

1 Instruction Set Reference List

		31	25	21	17	13	0
Formats	r-type:	opcode	rd	rs1	rs2	0	
	i-type:	opcode	rd	rs	18-bit immediate		
	l-type:	opcode	rd	22-bit immediate			

Instructions	NOP	000000	0			
	SET	000001	rd	rs	18-bit immediate	
	LOAD	000010	rd	22-bit immediate		
	MOV	000011	rd	rs	0	
	FADD	000011	rd	rs1	rs2	0
	FSUB	000100	rd	rs1	rs2	0
	NEG	000101	rd	rs1	0	0

NOP No operation

Opcode: 000000
 Syntax: NOP
 Purpose: Perform no operations.

SET Set register to floating-point value

Opcode: 000001
 Syntax r-type: SET, rd, #<32-bit FP value>
 Purpose: Assign a 32-bit floating point value to rd.
 Operation: $rd \leftarrow \text{FPvalue}$

LOAD Load value from memory

Opcode: 000010
 Syntax r-type: LOAD, rd, rs
 Purpose: Assign rd the value from the memory address in rs.
 Operation: $rd \leftarrow M[rs]$

STORE Store value to memory

Opcode: 000011
 Syntax r-type: STORE, rd, rs
 Purpose: Assign memory location specified in rd to value in rs.
 Operation: $M[rd] \leftarrow rs$

MOVE Copy value from a register to another

Opcode: 000100
 Syntax r-type: MOVE, rd, rs
 Purpose: Assign rd the value in rs.
 Operation: $rd \leftarrow rs$

FADD

Add 32-bit floating-point value

Opcode:	000101
Syntax r-type:	FADD, rd, rs1, rs2
Syntax i-type:	FADD, rd, rs1, <32-bit ? immediate>
Purpose:	Performs addition on two 32-bit floating-point values from rs1 and rs2, or an immediate in place of rs2, and stores the result in rd.
Operation:	$rd \leftarrow rs1 + rs2$ or $rd \leftarrow rs1 + \text{immediate}$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

FSUB

Subtract 32-bit floating-point values

Opcode:	000110
Syntax r-type:	FSUB, rd, rs1, rs2
Syntax i-type:	FSUB, rd, rs1, <32-bit ? immediate>
Purpose:	Performs subtraction on two 32-bit floating-point values from rs1 and rs2, or an immediate in place of rs2, and stores the result in rd.
Operation:	$rd \leftarrow rs1 - rs2$ or $rd \leftarrow rs1 - \text{immediate}$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

FNEG

Negate a 32-bit floating-point value

Opcode:	000111
Syntax:	FNEG, rd, rs
Purpose:	Performs negation on a 32-bit floating-point value from rs and stores the result in rd.
Operation:	$rd \leftarrow -rs$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

FMUL

Multiply two 32-bit floating-point values

Opcode:	001000
Syntax r-type:	FMUL, rd, rs1, rs2
Syntax i-type:	FMUL, rd, rs1, <32-bit ? immediate>
Purpose:	Performs multiplication on two 32-bit floating-point values from rs1 and rs2, or an immediate in place of rs2, and stores the result in rd.
Operation:	$rd \leftarrow rs1 * rs2$ or $rd \leftarrow rs1 * \text{immediate}$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

FDIV

Divide two 32-bit floating-point values

Opcode:	001001
Syntax r-type:	FDIV, rd, rs1, rs2
Syntax i-type:	FDIV, rd, rs1, <32-bit ? immediate>
Purpose:	Performs division on two 32-bit floating-point values from rs1 and rs2, or an immediate in place of rs2, and stores the result in rd.
Operation:	$rd \leftarrow rs1 \div rs2$ or $rd \leftarrow rs1 \div \text{immediate}$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

FLOOR

Compute the floor function

Opcode:	001010
Syntax r-type:	FLOOR, rd, rs
Purpose:	Rounds the value in rs to the nearest lowest integer and stores the result in rd.
Operation:	$rd \leftarrow \lfloor rs \rfloor$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

CEILING

Compute the ceiling function

Opcode:	001011
Syntax:	CEILING, rd, rs
Purpose:	Rounds the value in rs to the nearest highest integer and stores the result in rd.
Operation:	$rd \leftarrow \lceil rs \rceil$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

ROUND

Round a value

Opcode:	001100
Syntax:	ROUND, rd, rs
Purpose:	Rounds the value in rs and stores the result in rd.
Operation:	$rd \leftarrow \text{round}(rs)$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

FABS

Compute the absolute value

Opcode:	001101
Syntax:	FABS, rd, rs
Purpose:	Find the absolute value in rs and stores the result in rd.
Operation:	$rd \leftarrow rs $
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

MIN

Find the smallest value

Opcode:	001110
Syntax:	MIN, rd, rs1, rs2
Purpose:	Find the smallest value between rs1 and rs2 and store it in rd.
Operation:	$rd \leftarrow \min(rs1, rs2)$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

MAX

Find the largest

Opcode:	001111
Syntax:	MAX, rd, rs1, rs2
Purpose:	Find the largest value between rs1 and rs2 and store it in rd.
Operation:	$rd \leftarrow \max(rs1, rs2)$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

POW

Compute the power

Opcode:	010000
Syntax:	POW, rd, rs1, #<integer-value>
Purpose:	Find rs1 raised to an integer value and store it in rd.
Operation:	$rd \leftarrow rs1^{integer-value}$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

EXP

Compute the exponent

Opcode:	010001
Syntax:	EXP, rd, rs
Purpose:	Find e raised to the value in rs and store it in rd.
Operation:	$rd \leftarrow e^{rs}$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

SQRT

Compute the square root of a value

Opcode:	010010
Syntax:	SQRT , rd, rs
Purpose:	Find the square root of the value in rs and store the result in rd.
Operation:	$rd \leftarrow \text{sqrtrs}$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

B

Branch unconditionally

Opcode:	010011
Syntax:	B , rd
Purpose:	Set the program counter to the value in memory addressed by rd.
Operation:	$PC \leftarrow M[rd]$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

BZ

Branch if zero

Opcode:	010100
Syntax:	BZ , rd, <LABEL>
Purpose:	Branch to the label specified in the assembly program if rd is equal to zero.
Operation:	if (rd == 0) $PC \leftarrow \text{LABEL}$
Condition Codes:	$\frac{N \ Z \ V}{x \ - \ x}$

BNBranch if negative

Opcode: 010101

Syntax: BN, rd, <LABEL>

Purpose: Branch to the label specified in the assembly program if
rd is less than zero.Operation: if (rd < 0)
 $PC \leftarrow LABEL$ Condition Codes: $\frac{N \ Z \ V}{x \ - \ x}$