$t3	Temporary register
9	\$s0	Saved register
10	\$s1	Saved register
11	\$s2	Saved register
12	\$gp	Global pointer (static data)
13	\$sp	Stack pointer
14	\$fp	Frame Pointer
15	\$ra	Return address

Commands table

Opcode Number	Name	Meaning
0	add	$R[rd] = R[rs] + R[rt]$
1	sub	$R[rd] = R[rs] - R[rt]$
2	and	$R[rd] = R[rs] \& R[rt]$
3	or	$R[rd] = R[rs] R[rt]$
4	sll	$R[rd] = R[rs] \ll R[rt]$
5	sra	$R[rd] = R[rs] \ggg R[rt]$, arithmetic shift with sign extension
6	srl	$R[rd] = R[rs] \gg R[rt]$, logical shift
7	beq	if ($R[rs] == R[rt]$) $pc = R[rd]$
8	bne	if ($R[rs] != R[rt]$) $pc = R[rd]$
9	blt	if ($R[rs] < R[rt]$) $pc = R[rd]$
10	bgt	if ($R[rs] > R[rt]$) $pc = R[rd]$
11	ble	if ($R[rs] \leq R[rt]$) $pc = R[rd]$
12	bge	if ($R[rs] \geq R[rt]$) $pc = R[rd]$
13	jal	$R[15] = pc + 1$ (next instruction address), $pc = R[rd]$
14	lw	$R[rd] = \text{MEM}[R[rs] + R[rt]]$
15	sw	$\text{MEM}[R[rs] + R[rt]] = R[rd]$
16		Reserved for future use
17		Reserved for future use
18		Reserved for future use
19	halt	Halt execution, exit simulator