Architecture

- Software visible registers
- Memory Addressing
- The instruction set
- The exception / reset mechanism

Instruction Set

RISC Architecture

- Only simple instructions
- All instructions have the same size (32 bits)
- 3-operand instructions (2 read, 1 write)
- No operation involving operands in memory - Only load and store operations

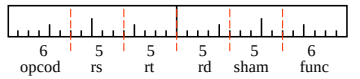
Instruction Set

3 instruction formats

- R Register-register instructions
- I Immediate instructions
- J Jump instructions

Instruction Set

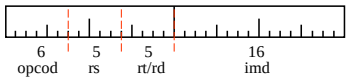
R format



opcod	Operation code
func	Function (opcod extension)
rs	# of source register
rt	# of source register
rd	# of destination register
sham	Shift amount (# of bit the opr. is shifted)

Instruction Set

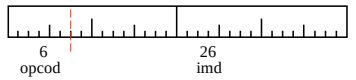
I format



opcod	Operation code
rs	# of source register
rt / rd	# of source or destination register
imd	Immediate value

Instruction Set

J format



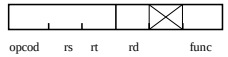
opcod	Operation code
imd	Immediate value

Instruction Set

- Arithmetic and logic instructions
- Memory access instructions
- Control instructions
- System instructions

Arithmetic and logic

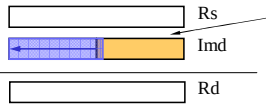
Assembly language	Action	Format
Add Rd, Rs, Rt	Rs+Rt ⇨ Rd	
add	Overflow ⇨ exc.	



R format

Arithmetic and logic

Assembly language	Action	Format
Addi Rd, Rs, Imd	Rs+Imd ⇨ Rd	opcod rs rd imd
add immediate	Overflow ⇨ exc. Imd is sign extended	I format

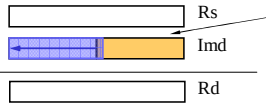


Arithmetic and logic

Assembly language	Action	Format
Addu Rd, Rs, Rt	Rs+Rt ⇨ Rd	opcod rs rt rd func
add unsigned	No overflow	R format

Arithmetic and logic

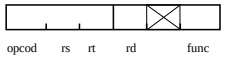
Assembly language	Action	Format
Addiu Rd, Rs, Imd	Rs+Imd ⇨ Rd	opcod rs rd imd
add immediate unsigned	No overflow Imd is sign extended	I format



Arithmetic and logic

Assembly language	Action	Format
Sub Rd, Rs, Rt	Rs–Rt ⇨ Rd	opcod rs rt rd func
subtract	Overflow ⇨ exc.	R format

Arithmetic and logic

Assembly language	Action	Format
Subu Rd, Rs, Rt	Rs–Rt ⇨ Rd	
Subtract unsigned	No overflow	R format

Arithmetic and logic

Assembly language	Action	Format
Or Rd, Rs, Rt	Rs or Rt ⇨ Rd	
bitwise logic OR		R format

Arithmetic and logic

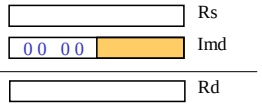
Assembly language	Action	Format
Ori Rd, Rs, Imd	Rs or Imd ⇨ Rd	
bitwise logic OR immediate	Imd is zero extended	I format

Arithmetic and logic

Assembly language	Action	Format
And Rd, Rs, Rt	Rs and Rt ⇨ Rd	
bitwise logic AND		R format

Arithmetic and logic

Assembly language	Action	Format
Andi Rd, Rs, Imd	Rs and Imd ⇨ Rd	opcod rs rd imd
bitwise logic AND immediate	Imd is zero extended	I format

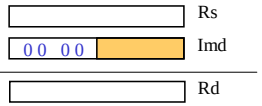


Arithmetic and logic

Assembly language	Action	Format
Xor Rd, Rs, Rt	Rs xor Rt ⇨ Rd	opcod rs rt rd func
bitwise logic XOR		R format

Arithmetic and logic

Assembly language	Action	Format
Xori Rd, Rs, Imd	Rs xor Imd ⇨ Rd	opcod rs rd imd
bitwise logic XOR immediate	Imd is zero extended	I format

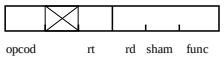


Arithmetic and logic

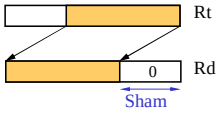
Assembly language	Action	Format
Nor Rd, Rs, Rt	Rs nor Rt ⇨ Rd	opcod rs rt rd func
bitwise logic NOR		R format

Arithmetic and logic

Assembly language	Action	Format
Sll Rd, Rt, Sham	Rt << Sham ⇨ Rd	R format



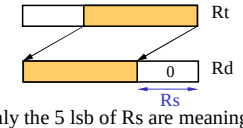
Shift left logic



Arithmetic and logic

Assembly language	Action	Format
Sllv Rd, Rt, Rs	Rt << Rs ⇨ Rd	R format

Shift left logic variable

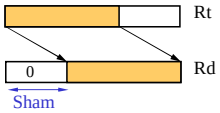


Only the 5 lsb of Rs are meaningful

Arithmetic and logic

Assembly language	Action	Format
Srl Rd, Rt, Sham	Rt >> Sham ⇨ Rd	R format

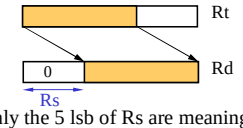
Shift right logic



Arithmetic and logic

Assembly language	Action	Format
Srlv Rd, Rt, Rs	Rt >> Rs ⇨ Rd	R format

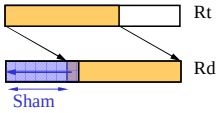
Shift right logic variable



Only the 5 lsb of Rs are meaningful

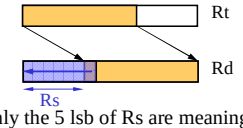
Arithmetic and logic

Assembly language	Action	Format
Sra Rd, Rt, Sham	Rt >> Sham ⇨ Rd	opcod rt rd sham func
Shift right arithmetic		R format



Arithmetic and logic

Assembly language	Action	Format
Srav Rd, Rt, Rs	Rt >> Rs ⇨ Rd	opcod rs rt rd func
Shift right arithmetic variable		R format



Only the 5 lsb of Rs are meaningful

Arithmetic and logic

Assembly language	Action	Format
Slt Rd, Rs, Rt	Rs < Rt ?	opcod rs rt rd func
Set if less than	Yes : 1 ⇨ Rd No : 0 ⇨ Rd	R format

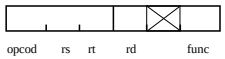
Operands are signed

Arithmetic and logic

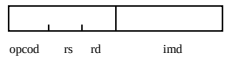
Assembly language	Action	Format
Slti Rd, Rs, Imd	Rs < Imd ?	opcod rs rd imd
Set if less than immediate	Yes : 1 ⇨ Rd No : 0 ⇨ Rd	I format

Operands are signed
Imd is sign extended

Arithmetic and logic

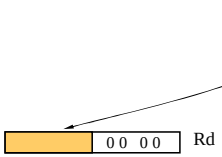
Assembly language	Action	Format
Sltu Rd, Rs, Rt	Rs < Rt ?	
Set if less than unsigned	Yes : 1 ⇨ Rd No : 0 ⇨ Rd	R format
	Operands are unsigned	

Arithmetic and logic

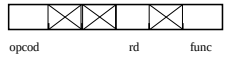
Assembly language	Action	Format
Sltiu Rd, Rs, Imd	Rs < Imd ?	
Set if less than immediate, unsigned	Yes : 1 ⇨ Rd No : 0 ⇨ Rd	I format
	Operands are unsigned Imd is sign extended	

Arithmetic and logic

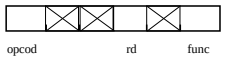
Assembly language	Action	Format
Lui Rd, Imd	Imd << 16 ⇨ Rd	
load upper immediate		I format



Arithmetic and logic

Assembly language	Action	Format
Mflo Rd	LO ⇨ Rd	
Move from LO		R format

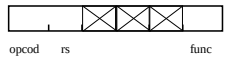
Arithmetic and logic

Assembly language	Action	Format
Mfhi Rd	HI ⇨ Rd	

Move from HI

R format

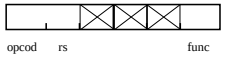
Arithmetic and logic

Assembly language	Action	Format
Mtlo Rd	Rs ⇨ LO	

Move to LO

R format

Arithmetic and logic

Assembly language	Action	Format
Mthi Rd	Rs ⇨ HI	

Move to HI

R format