RISC-V Base Instruction Set (RV32I) - Common Pseudoinstructions Included

Integer Arithmetic Instructions

add	: Add two registers
sub	: Subtract registers
sll	: Shift left logical
srl	: Shift right logical
sra	: Shift right arithmetic
and	: Bitwise AND
or	: Bitwise OR
xor	: Bitwise XOR
slt	: Set if less than (signed)
sltu	: Set if less than (unsigned)

Immediate Arithmetic Instructions

addi	: Add immediate
andi	: Bitwise AND with immediate
ori	: Bitwise OR with immediate
xori	: Bitwise XOR with immediate
slti	: Set if less than immediate (signed)
sltiu	: Set if less than immediate (unsigned)
slli	: Shift left logical immediate
srli	: Shift right logical immediate
srai	: Shift right arithmetic immediate
lui	: Load upper immediate
auipc	: Add upper immediate to PC

Control Flow (Branch & Jump)

beq	: Branch if equal
bne	: Branch if not equal
blt	: Branch if less than (signed)
bge	: Branch if greater or equal (signed)
bltu	: Branch if less than (unsigned)
bgeu	: Branch if greater or equal (unsigned)
jal	: Jump and link

jalr	: Jump and link register

Memory Access Instructions

lw	: Load word (32-bit)
lh	: Load halfword (16-bit, sign-extended)
lhu	: Load halfword unsigned
lb	: Load byte (8-bit, sign-extended)
lbu	: Load byte unsigned
sw	: Store word
sh	: Store halfword
sb	: Store byte

System Instructions

ecall	: Environment call (used for syscalls)
ebreak	: Breakpoint for debugger
fence	: Memory fence (for ordering operations)
fence.i	: Instruction memory fence (flush instruction cache)

Common Pseudoinstructions

li rd, imm	: Load immediate value into register
mv rd, rs	: Move value from one register to another
nop	: No operation
j label	: Jump to label
ret	: Return from function
call label	: Function call (PC-relative)
tail label	: Tail call (like jump to function)
la rd, label	: Load address of label

This set provides a comprehensive foundation for learning and writing RISC-V programs. Pseudoinstructions improve readability and convenience, especially for beginners and teaching tools like RARS and Venus.