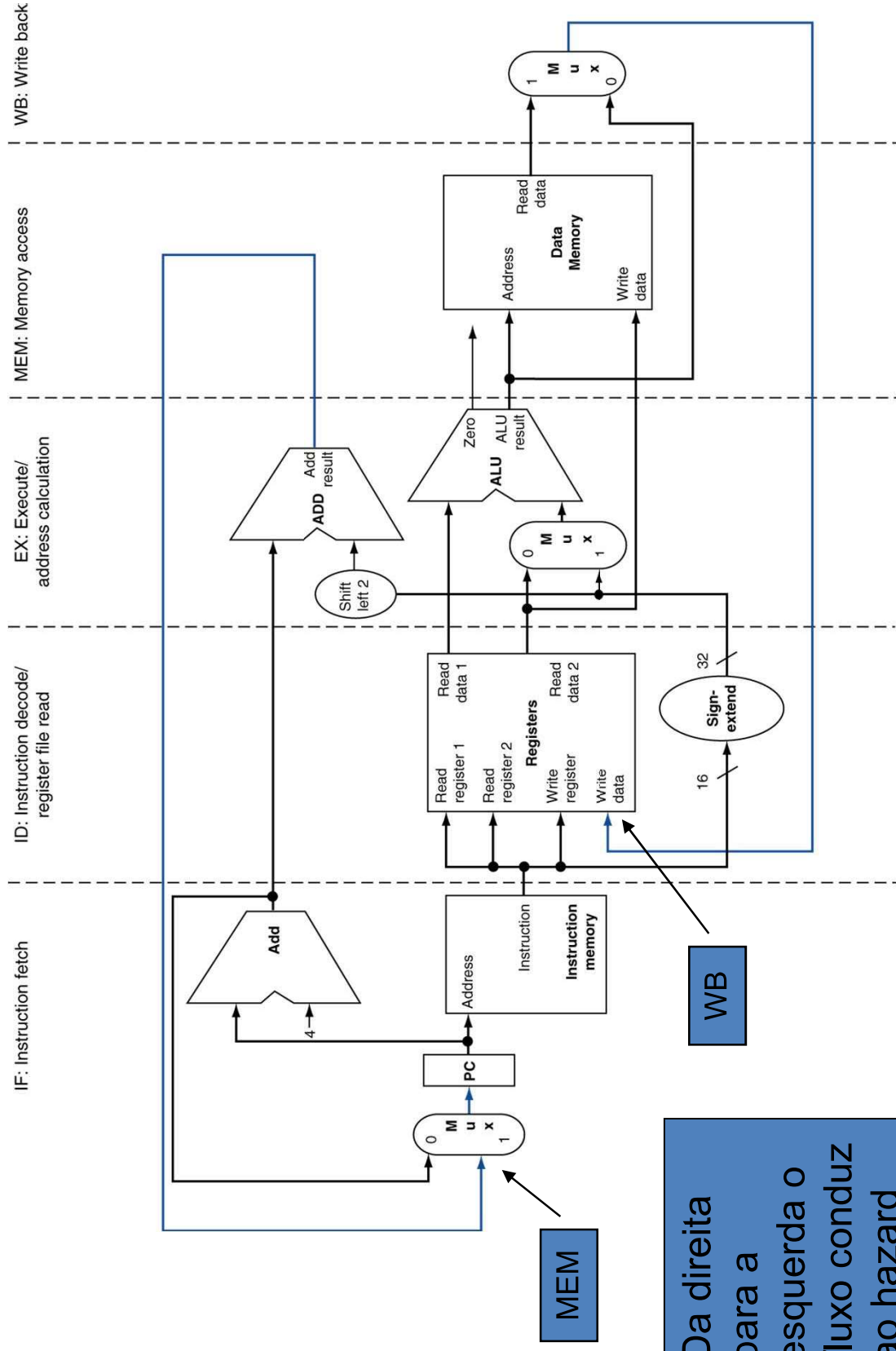


# Resumo do Pipeline

## The BIG Picture

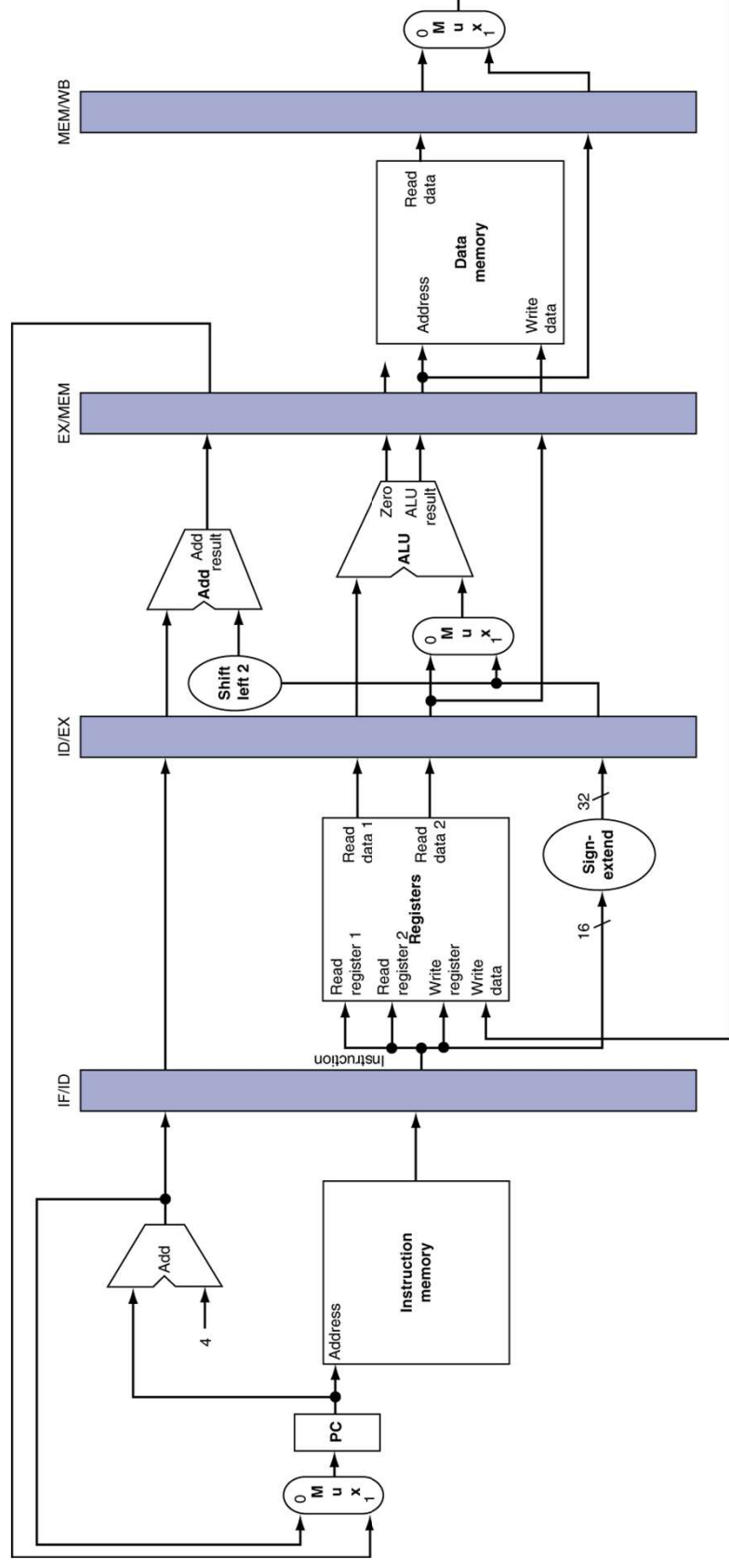
- Pipeline melhora o desempenho pelo aumento do throughput de instrução
  - Executa múltiplas instruções em paralelo
  - Cada instrução tem a mesma latência
- Tópicos de hazards
  - Estrutura, dados, controle
- O projeto do conjunto de instruções afeta a complexidade da implementação do pipeline

## Datapath MIPS com pipeline



# Registradores de Pipeline

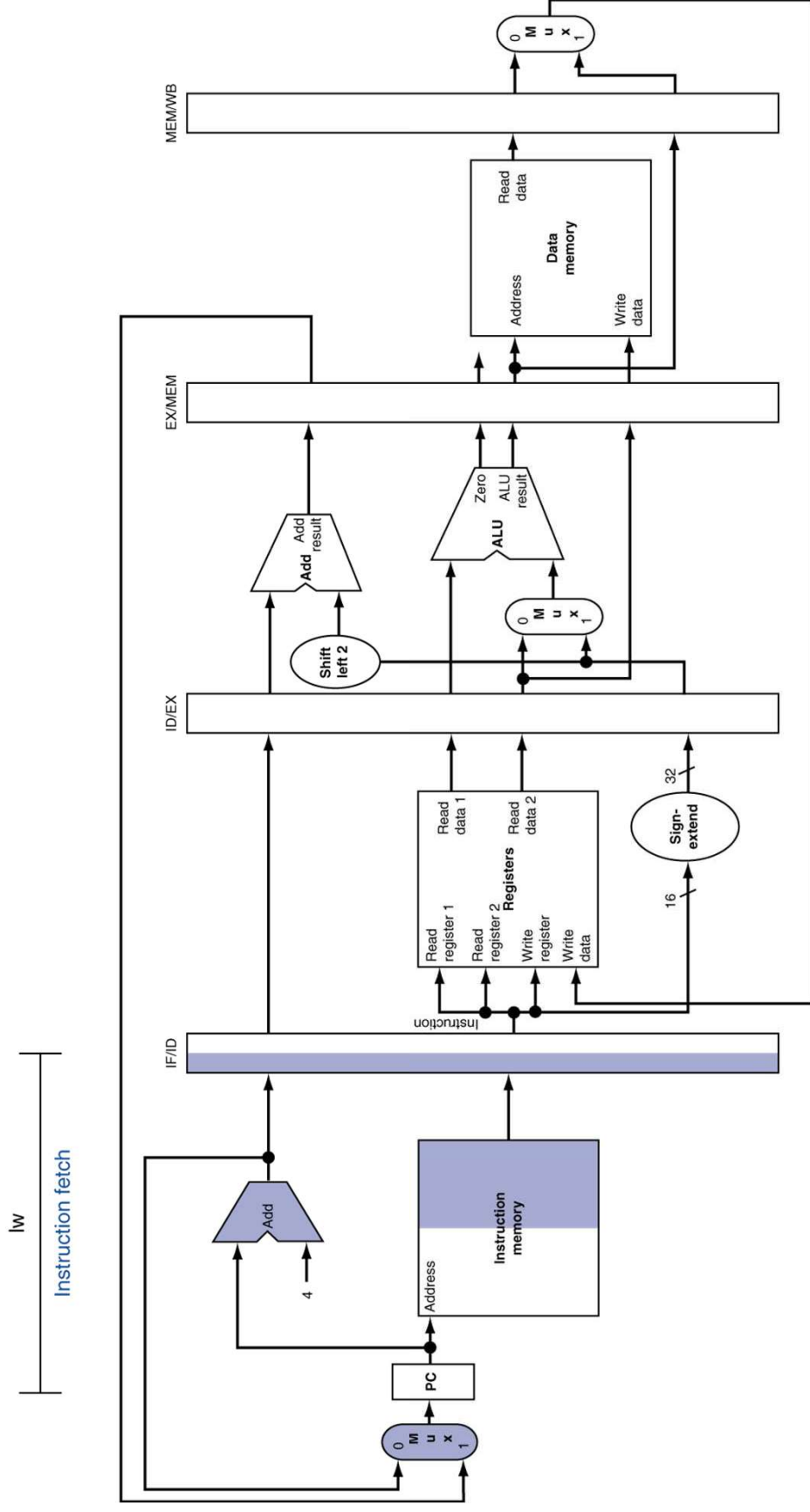
- Necessita registradores entre os estágios
  - Para manter a informação produzida no ciclo anterior



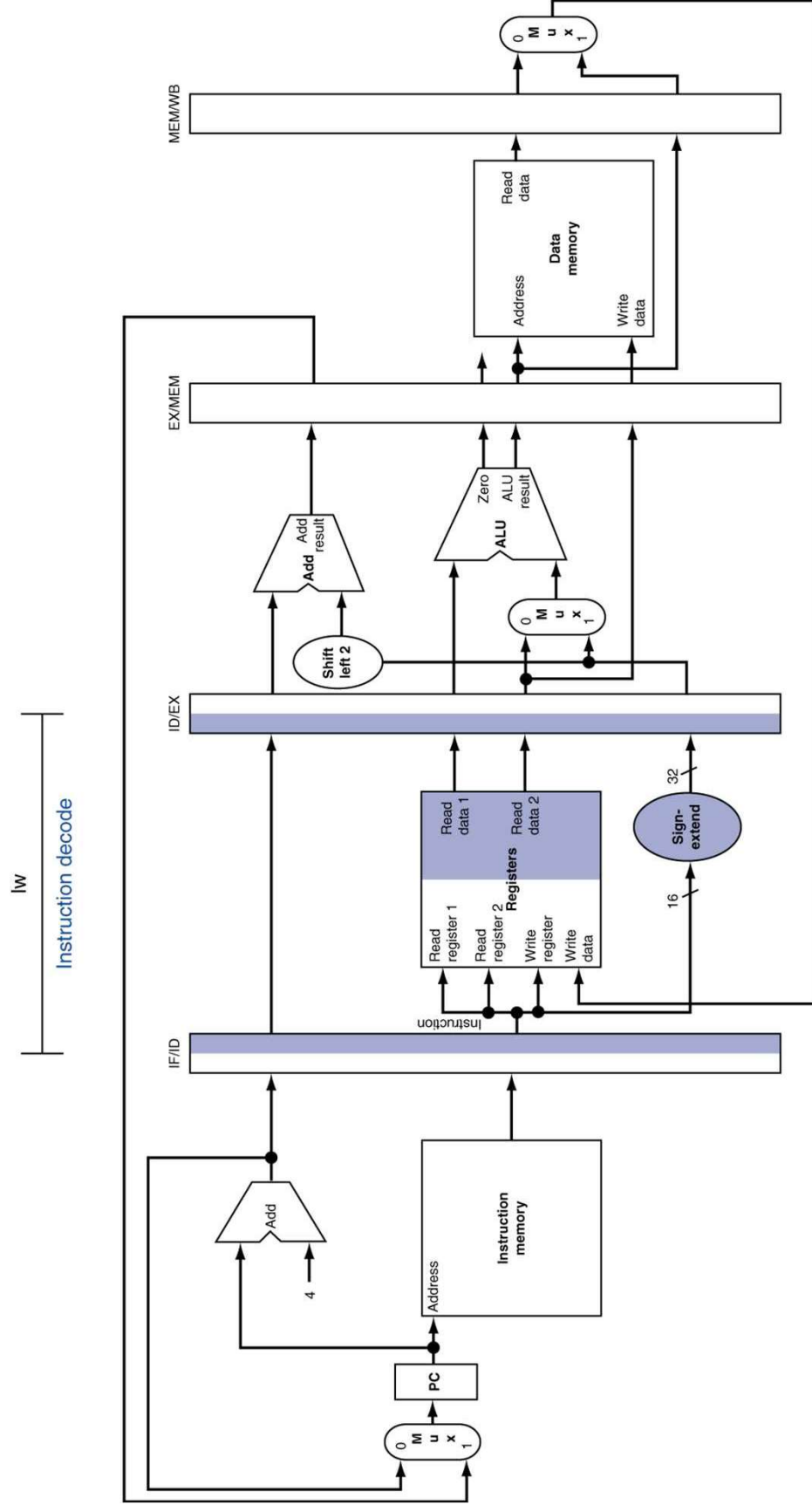
# Operação de Pipeline

- Fluxo de instruções ciclo por ciclo através do datapath com pipeline
  - Diagrama “Single-clock-cycle” do pipeline
    - Mostra o uso do pipeline em um ciclo
    - Realces dos recursos usados
  - obs. Diagrama “multi-clock-cycle”
    - Gráfico da operação ao longo do tempo
- Vamos ver o diagrama “single-clock-cycle” para o load & store

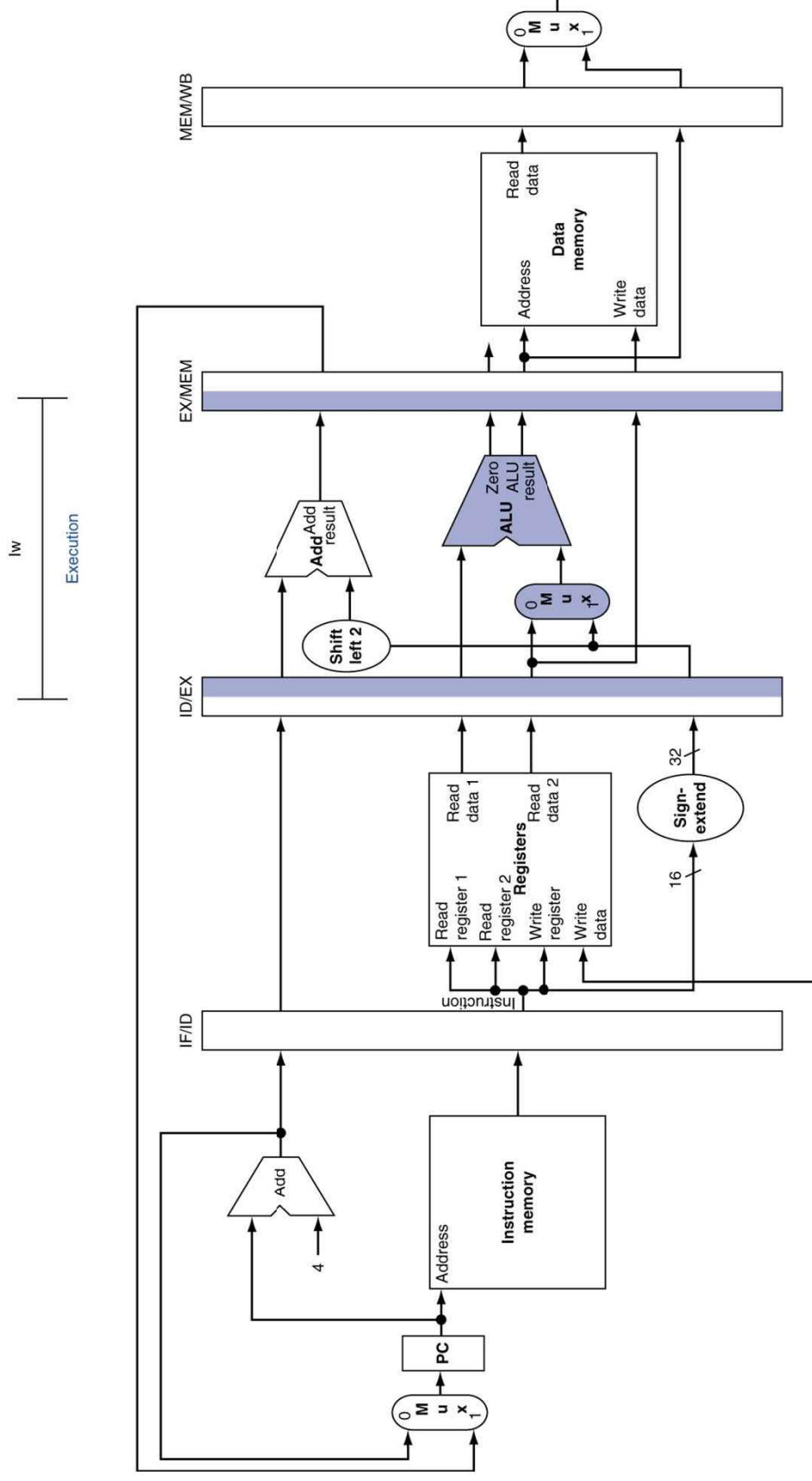
# IF para Load, Store, ...



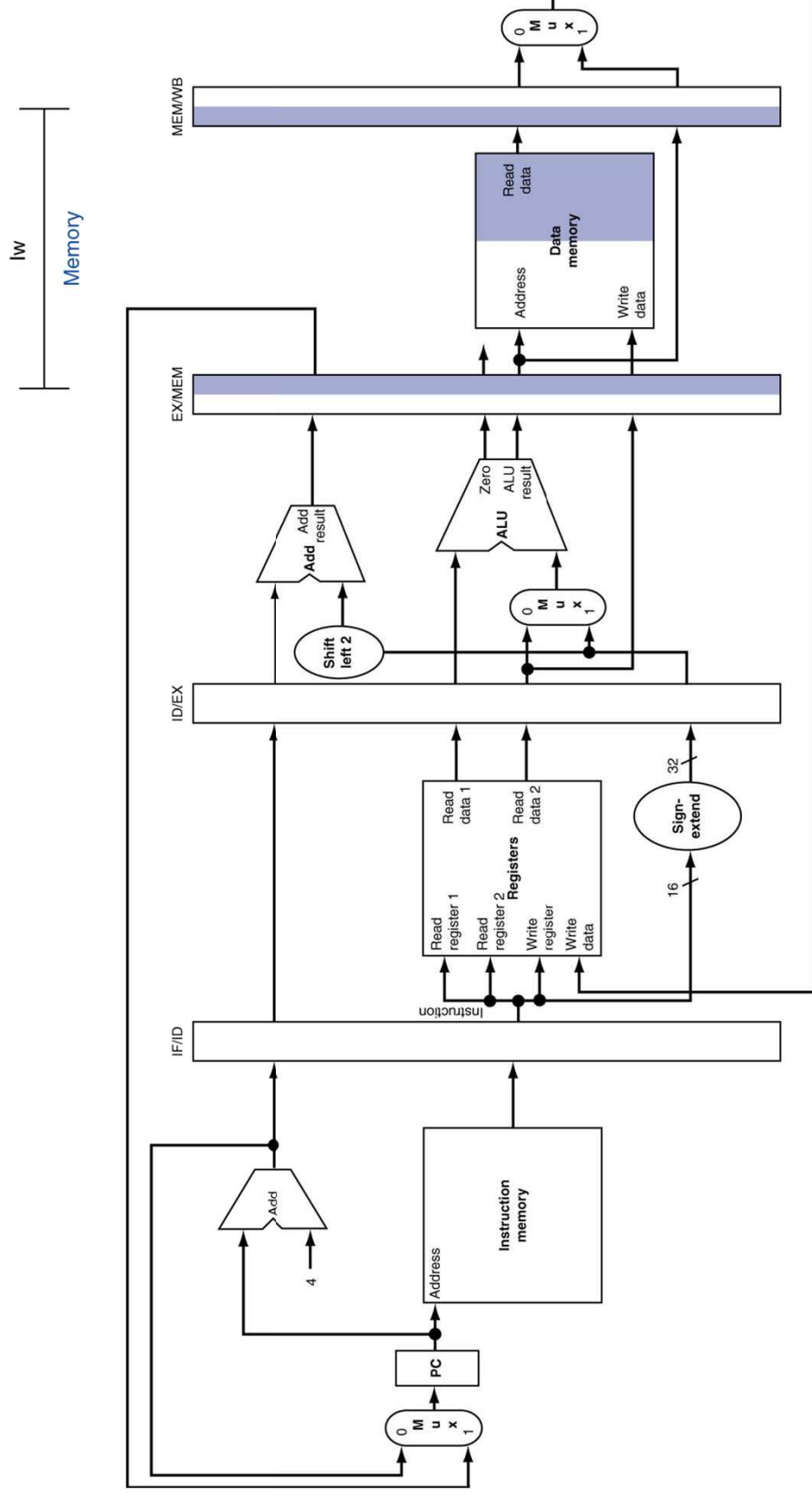
# ID para Load, Store, ...



# EX para Load

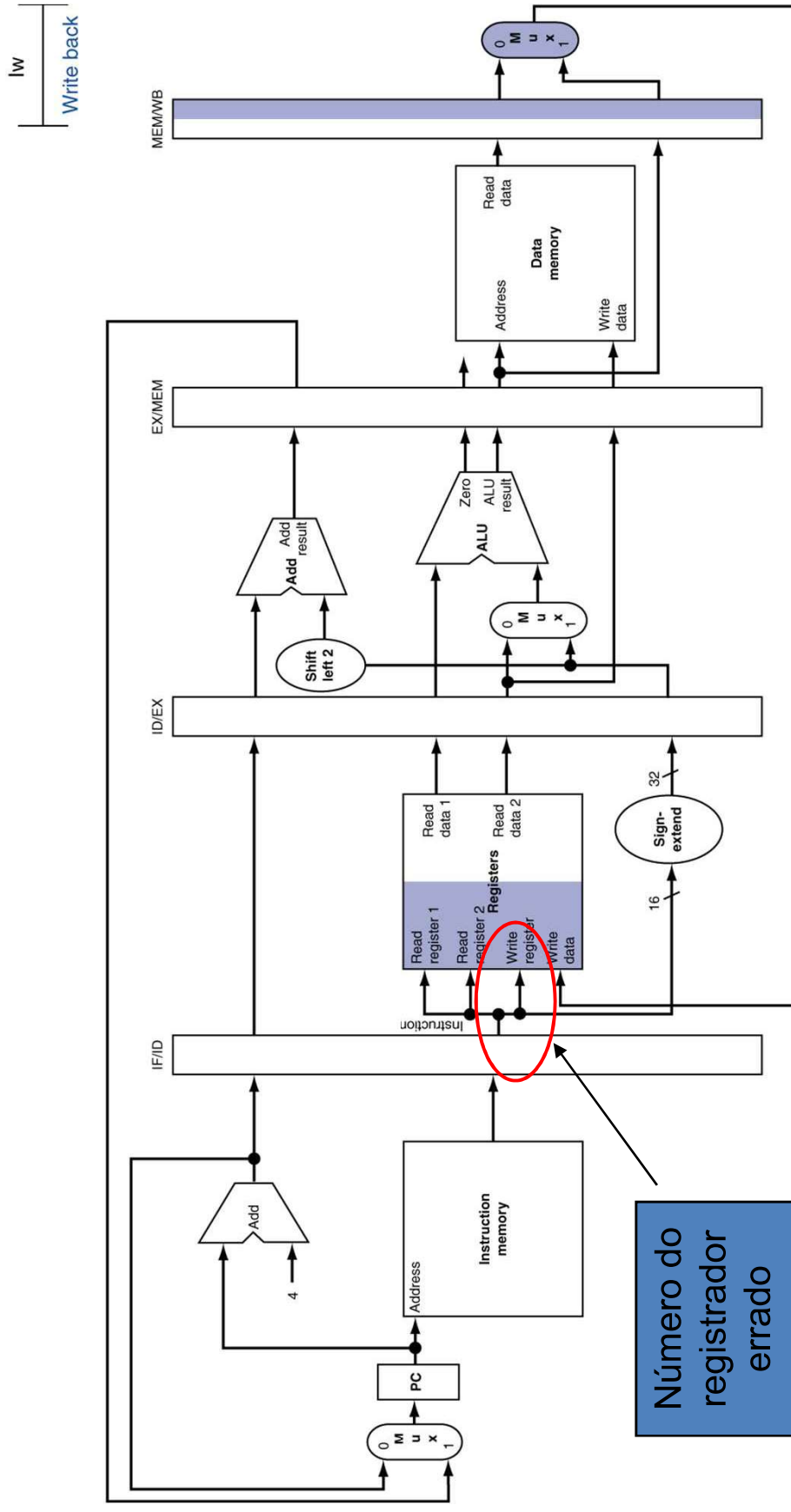


# MEM para Load

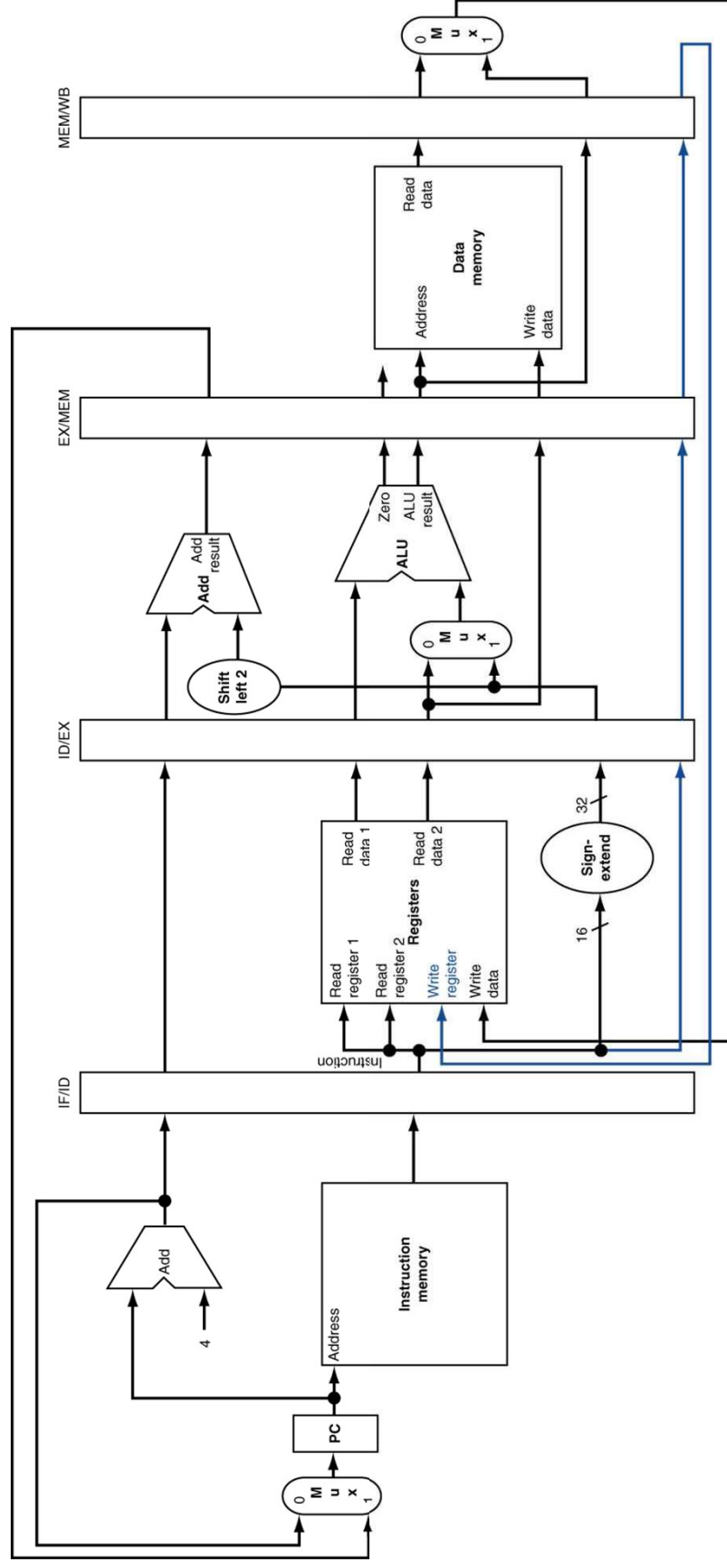




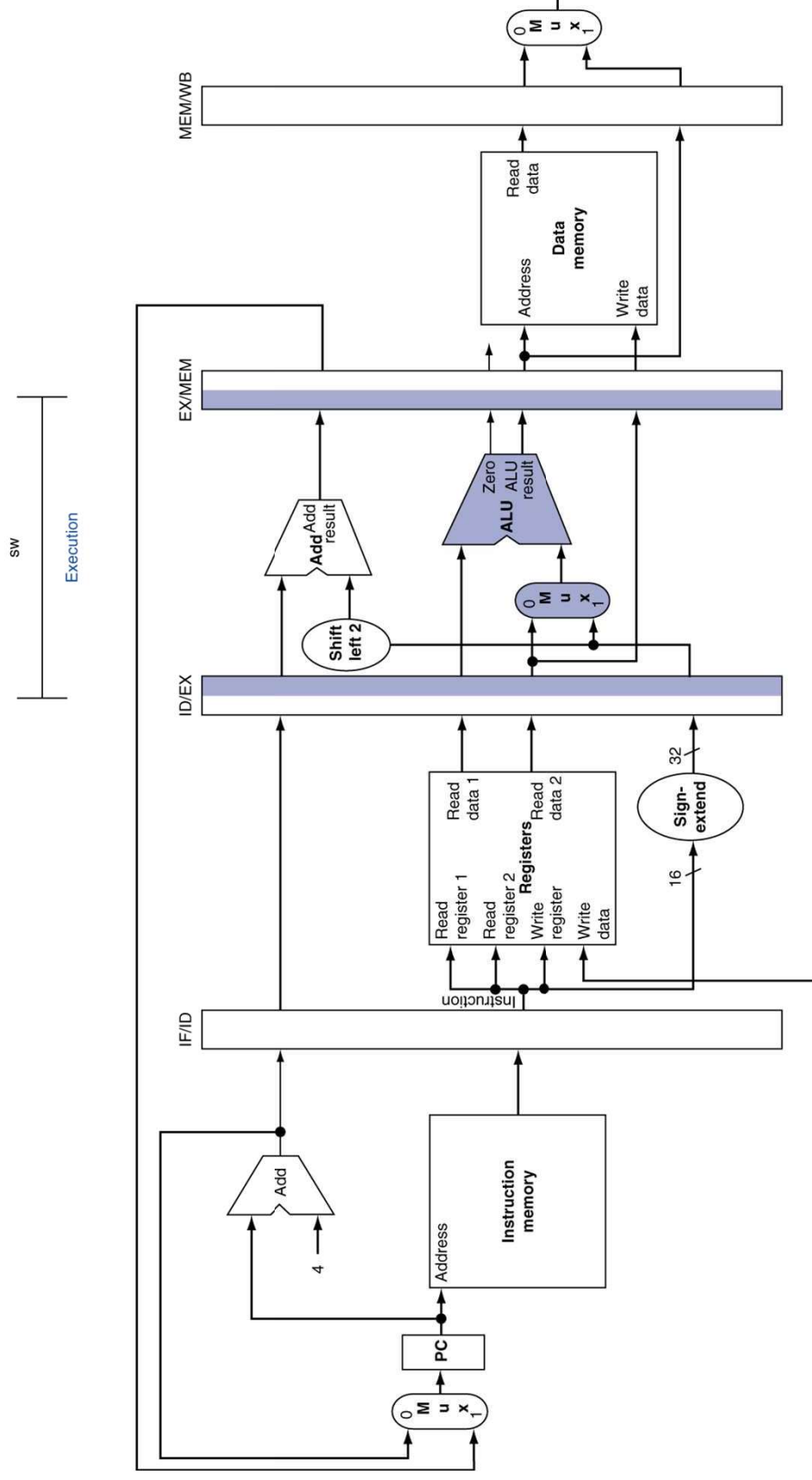
# WB para Load



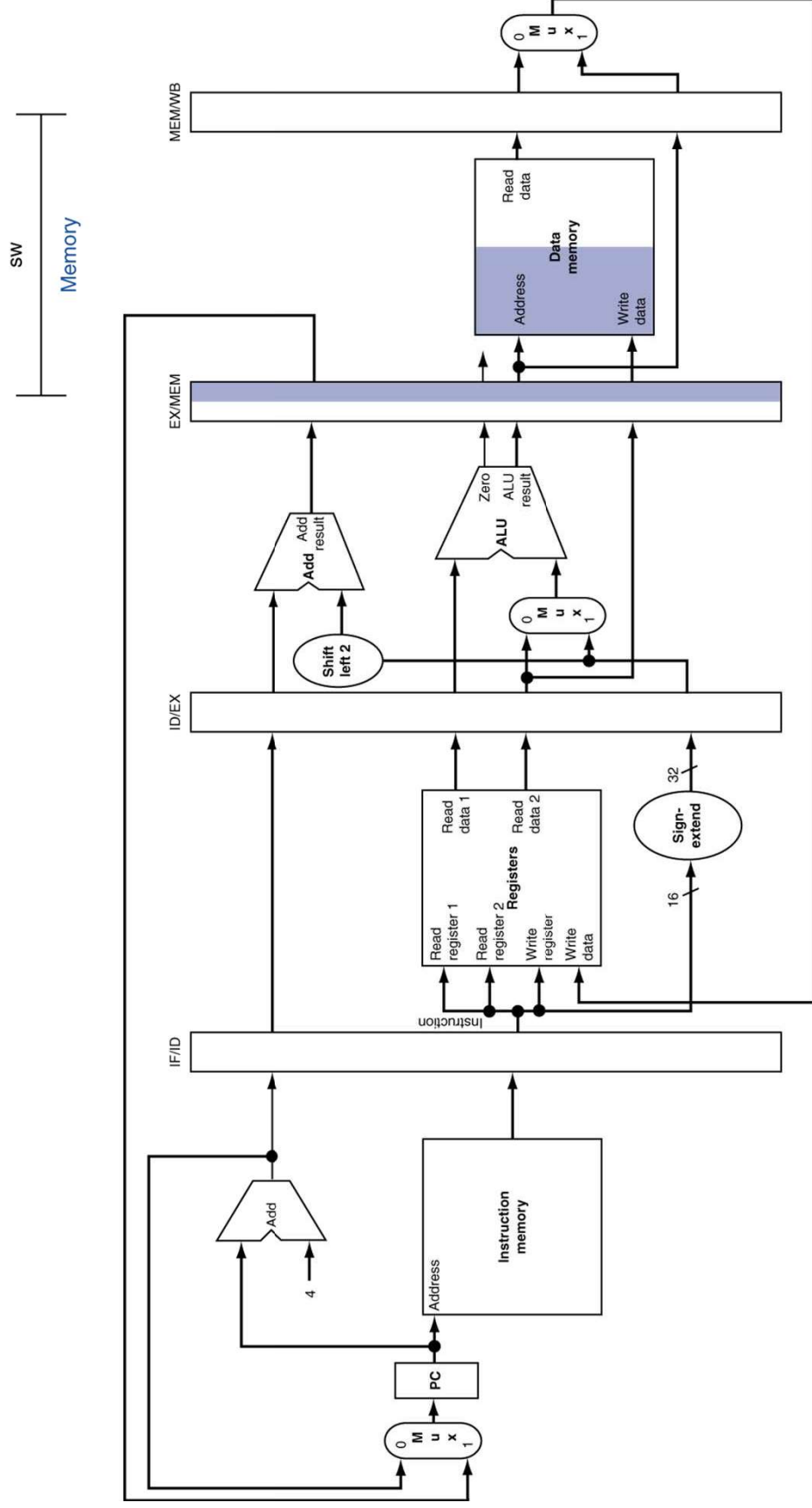
# Datapath para Load Corrigido



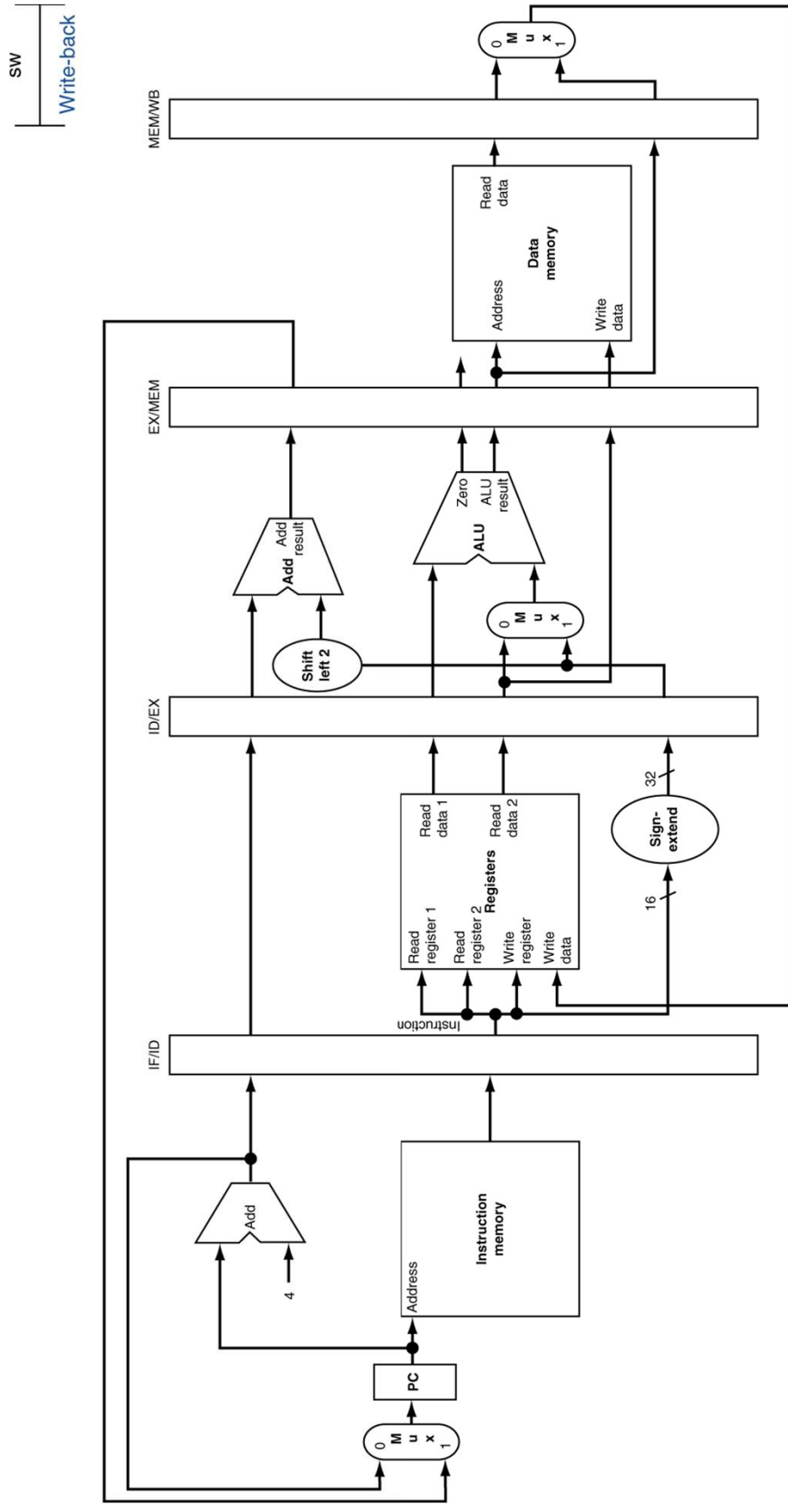
# EX para Store



# MEM para Store

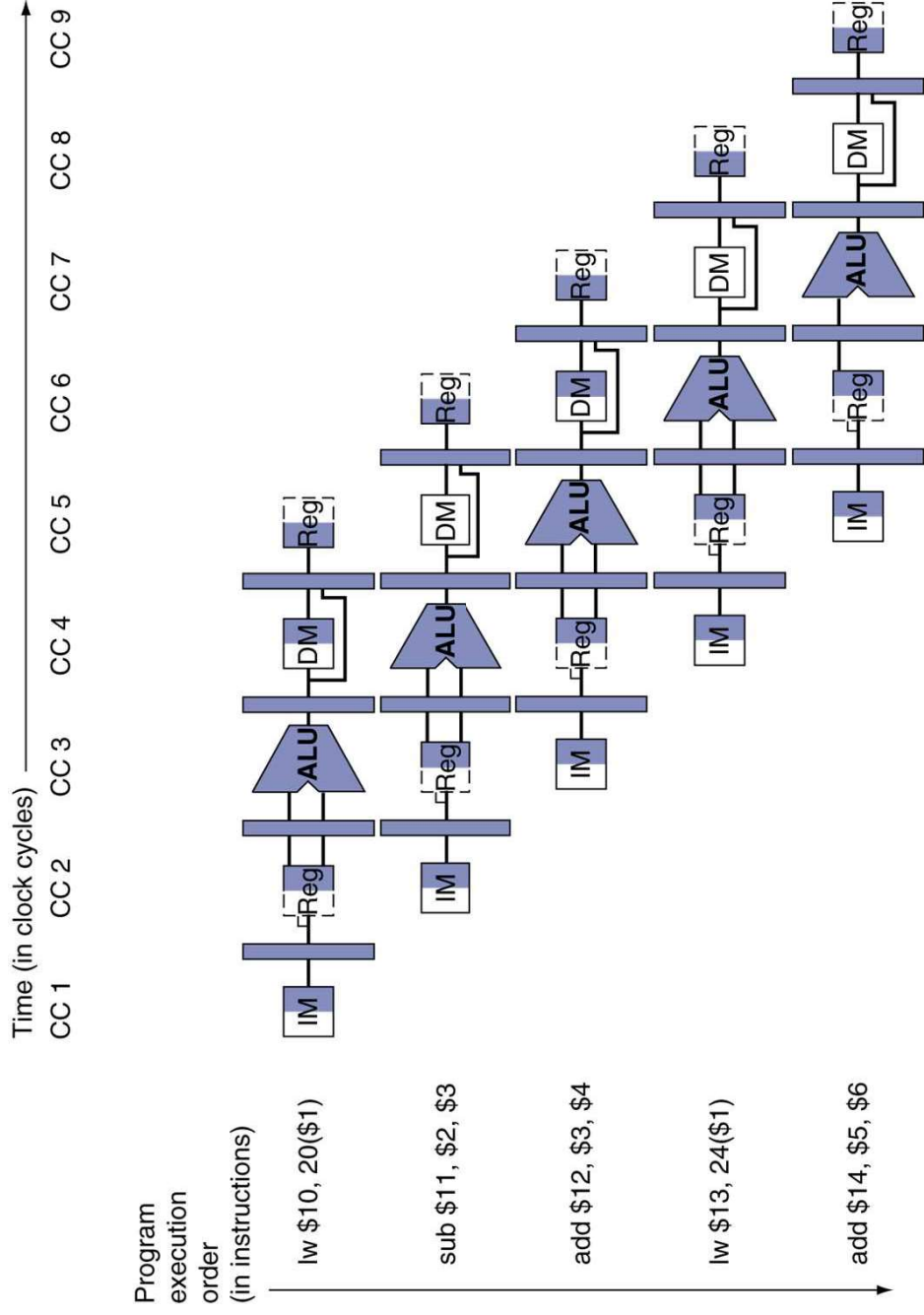


# WB para Store



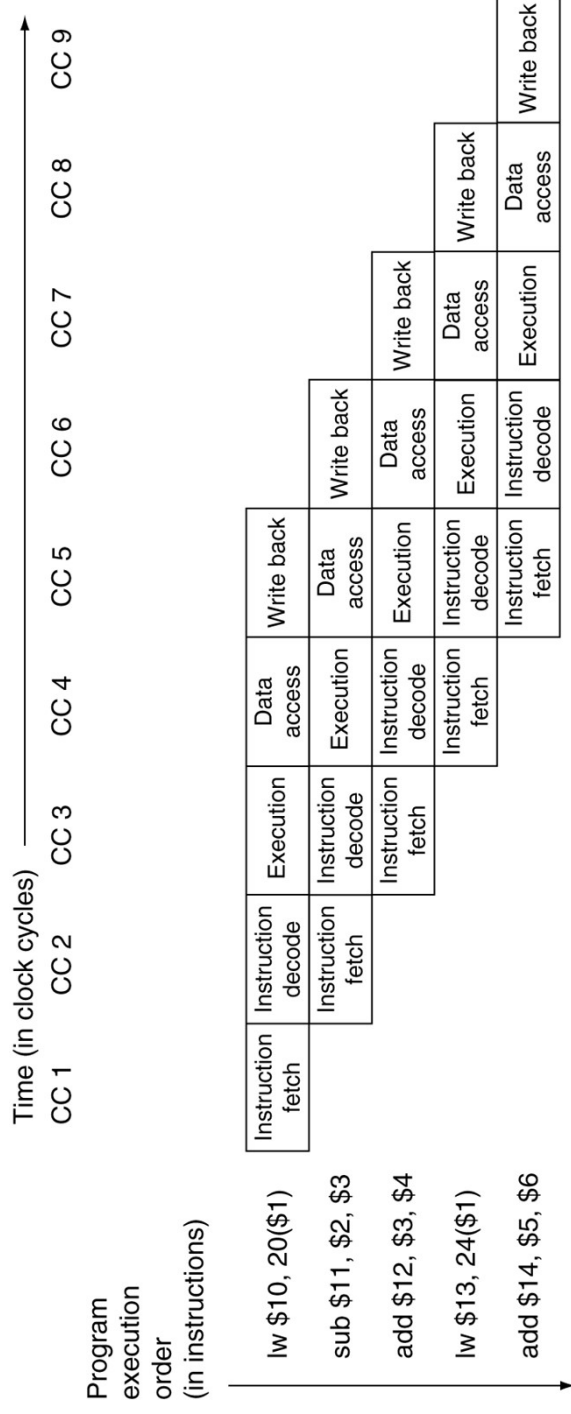
# Diagrama Multi-Ciclo do Pipeline

- Forma de mostrar o usos dos recursos

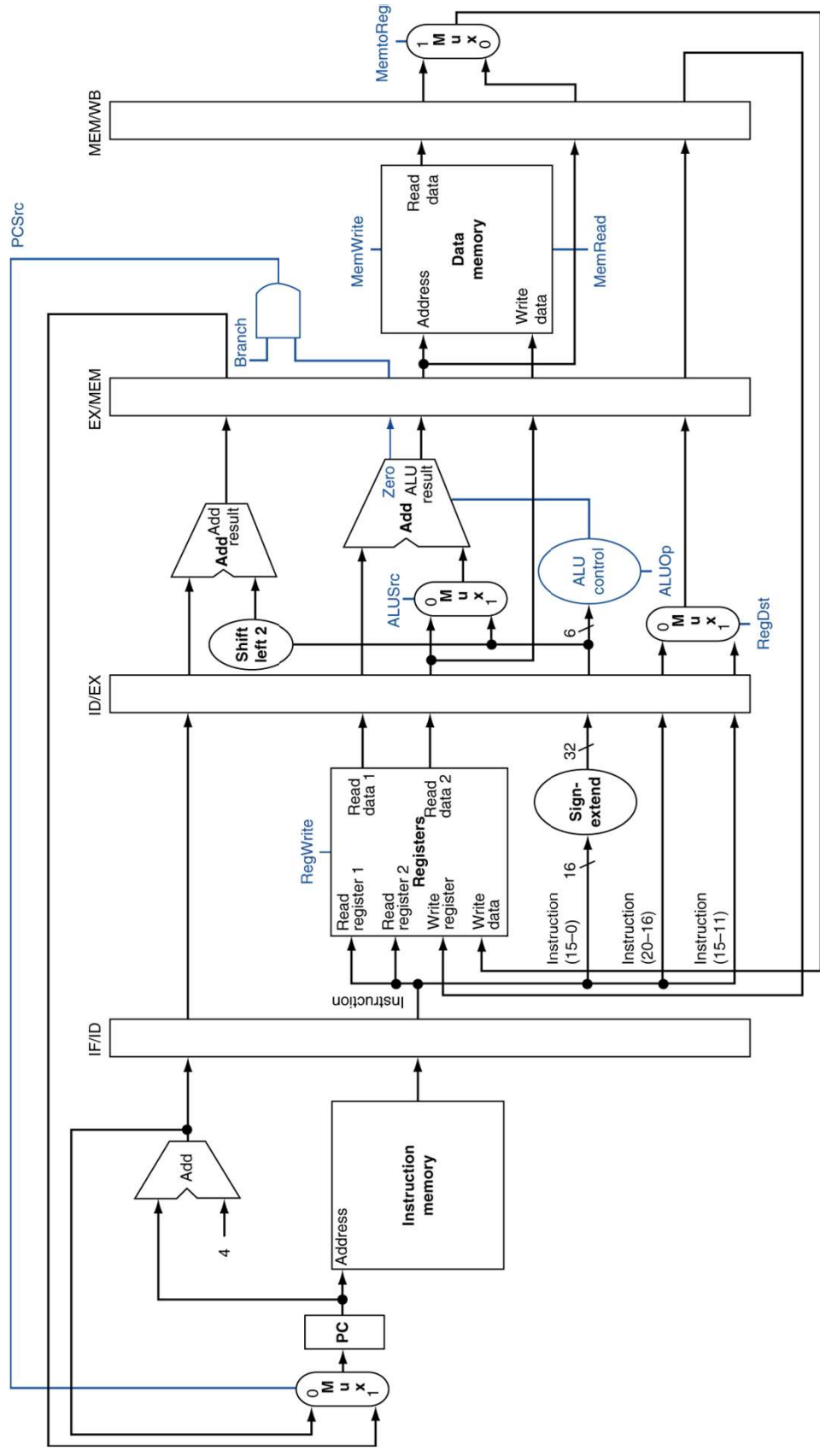


# Diagrama Multi-Ciclo do Pipeline

- Forma tradicional



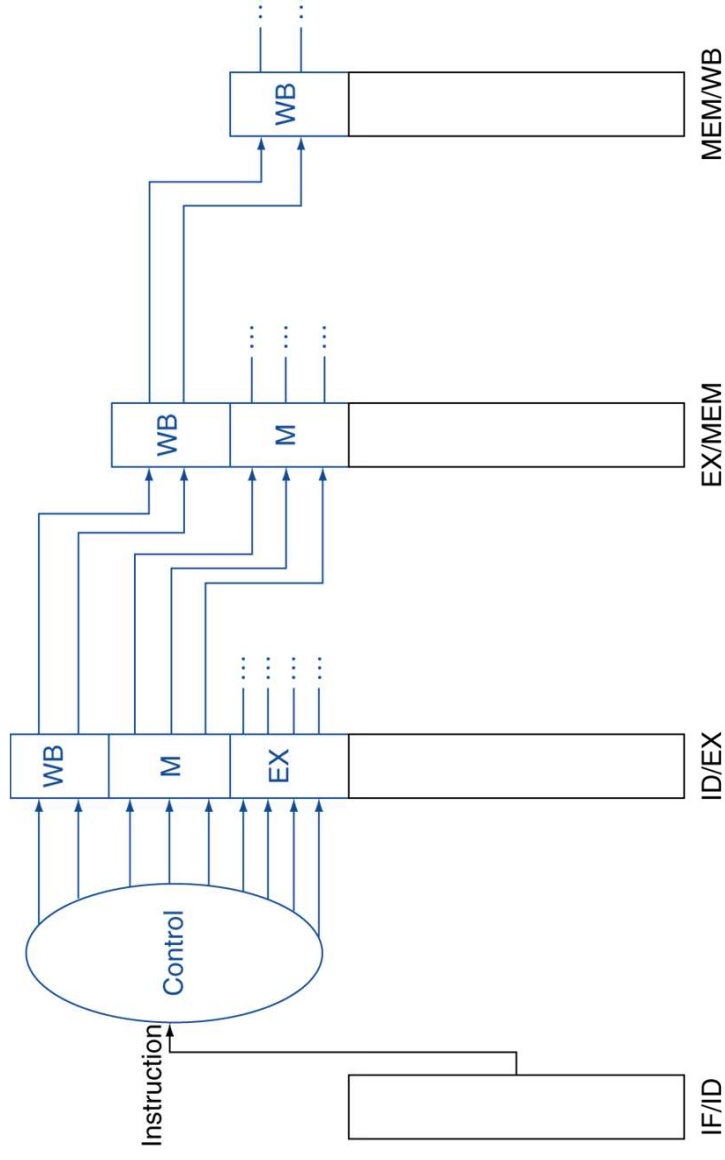
# Controle do Pipeline (Simplificado)





# Controle do Pipeline

- Sinais de control são derivados da instrução
  - Uma implementação no single-cycle



# Controle do Pipeline

