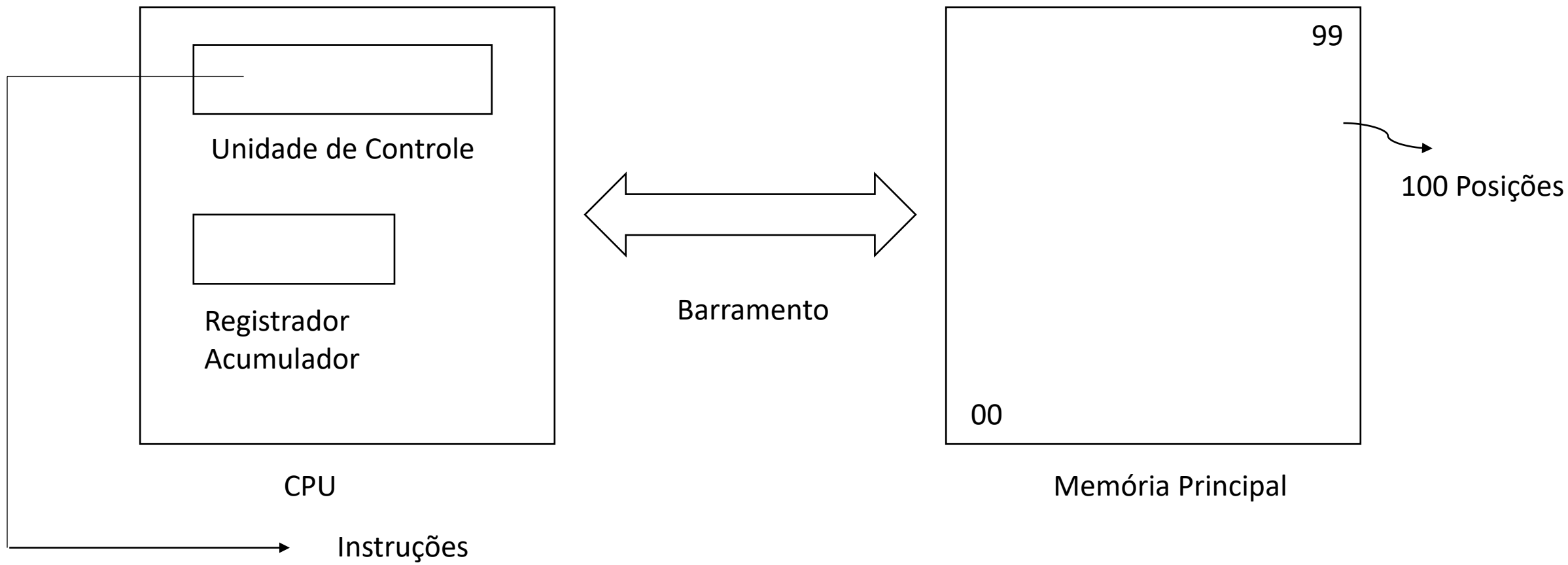


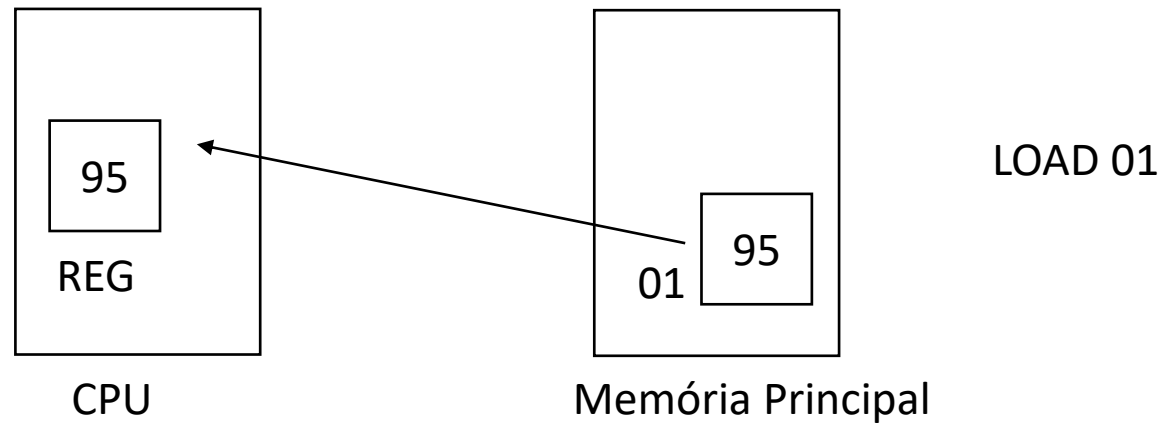
# Pseudo Código

# Uma Arquitetura Simples



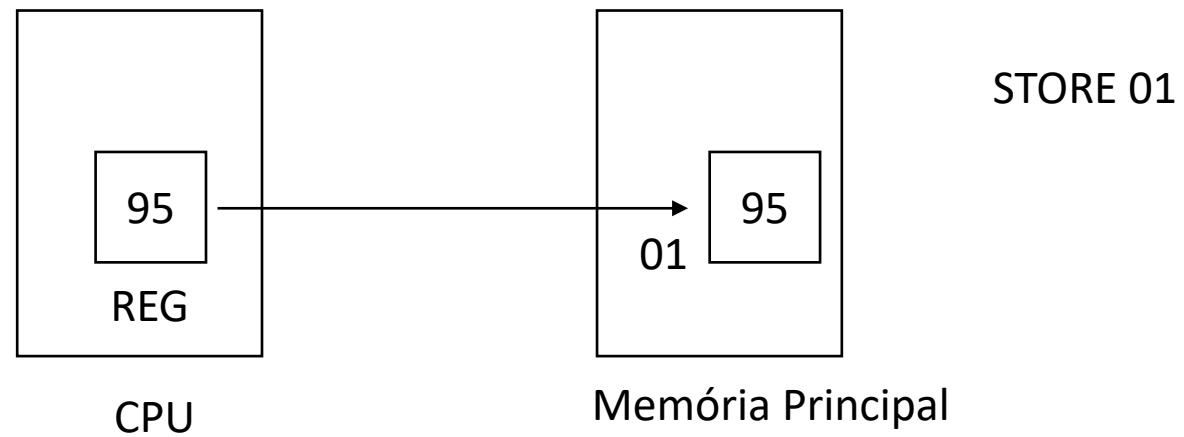
## MOVIMENTAÇÃO ENTRE CPU E MEMÓRIA

- **LOAD N** - carrega os dados da posição de memória N para o registrador.



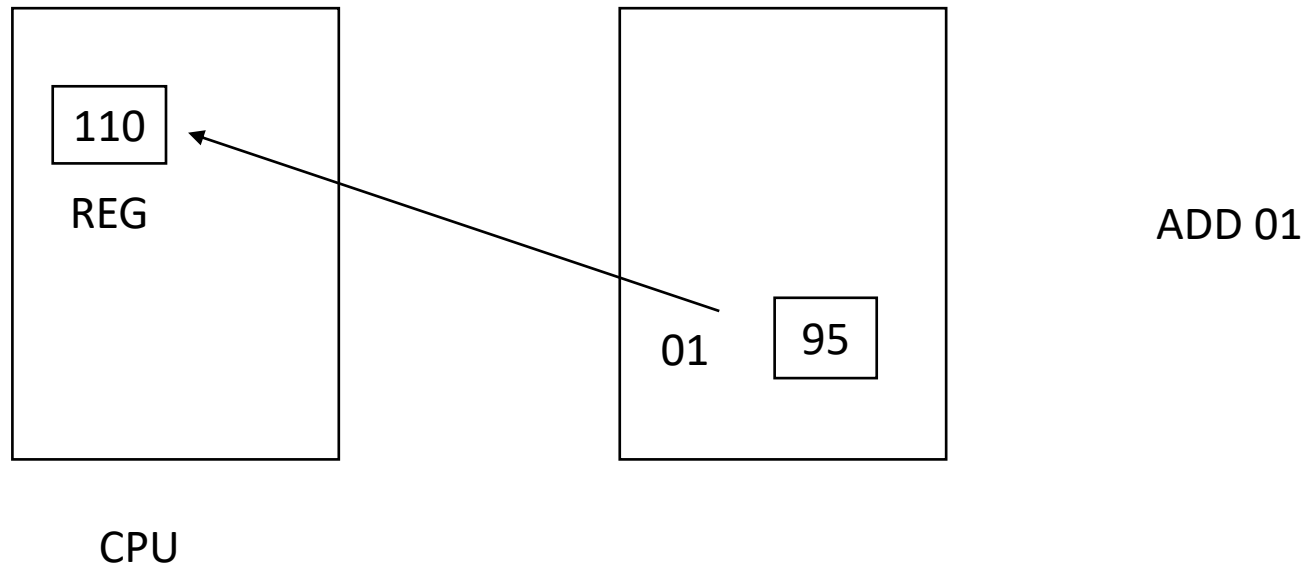
## MOVIMENTAÇÃO ENTRE CPU E MEMÓRIA

- **STORE N** - grava os dados do registrador na posição de memória N.



## OPERAÇÕES ARITMÉTICAS

- **ADD N** - adiciona o valor da posição de memória N ao atual valor do registrador, guardando o resultado no registrador.

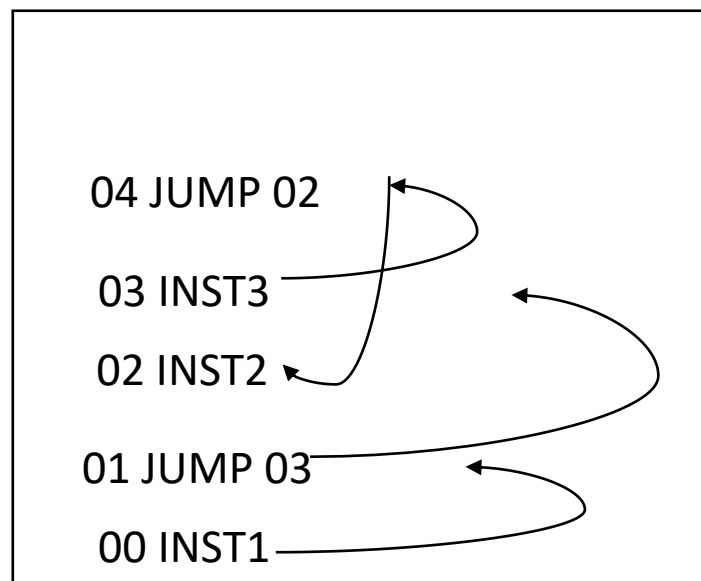


## OPERAÇÕES ARITMÉTICAS

- **SUB N** - subtrai o valor da posição memória N do atual valor do reg, guardando o resultado no registrador
- **MUL N** - multiplica o valor da posição de memória N pelo atual valor do registrador, guardando o resultado no registrador.
- **DIV N** – divide o valor da posição de memória N pelo atual valor do registrador, guardando o resultado no registrador.
- **(ADDI, SUBI, MULI, DIVI) N** - o mesmo que as acima, mas opera diretamente com o valor numérico N (e não a posição de memória).

## DESVIOS NO FLUXO DE EXECUÇÃO

- **JUMP N** - pula para a instrução na posição de memória N, sempre.



Memória Principal

JUMP 03/JUMP 02

## DESVIOS NO FLUXO DE EXECUÇÃO

- **JPOS N** - pula para a instrução na posição de memória N, caso o valor do registrador seja **MAIOR** que zero.
- **JZERO N** - pula para a instrução na posição de memória N, caso o valor do registrador seja **IGUAL** que zero.