RISCV instructions	Instruction as in objdump	Format type	32-bit code format	32-bit binary code	32-bit hexadecimal code
Subtract (SUB)	sub a2, a2, a0	R	funct7[31:25] | rs2[24:20] | rs1[19:15] | funct3[14:12] | opcode[6:0]	0000000 00000 01100 000 01000 0110011	0x40A60633
Add Immediate (ADDI)	addi a2, a2, 954	I	imm[11:0] | rs1[19:15] | funct3[14:12] | rd[11:7] | opcode[6:0]	1110111010 01100 000 01100 0010011	0x3BA60613
Store Double (SD)	sd ra, 8(sp)	S	imm[11:5] | rs2[24:20] | rs1[19:15] | funct3[14:12] | opcode[6:0]	0000001 11100 10000 010 01101 0100011	0x00113423
Branch if Less Than Unsigned (BLTU)	bltu a4, a3, 10310	B	imm[12] | imm[10:5] | rs2[24:20] | rs1[19:15] | funct3[14:12] | imm[4:1] | imm[11] | opcode[6:0]	1111 1110 1101 0111 0110 0010 1110 0011	0xFED76AE3
Load Upper Immediate (LUI)	lui a3, 0xffffe	U	imm[31:12] | rd[11:7] | opcode[6:0]	111111111111 11111 01101 0110111	0xFFFFE6B7
Jump and Link (JAL)	jal ra, 12c94	J	imm[20] | imm[10:1] | imm[11] | imm[19:12] | rd[11:7] | opcode[6:0]	000110001011 00101 000 00101 1101111	0x628020EF
Subtract Word (SUBW)	subw a5, a5, s11	R	funct7[31:25] | rs2[24:20] | rs1[19:15] | funct3[14:12] | rd[11:7] | opcode[6:0]	0000000 01011 00101 000 01011 0111011	0x41B787BB
Load Immediate (LI)	li a1, 100	I	imm[11:0] | rs1[19:15] | funct3[14:12] | rd[11:7] | opcode[6:0]	000000110100 00000 000 00001 0010011	0x06400593
Store Half (SH)	sh a5, 224(sp)	S	imm[11:5] | rs2[24:20] | rs1[19:15] | funct3[14:12] | opcode[6:0]	000000 11100 00101 001 11110 0100011	0x0EF11023
Branch if Not Equal to Zero (BNEZ)	bnez a5, 11ae0	B	imm[12] | imm[10:5] | rs2[24:20] | rs1[19:15] | funct3[14:12] | imm[4:1] | imm[11] | opcode[6:0]	000000000000 00101 001 11101 1100011	0x0E079EE3
Add Upper Immediate to PC (AUIPC)	auipc a3, 0x10	U	imm[31:12] | rd[11:7] | opcode[6:0]	000000000001 00011 01101 0110111	0x00010697
LH (Load Halfword)	lh a5,18(a5)	I	imm[11:0] | rs1[19:15] | funct3[14:12] | rd[11:7] | opcode[6:0]	000000010010 01111 001 01111 0000011	0x01279783
Add Immediate (ADDI)	addi t1, sp, 400	I	imm[11:0] | rs1[19:15] | funct3[14:12] | rd[11:7] | opcode[6:0]	000000011001 00000 000 10111 0010011	0x19010313
ORI (Bitwise OR Immediate)	ori s1, s1,16	I	imm[11:0] | rs1[19:15] | funct3[14:12] | rd[11:7] | opcode[6:0]	000000010000 01001 110 01001 0010011	0x0104e493
BNE (Branch if Not equal)	bne a5, a3, 10538	B	imm[12|10:5] | rs2[24:20] | rs1[19:15] | funct3[14:12] | imm[4:1|11] | opcode[6:0]	000001 01111 10001 001 0010 1100011	0xead798e3