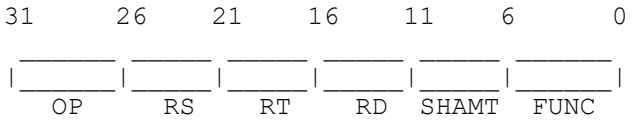
Reference Material – Lab 2

Introduction to MIPS Instruction

General Format



op	rs	rt	rd	shamt	funct
6 bits	5 bits	5 bits	5 bits	5 bits	6 bits

op
: Operation code

rs
: First source register operand

rt
: Second source register operand

rd
: Destination register operand

shamt
: Shift amount - used in shift instructions

funct
: Select the variant of the operation in the op code field

Specific Instruction Formats

Format	6 bits	5 bits	5 bits	5 bits	5 bits	6 bits	Comments
R	op	rs	rt	rd	shamt	funct	Arithmetic
I	op	rs	rt	address/immediate			Transfer, branch,immediate
J	op	target address					Jump

MIPS Instruction Set

The MIPS instruction set illustrates four underlying principles of hardware design:

1. Simplicity favors regularity.
2. Smaller is faster.
3. Good design demands compromise.
4. Make the common case fast.

Simplicity favors regularity

Consider the following example:

Category	Instruction	Example	Meaning	Comments
Arithmetic	add	add a,b,c	a=b+c	Always 3 operands
Arithmetic	subtract	sub a,b,c	a=b-c	Always 3 operands

Note that each operand has *exactly* three operands.

Arithmetic Instructions

Instruction	Example	Meaning	Comments
add	add $1,$2,$3	$1=$2+$3	Always 3 operands
subtract	sub $1,$2,$3	$1=$2-$3	Always 3 operands
add immediate	addi $1,$2,10	$1=$2+10	add constant
add unsigned	addu $1,$2,$3	$1=$2+$3	Always 3 operations
subtract unsigned	subu $1,$2,$3	$1=$2-$3	Always 3 operations
add immed.unsigned	addiu $1,$2,10	$1=$2+10	Always 3 operations

Logical

Instruction	Example	Meaning	Comments
and	and $1,$2,$3	$1=$2&$3	3 register operands
or	or $1,$2,$3	$1=$2|$3	3 register operands
and immediate	andi $1,$2,10	$1=$2&10	AND constant
or immediate	or $1,$2,10	$1=$2|10	OR constant
shift left logical	sll $1,$2,10	$1=$2<<10	Shift left by constant
shift right logical	srl $1,$2,10	$1=$2>>10	Shift right by constant

Data Transfer

Instruction	Example	Meaning	Comments
load word	lw $1,10($2)	$1=Memory[$2+10]	memory to register
store word	sw $1,10($2)	Memory[$2+10]=$1	register to memory
load upper immed.	lui $1,10	$1=10x2^16	load constant into upper 16 bits

Conditional Branch

Instruction	Example	Meaning	Comments
branch on equal	beq $1,$2,10	if($1==$2)go to PC+4+10	Equal test
branch on not equal	bne $1,$2,10	if($1!=$2)go to PC+4+10	Not equal test
set on less then	slt $1,$2,$3	if($2<$3)$1=1;else $1=0	Less than compare

Unconditional Jump

Instruction	Example	Meaning	Comments
jump	j 1000	go to 1000	Jump to target address
jump register	jr $31	go to $31	For switch, procedure return
jump and link	jal 1000	$31=PC+4; go to 1000	For procedure call