Addressing Modes

Means how instructions specify the location of its operands

(1) Immediate

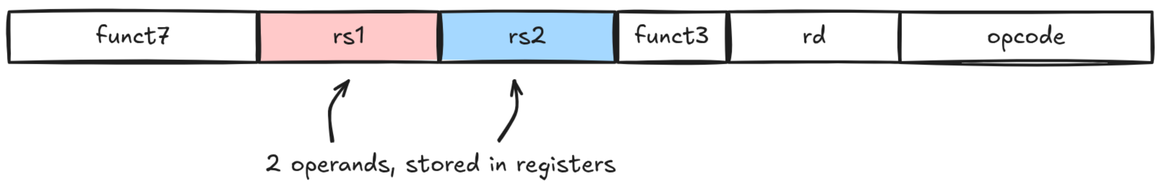
Operand is included as an immediate value in the instruction

Example: addi (U type)

(2) Register

Operand is stored in a register, which is referenced in the instruction

Example: add (R type)



(3) Absolute memory access

Operand is stored in memory. Referenced using the absolute memory address

Can only be done if memory address size is less than instruction size

Not available in RISC-V because both instructions and memory address size are 32-bits.

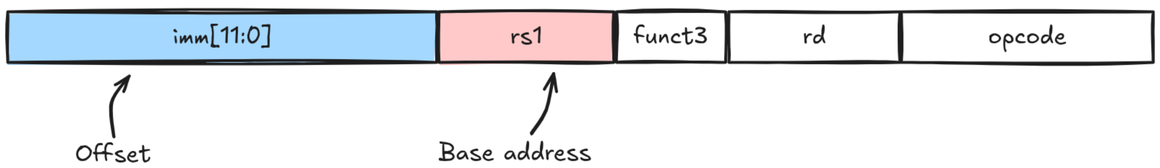
(4) Base + Offset (Displacement)

Operand is stored in memory. Referenced using a base memory address (stored in register) + offset (immediate)

Enables locality

A workaround for exact memory addressing as 32-bit memory address cannot be embedded in 32-bit instruction

Example: lw (I type)



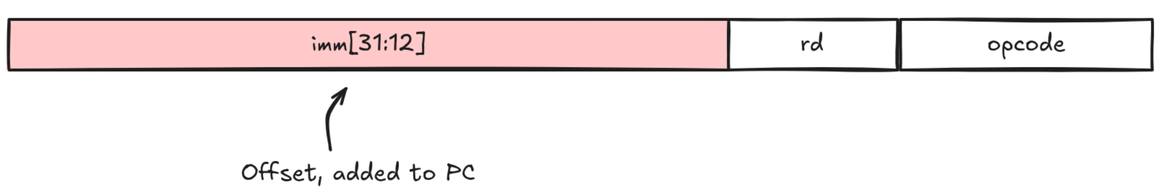
target_memory_address = value_in(rs1) + imm

(5) PC-relative

Operand is stored in memory, relative to the current instruction

Enables position-independent code

Example: auipc (U type)



(6) Register indirect

Operand is stored in memory. Register stores the memory address.

A special case of displacement mode with offset set to 0

Made by Sahithyan K.