<table>
<tr><td colspan="4">31 25</td><td colspan="2">24 20</td><td>19 15</td><td>14 12</td><td>11 7</td><td>6 0</td><td></td></tr>
<tr><td colspan="8">imm[31:12]</td><td>rd</td><td>opcode</td><td><b>Type-U</b></td></tr>
<tr><td colspan="8">imm[20|10:1|11|19:12]</td><td>rd</td><td>opcode</td><td><b>Type-UJ</b></td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>funct3</td><td>rd</td><td>opcode</td><td><b>Type-I</b></td></tr>
<tr><td colspan="4">imm[12|10:5]</td><td colspan="2">rs2</td><td>rs1</td><td>funct3</td><td>imm[4:1|11]</td><td>opcode</td><td><b>Type-SB</b></td></tr>
<tr><td colspan="4">imm[11:5]</td><td colspan="2">rs2</td><td>rs1</td><td>funct3</td><td>imm[4:0]</td><td>opcode</td><td><b>Type-S</b></td></tr>
<tr><td colspan="2">funct5</td><td colspan="2">funct2</td><td colspan="2">rs2</td><td>rs1</td><td>funct3</td><td>rd</td><td>opcode</td><td><b>Type-R</b></td></tr>
</table>

RV32I Base Integer Instruction Set

<table>
<tr><td colspan="8">imm[31:12]</td><td>rd</td><td>0110111</td><td>LUI rd, imm</td></tr>
<tr><td colspan="8">imm[31:12]</td><td>rd</td><td>0010111</td><td>AUIPC rd, offset</td></tr>
<tr><td colspan="8">imm[20|10:1|11|19:12]</td><td>rd</td><td>1101111</td><td>JAL rd, offset</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>000</td><td>rd</td><td>1100111</td><td>JALR rd, rs1, offset</td></tr>
<tr><td colspan="4">imm[12|10:5]</td><td colspan="2">rs2</td><td>rs1</td><td>000</td><td>imm[4:1|11]</td><td>1100011</td><td>BEQ rs1, rs2, offset</td></tr>
<tr><td colspan="4">imm[12|10:5]</td><td colspan="2">rs2</td><td>rs1</td><td>001</td><td>imm[4:1|11]</td><td>1100011</td><td>BNE rs1, rs2, offset</td></tr>
<tr><td colspan="4">imm[12|10:5]</td><td colspan="2">rs2</td><td>rs1</td><td>100</td><td>imm[4:1|11]</td><td>1100011</td><td>BLT rs1, rs2, offset</td></tr>
<tr><td colspan="4">imm[12|10:5]</td><td colspan="2">rs2</td><td>rs1</td><td>101</td><td>imm[4:1|11]</td><td>1100011</td><td>BGE rs1, rs2, offset</td></tr>
<tr><td colspan="4">imm[12|10:5]</td><td colspan="2">rs2</td><td>rs1</td><td>110</td><td>imm[4:1|11]</td><td>1100011</td><td>BLTU rs1, rs2, offset</td></tr>
<tr><td colspan="4">imm[12|10:5]</td><td colspan="2">rs2</td><td>rs1</td><td>111</td><td>imm[4:1|11]</td><td>1100011</td><td>BGEU rs1, rs2, offset</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>000</td><td>rd</td><td>0000011</td><td>LB rd, offset(rs1)</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>001</td><td>rd</td><td>0000011</td><td>LH rd, offset(rs1)</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>010</td><td>rd</td><td>0000011</td><td>LW rd, offset(rs1)</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>100</td><td>rd</td><td>0000011</td><td>LBU rd, offset(rs1)</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>101</td><td>rd</td><td>0000011</td><td>LHU rd, offset(rs1)</td></tr>
<tr><td colspan="4">imm[11:5]</td><td colspan="2">rs2</td><td>rs1</td><td>000</td><td>imm[4:0]</td><td>0100011</td><td>SB rs2, offset(rs1)</td></tr>
<tr><td colspan="4">imm[11:5]</td><td colspan="2">rs2</td><td>rs1</td><td>001</td><td>imm[4:0]</td><td>0100011</td><td>SH rs2, offset(rs1)</td></tr>
<tr><td colspan="4">imm[11:5]</td><td colspan="2">rs2</td><td>rs1</td><td>010</td><td>imm[4:0]</td><td>0100011</td><td>SW rs2, offset(rs1)</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>000</td><td>rd</td><td>0010011</td><td>ADDI rd, rs1, imm</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>010</td><td>rd</td><td>0010011</td><td>SLTI rd, rs1, imm</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>011</td><td>rd</td><td>0010011</td><td>SLTIU rd, rs1, imm</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>100</td><td>rd</td><td>0010011</td><td>XORI rd, rs1, imm</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>110</td><td>rd</td><td>0010011</td><td>ORI rd, rs1, imm</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>111</td><td>rd</td><td>0010011</td><td>ANDI rd, rs1, imm</td></tr>
<tr><td colspan="3">000000</td><td>0</td><td colspan="2">shamt5</td><td>rs1</td><td>001</td><td>rd</td><td>0010011</td><td>SLLI rd, rs1, imm</td></tr>
<tr><td colspan="3">000000</td><td>0</td><td colspan="2">shamt5</td><td>rs1</td><td>101</td><td>rd</td><td>0010011</td><td>SRLI rd, rs1, imm</td></tr>
<tr><td colspan="3">010000</td><td>0</td><td colspan="2">shamt5</td><td>rs1</td><td>101</td><td>rd</td><td>0010011</td><td>SRAI rd, rs1, imm</td></tr>
<tr><td colspan="2">00000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>000</td><td>rd</td><td>0110011</td><td>ADD rd, rs1, rs2</td></tr>
<tr><td colspan="2">01000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>000</td><td>rd</td><td>0110011</td><td>SUB rd, rs1, rs2</td></tr>
<tr><td colspan="2">00000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>001</td><td>rd</td><td>0110011</td><td>SLL rd, rs1, rs2</td></tr>
<tr><td colspan="2">00000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>010</td><td>rd</td><td>0110011</td><td>SLT rd, rs1, rs2</td></tr>
<tr><td colspan="2">00000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>011</td><td>rd</td><td>0110011</td><td>SLTU rd, rs1, rs2</td></tr>
<tr><td colspan="2">00000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>100</td><td>rd</td><td>0110011</td><td>XOR rd, rs1, rs2</td></tr>
<tr><td colspan="2">00000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>101</td><td>rd</td><td>0110011</td><td>SRL rd, rs1, rs2</td></tr>
<tr><td colspan="2">01000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>101</td><td>rd</td><td>0110011</td><td>SRA rd, rs1, rs2</td></tr>
<tr><td colspan="2">00000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>110</td><td>rd</td><td>0110011</td><td>OR rd, rs1, rs2</td></tr>
<tr><td colspan="2">00000</td><td colspan="2">00</td><td colspan="2">rs2</td><td>rs1</td><td>111</td><td>rd</td><td>0110011</td><td>AND rd, rs1, rs2</td></tr>
<tr><td>0000</td><td colspan="3">pred</td><td>pred</td><td>succ</td><td>00000</td><td>000</td><td>00000</td><td>0001111</td><td>FENCE</td></tr>
<tr><td colspan="4">0000000</td><td colspan="2">00000</td><td>00000</td><td>001</td><td>00000</td><td>0001111</td><td>FENCE.I</td></tr>
</table>

RV64I Base Integer Instruction Set (in addition to RV32I)

<table>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>110</td><td>rd</td><td>0000011</td><td>LWU rd, offset(rs1)</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>011</td><td>rd</td><td>0000011</td><td>LD rd, offset(rs1)</td></tr>
<tr><td colspan="4">imm[11:5]</td><td colspan="2">rs2</td><td>rs1</td><td>011</td><td>imm[4:0]</td><td>0100011</td><td>SD rs2, offset(rs1)</td></tr>
<tr><td colspan="3">000000</td><td colspan="3">shamt6</td><td>rs1</td><td>001</td><td>rd</td><td>0010011</td><td>SLLI rd, rs1, imm</td></tr>
<tr><td colspan="3">000000</td><td colspan="3">shamt6</td><td>rs1</td><td>101</td><td>rd</td><td>0010011</td><td>SRLI rd, rs1, imm</td></tr>
<tr><td colspan="3">010000</td><td colspan="3">shamt6</td><td>rs1</td><td>101</td><td>rd</td><td>0010011</td><td>SRAI rd, rs1, imm</td></tr>
<tr><td colspan="6">imm[11:0]</td><td>rs1</td><td>000</td><td>rd</td><td>0011011</td><td>ADDIW rd, rs1, imm</td></tr>
<tr><td colspan="4">0000000</td><td colspan="2">shamt5</td><td>rs1</td><td>001</td><td>rd</td><td>0011011</td><td>SLLIW rd, rs1, imm</td></tr>
<tr><td colspan="4">0000000</td><td colspan="2">shamt5</td><td>rs1</td><td>101</td><td>rd</td><td>0011011</td><td>SRLIW rd, rs1, imm</td></tr>
<tr><td colspan="4">0100000</td><td colspan="2">shamt5</td><td>rs1</td><td>101</td><td>rd</td><td>0011011</td><td>SRAIW rd, rs1, imm</td></tr>
</table>

31	25	24 20	19 15	14 12	11 7	6 0	
funct5	funct2	rs2	rs1	funct3	rd	opcode	**Type-R**

RV64I Base Integer Instruction Set (in addition to RV32I) contd

00000	00	rs2	rs1	000	rd	0111011	ADDW rd, rs1, rs2
01000	00	rs2	rs1	000	rd	0111011	SUBW rd, rs1, rs2
00000	00	rs2	rs1	001	rd	0111011	SLLW rd, rs1, rs2
00000	00	rs2	rs1	101	rd	0111011	SRLW rd, rs1, rs2
01000	00	rs2	rs1	101	rd	0111011	SRAW rd, rs1, rs2

RV32M Standard Extension for Integer Multiply and Divide

00000	01	rs2	rs1	000	rd	0110011	MUL rd, rs1, rs2
00000	01	rs2	rs1	001	rd	0110011	MULH rd, rs1, rs2
00000	01	rs2	rs1	010	rd	0110011	MULHSU rd, rs1, rs2
00000	01	rs2	rs1	011	rd	0110011	MULHU rd, rs1, rs2
00000	01	rs2	rs1	100	rd	0110011	DIV rd, rs1, rs2
00000	01	rs2	rs1	101	rd	0110011	DIVU rd, rs1, rs2
00000	01	rs2	rs1	110	rd	0110011	REM rd, rs1, rs2
00000	01	rs2	rs1	111	rd	0110011	REMU rd, rs1, rs2

RV64M Standard Extension for Integer Multiply and Divide (in addition to RV32M)

00000	01	rs2	rs1	000	rd	0111011	MULW rd, rs1, rs2
00000	01	rs2	rs1	100	rd	0111011	DIVW rd, rs1, rs2
00000	01	rs2	rs1	101	rd	0111011	DIVUW rd, rs1, rs2
00000	01	rs2	rs1	110	rd	0111011	REMW rd, rs1, rs2
00000	01	rs2	rs1	111	rd	0111011	REMUW rd, rs1, rs2

RV32A Standard Extension for Atomic Instructions

00010	aq	rl	00000	rs1	010	rd	0101111	LR.W aqrl, rd, (rs1)
00011	aq	rl	rs2	rs1	010	rd	0101111	SC.W aqrl, rd, rs2, (rs1)
00001	aq	rl	rs2	rs1	010	rd	0101111	AMOSWAP.W aqrl, rd, rs2, (rs1)
00000	aq	rl	rs2	rs1	010	rd	0101111	AMOADD.W aqrl, rd, rs2, (rs1)
00100	aq	rl	rs2	rs1	010	rd	0101111	AMOXOR.W aqrl, rd, rs2, (rs1)
01000	aq	rl	rs2	rs1	010	rd	0101111	AMOOR.W aqrl, rd, rs2, (rs1)
01100	aq	rl	rs2	rs1	010	rd	0101111	AMOAND.W aqrl, rd, rs2, (rs1)
10000	aq	rl	rs2	rs1	010	rd	0101111	AMOMIN.W aqrl, rd, rs2, (rs1)
10100	aq	rl	rs2	rs1	010	rd	0101111	AMOMAX.W aqrl, rd, rs2, (rs1)
11000	aq	rl	rs2	rs1	010	rd	0101111	AMOMINU.W aqrl, rd, rs2, (rs1)
11100	aq	rl	rs2	rs1	010	rd	0101111	AMOMAXU.W aqrl, rd, rs2, (rs1)

RV64A Standard Extension for Atomic Instructions (in addition to RV32A)

00010	aq	rl	00000	rs1	011	rd	0101111	LR.D aqrl, rd, (rs1)
00011	aq	rl	rs2	rs1	011	rd	0101111	SC.D aqrl, rd, rs2, (rs1)
00001	aq	rl	rs2	rs1	011	rd	0101111	AMOSWAP.D aqrl, rd, rs2, (rs1)
00000	aq	rl	rs2	rs1	011	rd	0101111	AMOADD.D aqrl, rd, rs2, (rs1)
00100	aq	rl	rs2	rs1	011	rd	0101111	AMOXOR.D aqrl, rd, rs2, (rs1)
01000	aq	rl	rs2	rs1	011	rd	0101111	AMOOR.D aqrl, rd, rs2, (rs1)
01100	aq	rl	rs2	rs1	011	rd	0101111	AMOAND.D aqrl, rd, rs2, (rs1)
10000	aq	rl	rs2	rs1	011	rd	0101111	AMOMIN.D aqrl, rd, rs2, (rs1)
10100	aq	rl	rs2	rs1	011	rd	0101111	AMOMAX.D aqrl, rd, rs2, (rs1)
11000	aq	rl	rs2	rs1	011	rd	0101111	AMOMINU.D aqrl, rd, rs2, (rs1)
11100	aq	rl	rs2	rs1	011	rd	0101111	AMOMAXU.D aqrl, rd, rs2, (rs1)

31	25	24 20	19 15	14 12	11 7	6 0	
imm[11:0]			rs1	funct3	rd	opcode	**Type-I**
imm[11:5]		rs2	rs1	funct3	imm[4:0]	opcode	**Type-S**
rs3	funct2	rs2	rs1	funct3	rd	opcode	**Type-R4**
funct5	funct2	rs2	rs1	funct3	rd	opcode	**Type-R**

RV32S Standard Extension for Supervisor-level Instructions

0000000		00000	00000	000	00000	1110011	ECALL
0000000		00001	00000	000	00000	1110011	EBREAK
0000000		00010	00000	000	00000	1110011	URET
0001000		00000	00000	000	00000	1110011	SRET
0010000		00010	00000	000	00000	1110011	HRET
0011000		00010	00000	000	00000	1110011	MRET
0111101		10010	00000	000	00000	1110011	DRET
0001000		00001	rs1	000	00000	1110011	SFENCE.VM
0001000		00010	00000	000	00000	1110011	WFI
csr			rs1	001	rd	1110011	CSRRW rd, csr, rs1
csr			rs1	010	rd	1110011	CSRRS rd, csr, rs1
csr			rs1	011	rd	1110011	CSRRC rd, csr, rs1
csr			zimm	101	rd	1110011	CSRRWI rd, csr, zimm
csr			zimm	110	rd	1110011	CSRRSI rd, csr, zimm
csr			zimm	111	rd	1110011	CSRRCI rd, csr, zimm

RV32F Standard Extension for Single-Precision Floating-Point

imm[11:0]			rs1	010	frd	0000111	FLW frd, offset(rs1)
imm[11:5]		frs2	rs1	010	imm[4:0]	0100111	FSW frs2, offset(rs1)
frs3	00	frs2	frs1	rm	frd	1000011	FMADD.S rm, frd, frs1, frs2, frs3
frs3	00	frs2	frs1	rm	frd	1000111	FMSUB.S rm, frd, frs1, frs2, frs3
frs3	00	frs2	frs1	rm	frd	1001011	FNMSUB.S rm, frd, frs1, frs2, frs3
frs3	00	frs2	frs1	rm	frd	1001111	FNMADD.S rm, frd, frs1, frs2, frs3
00000	00	frs2	frs1	rm	frd	1010011	FADD.S rm, frd, frs1, frs2
00001	00	frs2	frs1	rm	frd	1010011	FSUB.S rm, frd, frs1, frs2
00010	00	frs2	frs1	rm	frd	1010011	FMUL.S rm, frd, frs1, frs2
00011	00	frs2	frs1	rm	frd	1010011	FDIV.S rm, frd, frs1, frs2
00100	00	frs2	frs1	000	frd	1010011	FSGNJ.S frd, frs1, frs2
00100	00	frs2	frs1	001	frd	1010011	FSGNJN.S frd, frs1, frs2
00100	00	frs2	frs1	010	frd	1010011	FSGNJX.S frd, frs1, frs2
00101	00	frs2	frs1	000	frd	1010011	FMIN.S frd, frs1, frs2
00101	00	frs2	frs1	001	frd	1010011	FMAX.S frd, frs1, frs2
01011	00	00000	frs1	rm	frd	1010011	FSQRT.S rm, frd, frs1
10100	00	frs2	frs1	000	rd	1010011	FLE.S rd, frs1, frs2
10100	00	frs2	frs1	001	rd	1010011	FLT.S rd, frs1, frs2
10100	00	frs2	frs1	010	rd	1010011	FEQ.S rd, frs1, frs2
11000	00	00000	frs1	rm	rd	1010011	FCVT.W.S rm, rd, frs1
11000	00	00001	frs1	rm	rd	1010011	FCVT.WU.S rm, rd, frs1
11010	00	00000	rs1	rm	frd	1010011	FCVT.S.W rm, frd, rs1
11010	00	00001	rs1	rm	frd	1010011	FCVT.S.WU rm, frd, rs1
11100	00	00000	frs1	000	rd	1010011	FMV.X.S rd, frs1
11100	00	00000	frs1	001	rd	1010011	FCLASS.S rd, frs1
11110	00	00000	rs1	000	frd	1010011	FMV.S.X frd, rs1

RV64F Standard Extension for Single-Precision Floating-Point (in addition to RV32F)

11000	00	00010	frs1	rm	rd	1010011	FCVT.L.S rm, rd, frs1
11000	00	00011	frs1	rm	rd	1010011	FCVT.LU.S rm, rd, frs1
11010	00	00010	rs1	rm	frd	1010011	FCVT.S.L rm, frd, rs1
11010	00	00011	rs1	rm	frd	1010011	FCVT.S.LU rm, frd, rs1

31	25	24 20	19 15	14 12	11 7	6 0	
imm[11:0]			rs1	funct3	rd	opcode	**Type-I**
imm[11:5]		rs2	rs1	funct3	imm[4:0]	opcode	**Type-S**
rs3	funct2	rs2	rs1	funct3	rd	opcode	**Type-R4**
funct5	funct2	rs2	rs1	funct3	rd	opcode	**Type-R**

RV32D Standard Extension for Double-Precision Floating-Point

imm[11:0]			rs1	011	frd	0000111	FLD frd, offset(rs1)
imm[11:5]		frs2	rs1	011	imm[4:0]	0100111	FSD frs2, offset(rs1)
frs3	01	frs2	frs1	rm	frd	1000011	FMADD.D rm, frd, frs1, frs2, frs3
frs3	01	frs2	frs1	rm	frd	1000111	FMSUB.D rm, frd, frs1, frs2, frs3
frs3	01	frs2	frs1	rm	frd	1001011	FNMSUB.D rm, frd, frs1, frs2, frs3
frs3	01	frs2	frs1	rm	frd	1001111	FNMADD.D rm, frd, frs1, frs2, frs3
00000	01	frs2	frs1	rm	frd	1010011	FADD.D rm, frd, frs1, frs2
00001	01	frs2	frs1	rm	frd	1010011	FSUB.D rm, frd, frs1, frs2
00010	01	frs2	frs1	rm	frd	1010011	FMUL.D rm, frd, frs1, frs2
00011	01	frs2	frs1	rm	frd	1010011	FDIV.D rm, frd, frs1, frs2
00100	01	frs2	frs1	000	frd	1010011	FSGNJ.D frd, frs1, frs2
00100	01	frs2	frs1	001	frd	1010011	FSGNJN.D frd, frs1, frs2
00100	01	frs2	frs1	010	frd	1010011	FSGNJX.D frd, frs1, frs2
00101	01	frs2	frs1	000	frd	1010011	FMIN.D frd, frs1, frs2
00101	01	frs2	frs1	001	frd	1010011	FMAX.D frd, frs1, frs2
01000	00	00001	frs1	rm	frd	1010011	FCVT.S.D rm, frd, frs1
01000	01	00000	frs1	rm	frd	1010011	FCVT.D.S rm, frd, frs1
01011	01	00000	frs1	rm	frd	1010011	FSQRT.D rm, frd, frs1
10100	01	frs2	frs1	000	rd	1010011	FLE.D rd, frs1, frs2
10100	01	frs2	frs1	001	rd	1010011	FLT.D rd, frs1, frs2
10100	01	frs2	frs1	010	rd	1010011	FEQ.D rd, frs1, frs2
11000	01	00000	frs1	rm	rd	1010011	FCVT.W.D rm, rd, frs1
11000	01	00001	frs1	rm	rd	1010011	FCVT.WU.D rm, rd, frs1
11010	01	00000	rs1	rm	frd	1010011	FCVT.D.W rm, frd, rs1
11010	01	00001	rs1	rm	frd	1010011	FCVT.D.WU rm, frd, rs1
11100	01	00000	frs1	001	rd	1010011	FCLASS.D rd, frs1

RV64D Standard Extension for Double-Precision Floating-Point (in addition to RV32D)

11000	01	00010	frs1	rm	rd	1010011	FCVT.L.D rm, rd, frs1
11000	01	00011	frs1	rm	rd	1010011	FCVT.LU.D rm, rd, frs1
11100	01	00000	frs1	000	rd	1010011	FMV.X.D rd, frs1
11010	01	00010	rs1	rm	frd	1010011	FCVT.D.L rm, frd, rs1
11010	01	00011	rs1	rm	frd	1010011	FCVT.D.LU rm, frd, rs1
11110	01	00000	rs1	000	frd	1010011	FMV.D.X frd, rs1

<table>
<tr><th>15 13</th><th colspan="2">12 10</th><th>9 7</th><th>6 5</th><th>4 2</th><th>1 0</th><th></th></tr>
<tr><td>funct3</td><td colspan="4">imm8</td><td>rd’</td><td>op</td><td><b>Type-CIW</b></td></tr>
<tr><td>funct3</td><td colspan="2">imm3</td><td>rs1’</td><td>imm2</td><td>rd’</td><td>op</td><td><b>Type-CL</b></td></tr>
<tr><td>funct3</td><td colspan="2">imm3</td><td>rs1’</td><td>imm2</td><td>rs2’</td><td>op</td><td><b>Type-CS</b></td></tr>
<tr><td>funct3</td><td>imm1</td><td colspan="2">rd/rs1</td><td colspan="2">imm5</td><td>op</td><td><b>Type-CI</b></td></tr>
<tr><td>funct3</td><td colspan="5">imm11</td><td>op</td><td><b>Type-CJ</b></td></tr>
<tr><td>funct3</td><td colspan="2">imm3</td><td>rs1’</td><td colspan="2">imm5</td><td>op</td><td><b>Type-CB</b></td></tr>
<tr><td colspan="2">funct4</td><td colspan="2">rd/rs1</td><td colspan="2">rs2</td><td>op</td><td><b>Type-CR</b></td></tr>
<tr><td>funct3</td><td colspan="3">imm6</td><td colspan="2">rs2</td><td>op</td><td><b>Type-CSS</b></td></tr>
</table>

RV32C Standard Extension for Compressed Instructions

<table>
<tr><td>000</td><td colspan="4">imm[5:4|9:6|2|3]</td><td>rd’</td><td>00</td><td>C.ADDI4SPN rd, rs1, imm</td></tr>
<tr><td>001</td><td colspan="2">imm[5:3]</td><td>rs1’</td><td>imm[7:6]</td><td>frd’</td><td>00</td><td>C.FLD frd, offset(rs1)</td></tr>
<tr><td>010</td><td colspan="2">imm[5:3]</td><td>rs1’</td><td>imm[2|6]</td><td>rd’</td><td>00</td><td>C.LW rd, offset(rs1)</td></tr>
<tr><td>011</td><td colspan="2">imm[5:3]</td><td>rs1’</td><td>imm[2|6]</td><td>frd’</td><td>00</td><td>C.FLW frd, offset(rs1)</td></tr>
<tr><td>101</td><td colspan="2">imm[5:3]</td><td>rs1’</td><td>imm[7:6]</td><td>frs2’</td><td>00</td><td>C.FSD frs2, offset(rs1)</td></tr>
<tr><td>110</td><td colspan="2">imm[5:3]</td><td>rs1’</td><td>imm[2|6]</td><td>rs2’</td><td>00</td><td>C.SW rs2, offset(rs1)</td></tr>
<tr><td>111</td><td colspan="2">imm[5:3]</td><td>rs1’</td><td>imm[2|6]</td><td>frs2’</td><td>00</td><td>C.FSW frs2, offset(rs1)</td></tr>
<tr><td>000</td><td>0</td><td colspan="2">00000</td><td colspan="2">00000</td><td>01</td><td>C.NOP</td></tr>
<tr><td>000</td><td>imm[5]</td><td colspan="2">rs1/rd≠ 0</td><td colspan="2">imm[4:0]</td><td>01</td><td>C.ADDI rd, rs1, imm</td></tr>
<tr><td>001</td><td colspan="5">imm[11|4|9:8|10|6|7|3:1|5]</td><td>01</td><td>C.JAL rd, offset</td></tr>
<tr><td>010</td><td>imm[5]</td><td colspan="2">rs1/rd≠ 0</td><td colspan="2">imm[4:0]</td><td>01</td><td>C.LI rd, rs1, imm</td></tr>
<tr><td>011</td><td>imm[9]</td><td colspan="2">rs1/rd=2</td><td colspan="2">imm[4|6|8:7|5]</td><td>01</td><td>C.ADDI16SP rd, rs1, imm</td></tr>
<tr><td>011</td><td>imm[17]</td><td colspan="2">rd≠ {0, 2}</td><td colspan="2">imm[16:12]</td><td>01</td><td>C.LUI rd, imm</td></tr>
<tr><td>100</td><td colspan="2">000</td><td>rs1’/rd’</td><td colspan="2">imm[4:0]</td><td>01</td><td>C.SRLI rd, rs1, imm</td></tr>
<tr><td>100</td><td colspan="2">001</td><td>rs1’/rd’</td><td colspan="2">imm[4:0]</td><td>01</td><td>C.SRAI rd, rs1, imm</td></tr>
<tr><td>100</td><td>imm[5]</td><td>10</td><td>rs1’/rd’</td><td colspan="2">imm[4:0]</td><td>01</td><td>C.ANDI rd, rs1, imm</td></tr>
<tr><td>100</td><td colspan="2">011</td><td>rs1’/rd’</td><td>00</td><td>rs2’</td><td>01</td><td>C.SUB rd, rs1, rs2</td></tr>
<tr><td>100</td><td colspan="2">011</td><td>rs1’/rd’</td><td>01</td><td>rs2’</td><td>01</td><td>C.XOR rd, rs1, rs2</td></tr>
<tr><td>100</td><td colspan="2">011</td><td>rs1’/rd’</td><td>10</td><td>rs2’</td><td>01</td><td>C.OR rd, rs1, rs2</td></tr>
<tr><td>100</td><td colspan="2">011</td><td>rs1’/rd’</td><td>11</td><td>rs2’</td><td>01</td><td>C.AND rd, rs1, rs2</td></tr>
<tr><td>100</td><td colspan="2">111</td><td>rs1’/rd’</td><td>00</td><td>rs2’</td><td>01</td><td>C.SUBW rd, rs1, rs2</td></tr>
<tr><td>100</td><td colspan="2">111</td><td>rs1’/rd’</td><td>01</td><td>rs2’</td><td>01</td><td>C.ADDW rd, rs1, rs2</td></tr>
<tr><td>101</td><td colspan="5">imm[11|4|9:8|10|6|7|3:1|5]</td><td>01</td><td>C.J rd, offset</td></tr>
<tr><td>110</td><td colspan="2">imm[8|4:3]</td><td>rs1’</td><td colspan="2">imm[7:6|2:1|5]</td><td>01</td><td>C.BEQZ rs1, rs2, offset</td></tr>
<tr><td>111</td><td colspan="2">imm[8|4:3]</td><td>rs1’</td><td colspan="2">imm[7:6|2:1|5]</td><td>01</td><td>C.BNEZ rs1, rs2, offset</td></tr>
<tr><td>000</td><td>0</td><td colspan="2">rs1/rd≠ 0</td><td colspan="2">imm[4:0]</td><td>10</td><td>C.SLLI rd, rs1, imm</td></tr>
<tr><td>001</td><td>imm[5]</td><td colspan="2">frd</td><td colspan="2">imm[4:3|8:6]</td><td>10</td><td>C.FLDSP frd, offset(rs1)</td></tr>
<tr><td>010</td><td>imm[5]</td><td colspan="2">rd≠ 0</td><td colspan="2">imm[4:2|7:6]</td><td>10</td><td>C.LWSP rd, offset(rs1)</td></tr>
<tr><td>011</td><td>imm[5]</td><td colspan="2">frd</td><td colspan="2">imm[4:2|7:6]</td><td>10</td><td>C.FLWSP frd, offset(rs1)</td></tr>
<tr><td>100</td><td>rd,</td><td colspan="2">rs1≠ 0</td><td colspan="2">00000</td><td>10</td><td>C.JR rd, rs1, offset</td></tr>
<tr><td colspan="2">1000</td><td colspan="2">rd≠ 0</td><td colspan="2">rs2≠0</td><td>10</td><td>C.MV rd, rs1, rs2</td></tr>
<tr><td>100</td><td>1</td><td colspan="2">00000</td><td colspan="2">00000</td><td>10</td><td>C.EBREAK</td></tr>
<tr><td>100</td><td>rd,</td><td colspan="2">rs1≠ 0</td><td colspan="2">00000</td><td>10</td><td>C.JALR rd, rs1, offset</td></tr>
<tr><td colspan="2">1001</td><td colspan="2">rs1/rd≠ 0</td><td colspan="2">rs2≠0</td><td>10</td><td>C.ADD rd, rs1, rs2</td></tr>
<tr><td>101</td><td colspan="3">imm[5:3|8:6]</td><td colspan="2">frs2</td><td>10</td><td>C.FSDSP frs2, offset(rs1)</td></tr>
<tr><td>110</td><td colspan="3">imm[5:2|7:6]</td><td colspan="2">rs2</td><td>10</td><td>C.SWSP rs2, offset(rs1)</td></tr>
<tr><td>111</td><td colspan="3">imm[5:2|7:6]</td><td colspan="2">frs2</td><td>10</td><td>C.FSWSP frs2, offset(rs1)</td></tr>
</table>

RV64C Standard Extension for Compressed Instructions (in addition to RV32C)

<table>
<tr><td>011</td><td colspan="2">imm[5:3]</td><td>rs1’</td><td>imm[7:6]</td><td>rd’</td><td>00</td><td>C.LD rd, offset(rs1)</td></tr>
<tr><td>111</td><td colspan="2">imm[5:3]</td><td>rs1’</td><td>imm[7:6]</td><td>rs2’</td><td>00</td><td>C.SD rs2, offset(rs1)</td></tr>
<tr><td>001</td><td>imm[5]</td><td colspan="2">rs1/rd≠ 0</td><td colspan="2">imm[4:0]</td><td>01</td><td>C.ADDIW rd, rs1, imm</td></tr>
<tr><td>100</td><td>imm[5]</td><td>00</td><td>rs1’/rd’</td><td colspan="2">imm[4:0]</td><td>01</td><td>C.SRLI rd, rs1, imm</td></tr>
<tr><td>100</td><td>imm[5]</td><td>01</td><td>rs1’/rd’</td><td colspan="2">imm[4:0]</td><td>01</td><td>C.SRAI rd, rs1, imm</td></tr>
<tr><td>000</td><td>imm[5]</td><td colspan="2">rs1/rd≠ 0</td><td colspan="2">imm[4:0]</td><td>10</td><td>C.SLLI rd, rs1, imm</td></tr>
<tr><td>011</td><td>imm[5]</td><td colspan="2">rd≠ 0</td><td colspan="2">imm[4:3|8:6]</td><td>10</td><td>C.LDSP rd, offset(rs1)</td></tr>
<tr><td>111</td><td colspan="3">imm[5:3|8:6]</td><td colspan="2">rs2</td><td>10</td><td>C.SDSP rs2, offset(rs1)</td></tr>
</table>