Computer Architecture Note

MIPS Instruction Opcode

Mnemonic	Meaning	Type	Opcode	Funct
`add`	Add	R	0x00	0x20
`addi`	Add Immediate	I	0x08	NA
`addiu`	Add Unsigned Immediate	I	0x09	NA
`addu`	Add Unsigned	R	0x00	0x21
`and`	Bitwise AND	R	0x00	0x24
`andi`	Bitwise AND Immediate	I	0x0C	NA
`beq`	Branch if Equal	I	0x04	NA
`blez`	Branch if Less Than or Equal to Zero	I	0x06	NA
`bne`	Branch if Not Equal	I	0x05	NA
`bgtz`	Branch on Greater Than Zero	I	0x07	NA
`div`	Divide	R	0x00	0x1A
`divu`	Unsigned Divide	R	0x00	0x1B
`j`	Jump to Address	J	0x02	NA
`jal`	Jump and Link	J	0x03	NA
`jalr`	Jump and Link Register	R	0x00	0x09
`jr`	Jump to Address in Register	R	0x00	0x08
`lb`	Load Byte	I	0x20	NA
`lbu`	Load Byte Unsigned	I	0x24	NA
`lhu`	Load Halfword Unsigned	I	0x25	NA
`lui`	Load Upper Immediate	I	0x0F	NA
`lw`	Load Word	I	0x23	NA

`mfhi`	Move from HI Register	R	0x00	0x10
`mthi`	Move to HI Register	R	0x00	0x11
`mflo`	Move from LO Register	R	0x00	0x12
`mtlo`	Move to LO Register	R	0x00	0x13
`mfc0`	Move from Coprocessor 0	R	0x10	NA
`mult`	Multiply	R	0x00	0x18
`multu`	Unsigned Multiply	R	0x00	0x19
`nor`	Bitwise NOR (NOT-OR)	R	0x00	0x27
`xor`	Bitwise XOR (Exclusive-OR)	R	0x00	0x26
`or`	Bitwise OR	R	0x00	0x25
`ori`	Bitwise OR Immediate	I	0x0D	NA
`sb`	Store Byte	I	0x28	NA
`sh`	Store Halfword	I	0x29	NA
`slt`	Set to 1 if Less Than	R	0x00	0x2A
`slti`	Set to 1 if Less Than Immediate	I	0x0A	NA
`sltiu`	Set to 1 if Less Than Unsigned Immediate	I	0x0B	NA
`sltu`	Set to 1 if Less Than Unsigned	R	0x00	0x2B
`sll`	Logical Shift Left	R	0x00	0x00
`srl`	Logical Shift Right (0-extended)	R	0x00	0x02
`sra`	Arithmetic Shift Right (sign-extended)	R	0x00	0x03
`sub`	Subtract	R	0x00	0x22
`subu`	Unsigned Subtract	R	0x00	0x23
`sw`	Store Word	I	0x2B	NA

Register Table

register	assembly name	Comment
r0	$zero	Always 0
r1	$at	Reserved for assembler
r2-r3	$v0-$v1	Stores results
r4-r7	$a0-$a3	Stores arguments
r8-r15	$t0-$t7	Temporaries, not saved
r16-r23	$s0-$s7	Contents saved for use later
r24-r25	$t8-$t9	More temporaries, not saved
r26-r27	$k0-$k1	Reserved by operating system
r28	$gp	Global pointer
r29	$sp	Stack pointer
r30	$fp	Frame pointer
r31	$ra	Return address