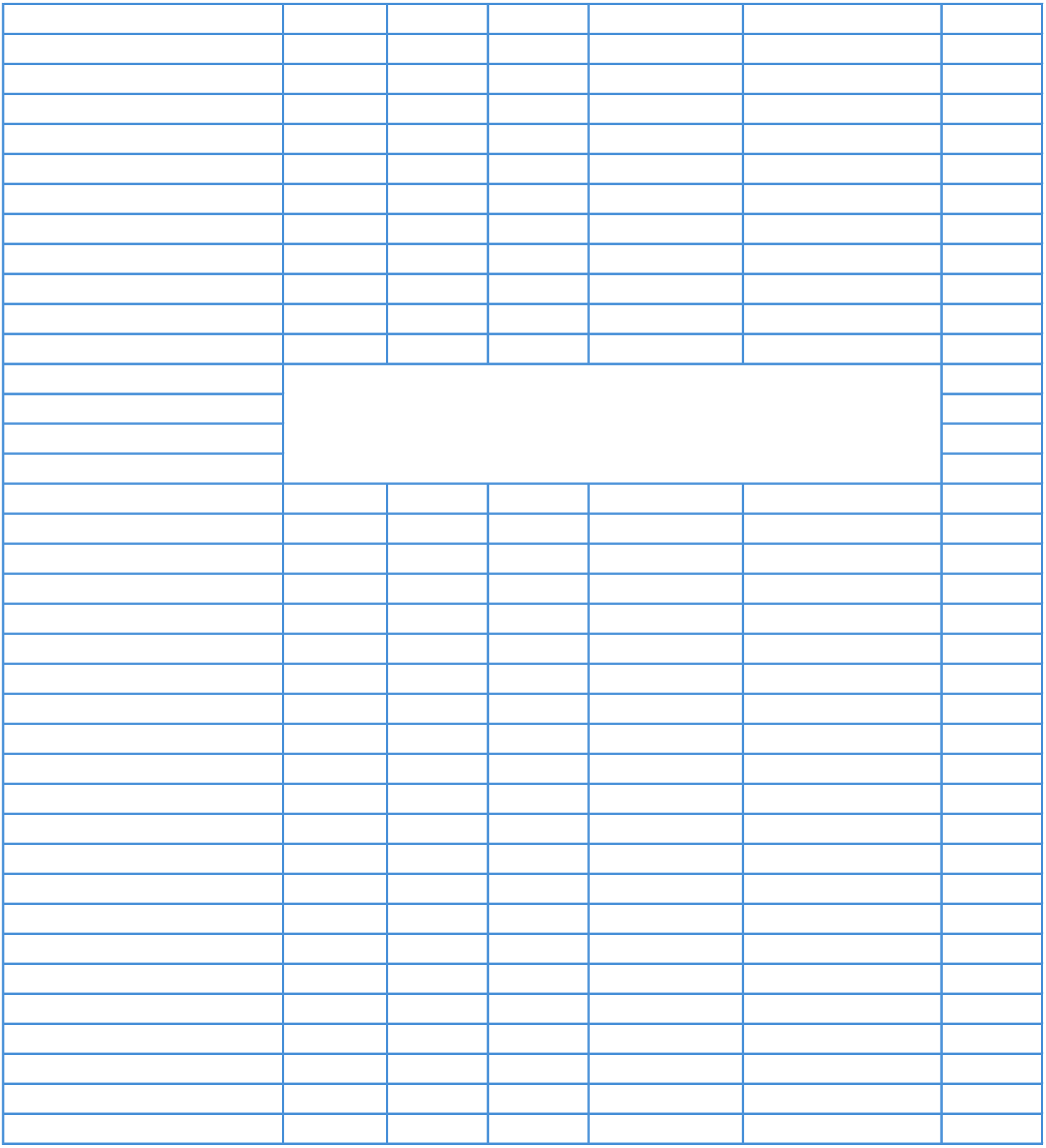
Mnemonic	Format	Opcode / Funct	Syntax	Name	equal 0 (shamt = mostly 0)
sll	R	0 / 0	$rd, $rt, shamt	Shift Left Logical	$rs
srl	R	0 / 2	$rd, $rt, shamt	Shift Right Logical	$rs
sra	R	0 / 3	$rd, $rt, shamt	Shift Right Arithmetic	$rs
jr	R	0 / 8	$rs	Jump Register	$rt, $rd
jalr	R	0 / 9	$rd, $rs	Jump and Link Register	$rt
div	R	0 / 1A (hex)	$rs, $rt	Divide	$rd
divu	R	0 / 1B (hex)	$rs, $rt	Divide Unsigned	$rd
mfhi	R	0 / 10 (hex)	$rd	Move from HI	$rs, $rt
mthi	R	0 / 11 (hex)	$rs	Move to HI	$rt, $rd
mflo	R	0 / 12 (hex)	$rd	Move from LO	$rs, $rt,
mtlo	R	0 / 13 (hex)	$rs	Move to LO	$rt, $rd
mult	R	0 / 18 (hex)	$rs, $rt	Multiply	$rd
multu	R	0 / 19 (hex)	$rs, $rt	Multiply Unsigned	$rd
slt	R	0 / 2A (hex)	$rd, $rs, $rt	Set Less Than	
sltu	R	0 / 2B (hex)	$rd, $rs, $rt	Set Less Than Unsigned	
add	R	0 / 20 (hex)	$rd, $rs, $rt	Add	
addu	R	0 / 21 (hex)	$rd, $rs, $rt	Add Unsigned	
sub	R	0 / 22 (hex)	$rd, $rs, $rt	Subtract	
subu	R	0 / 23 (hex)	$rd, $rs, $rt	Subtract Unsigned	
and	R	0 / 24 (hex)	$rd, $rs, $rt	And	
or	R	0 / 25 (hex)	$rd, $rs, $rt	Or	
xor	R	0 / 26 (hex)	$rd, $rs, $rt	Xor	
nor	R	0 / 27 (hex)	$rd, $rs, $rt	Nor	
mfc0	R	10 / 0 (hex)	$rt, $rd	Move from Coprocessor 0	$rs
clz	R	1C / 20 (hex)	$rd, $rs	Count Leading Zeros	$rt
j	J	2	target	Jump	
jal	J	3	target	Jump and Link	
beq	I	4	$rs, $rt, offset	Branch on Equal	
bne	I	5	$rs, $rt, offset	Branch on Not Equal	
blez	I	6	$rs, offset	Branch on Less Than or Equal Zero	$rt
bgtz	I	7	$rs, offset	Branch on Greater Than Zero	$rt
addi	I	8	$rt, $rs, immediate	Add Immediate	
addiu	I	9	$rt, $rs, immediate	Add Immediate Unsigned	
slti	I	A (hex)	$rt, $rs, immediate	Set Less Than Immediate	
sltiu	I	B (hex)	$rt, $rs, immediate	Set Less Than Imm. Unsigned	
andi	I	C (hex)	$rt, $rs, immediate	And Immediate	
ori	I	D (hex)	$rt, $rs, immediate	Or Immediate	
lui	I	F (hex)	$rt, immediate	Load Upper Immediate	$rs
lb	I	20 (hex)	$rt, offset($rs)	Load Byte	
lh	I	21 (hex)	$rt, offset($rs)	Load Half Word	
lw	I	23 (hex)	$rt, offset($rs)	Load Word	
lbu	I	24 (hex)	$rt, offset($rs)	Load Byte Unsigned	

lhu	I	25 (hex)	$rt, offset($rs)	Load Half Word Unsigned	
sb	I	28 (hex)	$rt, offset($rs)	Store Byte	
sh	I	29 (hex)	$rt, offset($rs)	Store Half Word	
sw	I	2B (hex)	$rt, offset($rs)	Store Word	
ll	I	30 (hex)	$rt, offset($rs)	Load Linked	
lwcl	I	31 (hex)	$rt, offset($rs)	Load Word Coprocessor Load	
ldlc	I	35 (hex)	$rt, offset($rs)	Load Doubleword Left Coprocessor	
sc	I	38 (hex)	$rt, offset($rs)	Store Conditional	
swcl	I	39 (hex)	$rt, offset($rs)	Store Word Coprocessor Load	
sdcl	I	3D (hex)	$rt, offset($rs)	Store Doubleword Coprocessor	

The MIPS Blue Card



The MIPS Blue Card