

RV32I Base Instruction Set

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	rs2	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	
fm	pred	succ	rs1	000	rd	0001111	FENCE
000000000000			00000	000	00000	1110011	ECALL
000000000001			00000	000	00000	1110011	EBREAK

RV32I Reference Card

Usage Template	Type	Description	Detailed Description
add rd, rs1, rs2	R	Add	$rd \leftarrow rs1 + rs2, pc \leftarrow pc+4$
addi rd, rs1, imm	I	Add Immediate	$rd \leftarrow rs1 + \text{imm_i}, pc \leftarrow pc+4$
and rd, rs1, rs2	R	And	$rd \leftarrow rs1 \& rs2, pc \leftarrow pc+4$
andi rd, rs1, imm	I	And Immediate	$rd \leftarrow rs1 \& \text{imm_i}, pc \leftarrow pc+4$
auipc rd, imm	U	Add Upper Immediate to PC	$rd \leftarrow pc + \text{imm_u}, pc \leftarrow pc+4$
beq rs1, rs2, pcrel_13	B	Branch Equal	$pc \leftarrow pc + ((rs1 == rs2) ? \text{imm_b} : 4)$
bge rs1, rs2, pcrel_13	B	Branch Greater or Equal	$pc \leftarrow pc + ((rs1 \geq rs2) ? \text{imm_b} : 4)$
bgeu rs1, rs2, pcrel_13	B	Branch Greater or Equal Unsigned	$pc \leftarrow pc + ((rs1 \geq rs2) ? \text{imm_b} : 4)$
blt rs1, rs2, pcrel_13	B	Branch Less Than	$pc \leftarrow pc + ((rs1 < rs2) ? \text{imm_b} : 4)$
bltu rs1, rs2, pcrel_13	B	Branch Less Than Unsigned	$pc \leftarrow pc + ((rs1 < rs2) ? \text{imm_b} : 4)$
bne rs1, rs2, pcrel_13	B	Branch Not Equal	$pc \leftarrow pc + ((rs1 != rs2) ? \text{imm_b} : 4)$
jal rd, pcrel_21	J	Jump And Link	$rd \leftarrow pc+4, pc \leftarrow pc + \text{imm_j}$
jalr rd, imm(rs1)	I	Jump And Link Register	$rd \leftarrow pc+4, pc \leftarrow (rs1 + \text{imm_i}) \&^1$
lb rd, imm(rs1)	I	Load Byte	$rd \leftarrow \text{sx}(m8(rs1 + \text{imm_i})), pc \leftarrow pc+4$
lbu rd, imm(rs1)	I	Load Byte Unsigned	$rd \leftarrow \text{zx}(m8(rs1 + \text{imm_i})), pc \leftarrow pc+4$
lh rd, imm(rs1)	I	Load Halfword	$rd \leftarrow \text{sx}(m16(rs1 + \text{imm_i})), pc \leftarrow pc+4$
lhu rd, imm(rs1)	I	Load Halfword Unsigned	$rd \leftarrow \text{zx}(m16(rs1 + \text{imm_i})), pc \leftarrow pc+4$
lui rd, imm	U	Load Upper Immediate	$rd \leftarrow \text{imm_u}, pc \leftarrow pc+4$
lw rd, imm(rs1)	I	Load Word	$rd \leftarrow \text{sx}(m32(rs1 + \text{imm_i})), pc \leftarrow pc+4$
or rd, rs1, rs2	R	Or	$rd \leftarrow rs1 rs2, pc \leftarrow pc+4$
ori rd, rs1, imm	I	Or Immediate	$rd \leftarrow rs1 \text{imm_i}, pc \leftarrow pc+4$
sb rs2, imm(rs1)	S	Store Byte	$m8(rs1 + \text{imm_s}) \leftarrow rs2[7:0], pc \leftarrow pc+4$
sh rs2, imm(rs1)	S	Store Halfword	$m16(rs1 + \text{imm_s}) \leftarrow rs2[15:0], pc \leftarrow pc+4$
sll rd, rs1, rs2	R	Shift Left Logical	$rd \leftarrow rs1 \ll (rs2 \% XLEN), pc \leftarrow pc+4$
slli rd, rs1, shamt	I	Shift Left Logical Immediate	$rd \leftarrow rs1 \ll \text{shamt_i}, pc \leftarrow pc+4$
slt rd, rs1, rs2	R	Set Less Than	$rd \leftarrow (rs1 < rs2) ? 1 : 0, pc \leftarrow pc+4$
slti rd, rs1, imm	I	Set Less Than Immediate	$rd \leftarrow (rs1 < \text{imm_i}) ? 1 : 0, pc \leftarrow pc+4$
sltiu rd, rs1, imm	I	Set Less Than Immediate Unsigned	$rd \leftarrow (rs1 < \text{imm_i}) ? 1 : 0, pc \leftarrow pc+4$
sltu rd, rs1, rs2	R	Set Less Than Unsigned	$rd \leftarrow (rs1 < rs2) ? 1 : 0, pc \leftarrow pc+4$
sra rd, rs1, rs2	R	Shift Right Arithmetic	$rd \leftarrow rs1 \gg (rs2 \% XLEN), pc \leftarrow pc+4$
srai rd, rs1, shamt	I	Shift Right Arithmetic Immediate	$rd \leftarrow rs1 \gg \text{shamt_i}, pc \leftarrow pc+4$
srl rd, rs1, rs2	R	Shift Right Logical	$rd \leftarrow rs1 \gg (rs2 \% XLEN), pc \leftarrow pc+4$
srli rd, rs1, shamt	I	Shift Right Logical Immediate	$rd \leftarrow rs1 \gg \text{shamt_i}, pc \leftarrow pc+4$
sub rd, rs1, rs2	R	Subtract	$rd \leftarrow rs1 - rs2, pc \leftarrow pc+4$
sw rs2, imm(rs1)	S	Store Word	$m32(rs1 + \text{imm_s}) \leftarrow rs2[31:0], pc \leftarrow pc+4$
xor rd, rs1, rs2	R	Exclusive Or	$rd \leftarrow rs1 \wedge rs2, pc \leftarrow pc+4$
xori rd, rs1, imm	I	Exclusive Or Immediate	$rd \leftarrow rs1 \wedge \text{imm_i}, pc \leftarrow pc+4$