RISC-V Summary

Register name	Alias	Use	Saver
x0	zero	read-only (0)	n/a
x1	ra	return address	caller
x2	sp	stack pointer	callee
x3	gp	global pointer	n/a
x4	tp	thread pointer	n/a
x5-x7	t0-t2	temporary	caller
x8-x9	s0-s1	saved	callee
x10-x11	a0-a1	arguments/return values	caller
x12-x17	a2-a7	arguments	caller
x18-x27	s2-s11	saved	callee
x28-x31	t3-t6	temporary	caller

31 25 24 20 19 15 14 12 11 7 6 0

funct7	rs2	rs1	funct3	rd	opcode	R-type
imm[11:0]		rs1	000	rd	opcode	I-type
imm[11:5]	rs2	rs1	funct3	imm[4:0]	opcode	S-type
imm[12|10:5]	rs2	rs1	fuct3	imm[4:1|11]	opcode	B-type
imm[31:12]				rd	opcode	U-type
imm[20|10:1|11|19:12]				rd	opcode	J-type

Pseudoinstructions

bgt *rs1, rs2, label*
ble *rs1, rs2, label*
bgtu *rs1, rs2, label*
bleu *rs1, rs2, label*

beqz *rs1, label*
bnez *rs1, label*
bltz *rs1 label*
bgez *rs1, label*
blez *rs1, label*

j *label*
jal *label*
jr *rs1*
jalr *rs1*
ret

li *rd, constant*
la *rd, label*
mv *rd, rs1*
not *rd, rs1*
neg rd, rs1

seqz *rd, rs1*
snez *rd, rs1*
sltz *rd, rs1*
sgtz *rd, rs1*

Directives

.ascii
.asciz
.bss
.data
.dword
.equ
.extern
.global
.globl
.rodata
.space
.start
.string
.text
.word
.zero

RV32M Standard Extension

31 25 24 20 19 15 14 12 11 7 6 0

0000001	rs2	rs1	000	rd	0110011	mul
0000001	rs2	rs1	001	rd	0110011	mulh
0000001	rs2	rs1	010	rd	0110011	mulhsu
0000001	rs2	rs1	011	rd	0110011	mulhu
0000001	rs2	rs1	100	rd	0110011	div
0000001	rs2	rs1	101	rd	0110011	divu
0000001	rs2	rs1	110	rd	0110011	rem
0000001	rs2	rs1	111	rd	0110011	remu

RV32I Base Instruction Set[1]

31 25	24 20	19 15	14 12	11 7	6 0	
imm[31:12]				rd	0110111	lui
imm[31:12]				rd	0010111	auipc
imm[20|10:1|11|19:12]				rd	1101111	jal
imm[11:0]		rs1	000	rd	1100111	jalr
imm[12|10:5]	rs2	rs1	000	imm[4:1|11]	1100011	beq
imm[12|10:5]	rs2	rs1	001	imm[4:1|11]	1100011	bne
imm[12|10:5]	rs2	rs1	100	imm[4:1|11]	1100011	blt
imm[12|10:5]	rs2	rs1	101	imm[4:1|11]	1100011	bge
imm[12|10:5]	rs2	rs1	110	imm[4:1|11]	1100011	bltu
imm[12|10:5]	rs2	rs1	111	imm[4:1|11]	1100011	bgeu
imm[11:0]		rs1	000	rd	0000011	lb
imm[11:0]		rs1	001	rd	0000011	lh
imm[11:0]		rs1	010	rd	0000011	lw
imm[11:0]		rs1	100	rd	0000011	lbu
imm[11:0]		rs1	101	rd	0000011	lhu
imm[11:5]	rs2	rs1	000	imm[4:0]	0100011	sb
imm[11:5]	rs2	rs1	001	imm[4:0]	0100011	sh
imm[11:5]	rs2	rs1	010	imm[4:0]	0100011	sw
imm[11:0]		rs1	000	rd	0010011	addi
imm[11:0]		rs1	010	rd	0010011	slti
imm[11:0]		rs1	011	rd	0010011	sltiu
imm[11:0]		rs1	100	rd	0010011	xori
imm[11:0]		rs1	110	rd	0010011	ori
imm[11:0]		rs1	111	rd	0010011	andi
0000000	shamt	rs1	001	rd	0010011	slli
0000000	shamt	rs1	101	rd	0010011	srli
0100000	shamt	rs1	101	rd	0010011	srai
0000000	rs2	rs1	000	rd	0110011	add
0100000	rs2	rs1	000	rd	0110011	sub
0000000	rs2	rs1	001	rd	0110011	sll
0000000	rs	rs1	010	rd	0110011	slt
0000000	rs2	rs1	011	rd	0110011	sltu
0000000	rs2	rs1	100	rd	0110011	xor
0000000	rs2	rs1	101	rd	0110011	srl
0100000	rs2	rs1	101	rd	0110011	sra
0000000	rs2	rs1	110	rd	0110011	or
0000000	rs2	rs1	111	rd	0110011	and
0000000			000		0000000	halt[2]
0000001			000		0000000	nl[2]
0000010		rs1	000		0000000	dout[2]
0000011		rs1	000		0000000	udout[2]
0000100		rs1	000		0000000	hout[2]
0000101		rs1	000		0000000	aout[2]
0000110		rs1	000		0000000	sout[2]
0000111			000	rd	0000000	din[2]
0001000			000	rd	0000000	hin[2]
0001001			000	rd	0000000	ain[2]
0001010		rs1	000		0000000	sin[2]
0001011			000		0000000	m[2]
0001100			000		0000000	x[2]
0001101			000		0000000	s[2]
0001110			000		0000000	bp[2]
0001111		rs1	000		0000000	ddout[2]
0010000		rs1	000		0000000	dudout[2]
0010001		rs1	000		0000000	dhout[2]

1: `fence`, `fence.1`, `ecall`, `ebreak` not supported by `rv` 2: `rv`-supported but not in RISC-V