

Arquitetura e Organização de Computadores I

Prof. Mateus Felipe Tymburibá Ferreira
mateustymbu@cefetmg.br

Parte V: Pipelining

Caminho de Dados do MIPS com Pipeline

IF: Instruction fetch

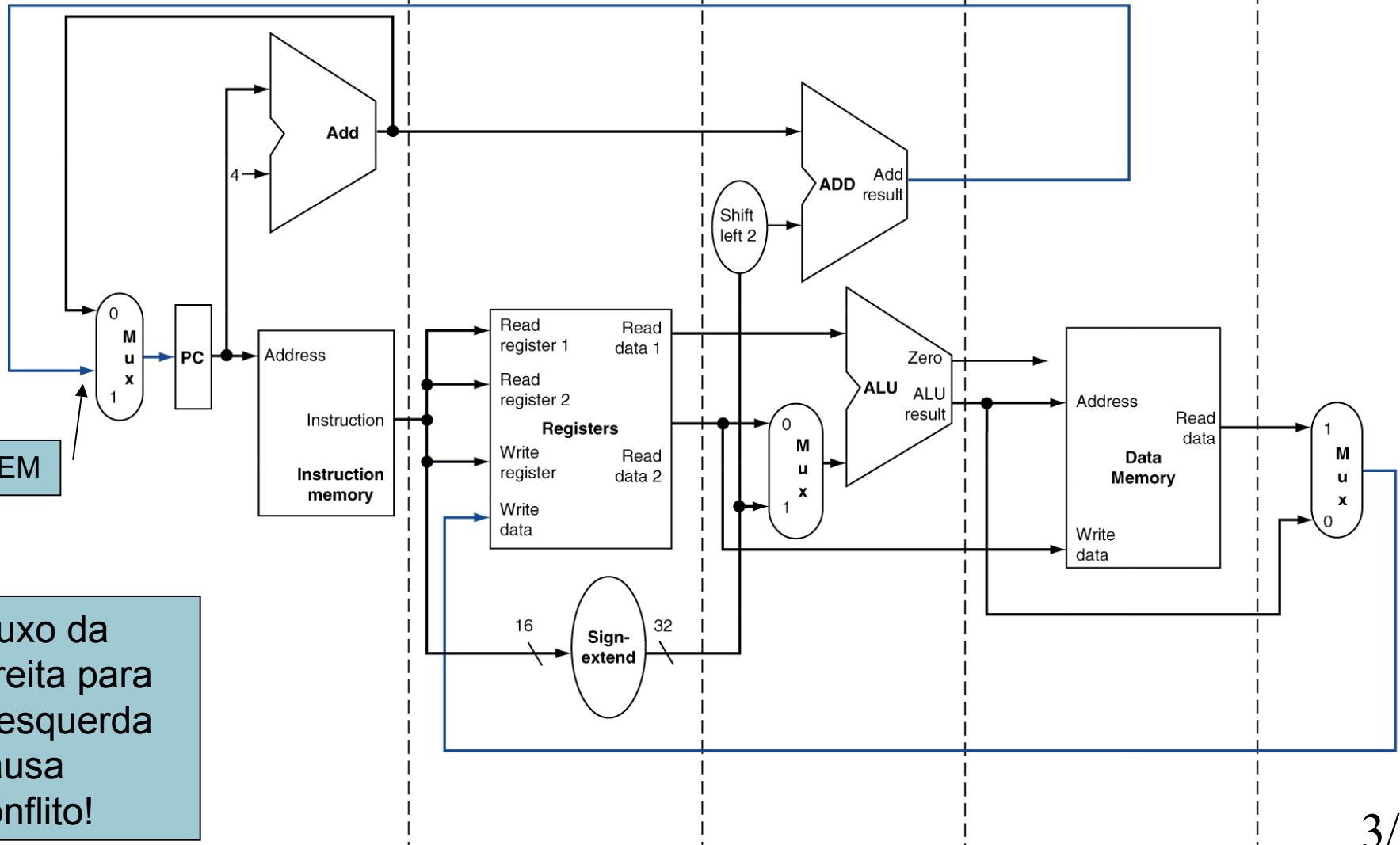
ID: Instruction decode/
register file read

EX: Execute/
address calculation

MEM: Memory access

WB: Write back

Qual o tipo de
conflito?



Caminho de Dados do MIPS com Pipeline

IF: Instruction fetch

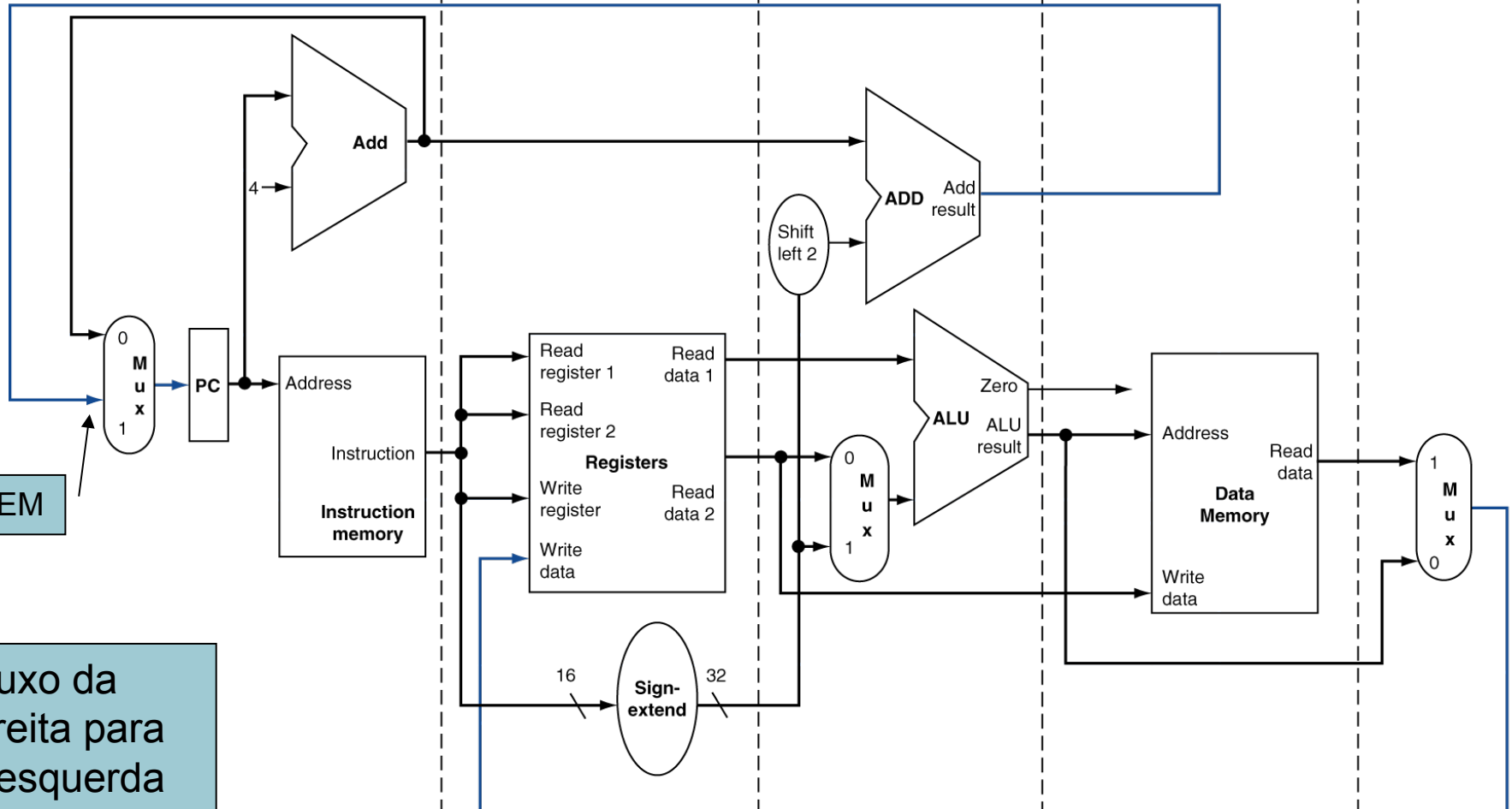
ID: Instruction decode/
register file read

EX: Execute/
address calculation

MEM: Memory access

WB: Write back

Qual o tipo deste
conflito? **De controle**



Fluxo da
direita para
a esquerda
causa
conflito!

Caminho de Dados do MIPS com Pipeline

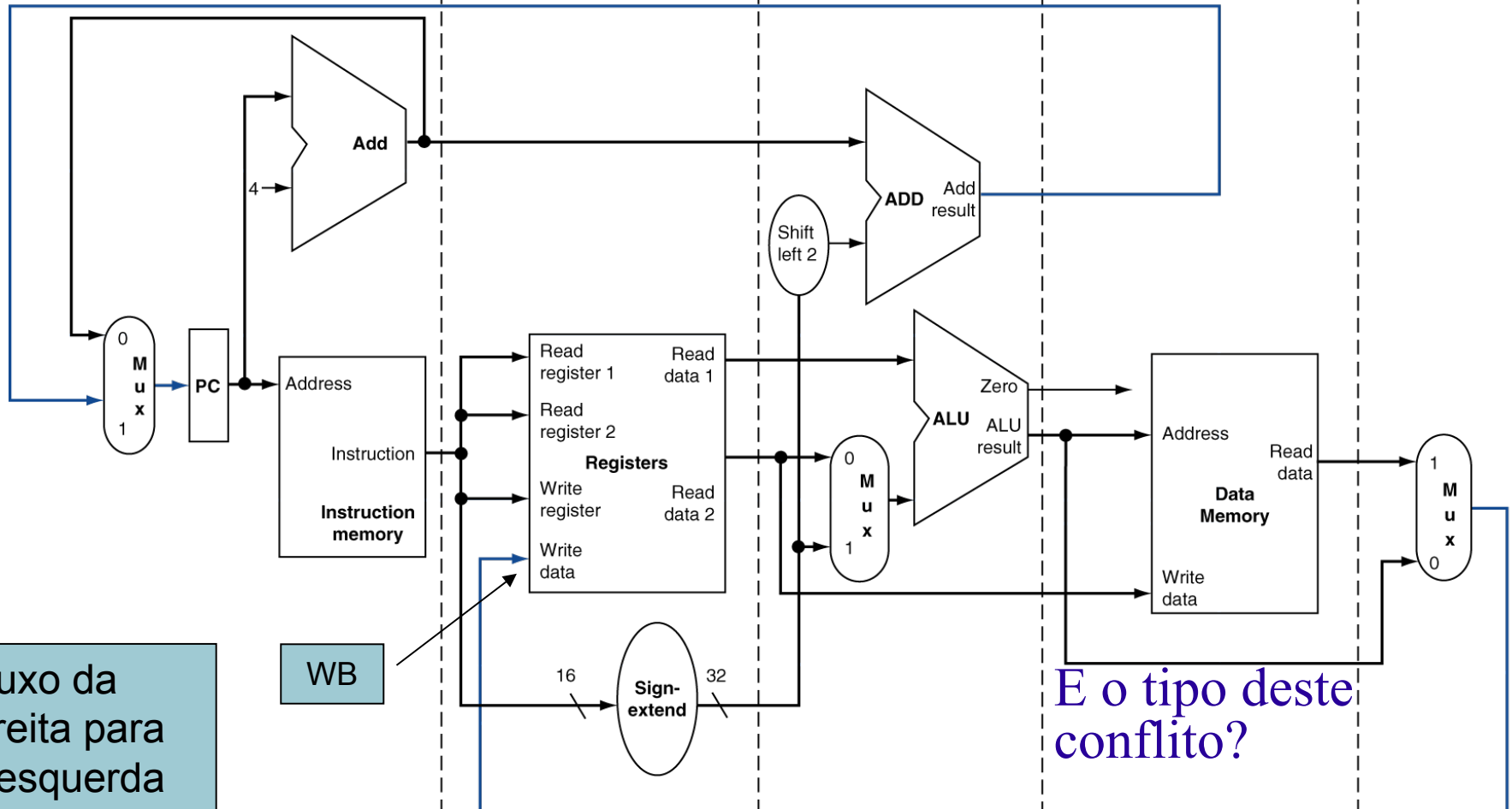
IF: Instruction fetch

ID: Instruction decode/
register file read

EX: Execute/
address calculation

MEM: Memory access

WB: Write back



Fluxo da
direita para
a esquerda
causa
conflito!

E o tipo deste
conflito?

Caminho de Dados do MIPS com Pipeline

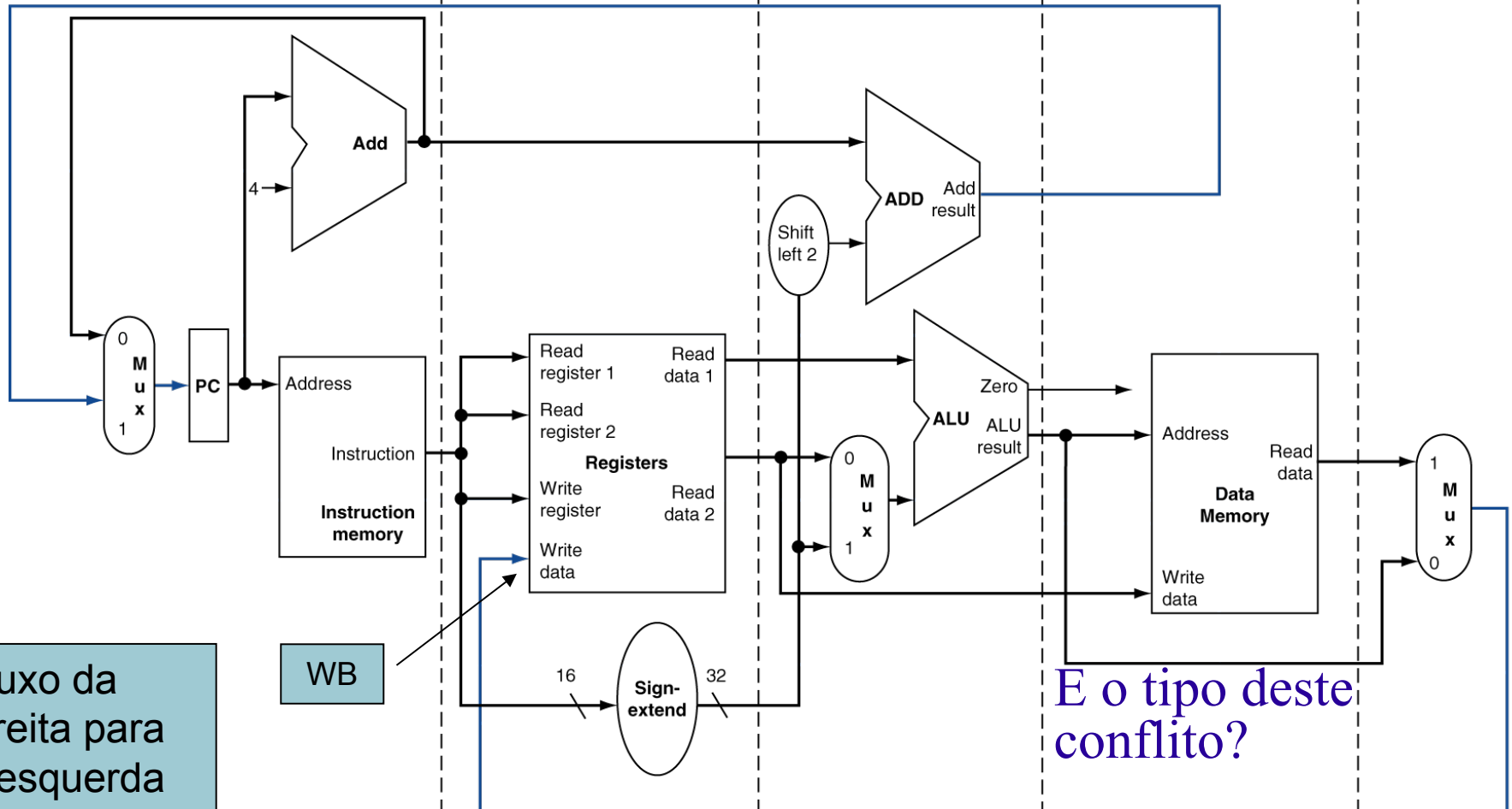
IF: Instruction fetch

ID: Instruction decode/
register file read

EX: Execute/
address calculation

MEM: Memory access

WB: Write back



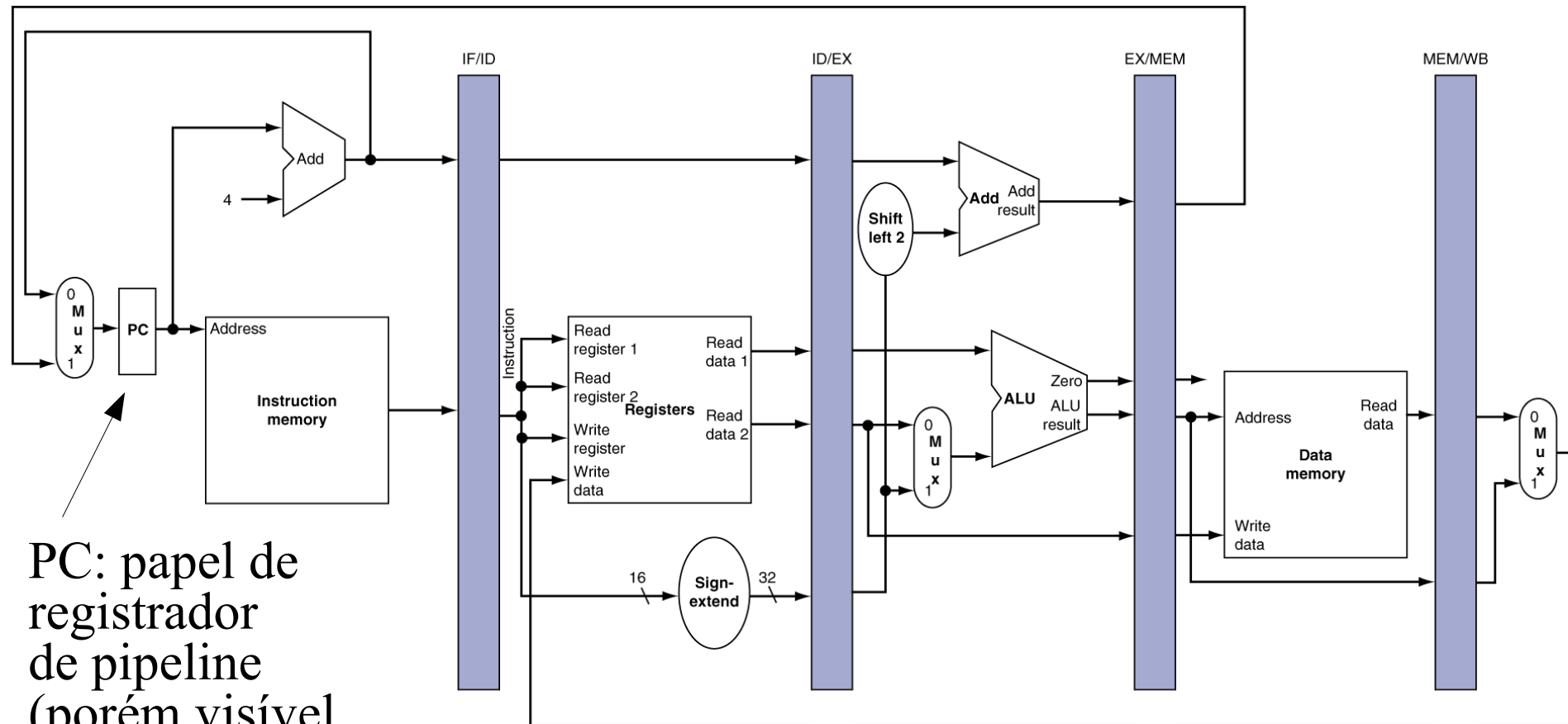
Fluxo da
direita para
a esquerda
causa
conflito!

E o tipo deste
conflito?

De dados.

Registradores de Pipeline (pipes ou tubos)

Necessidade de registradores entre estágios → guardar dados e sinais de controle produzidos em ciclos anteriores

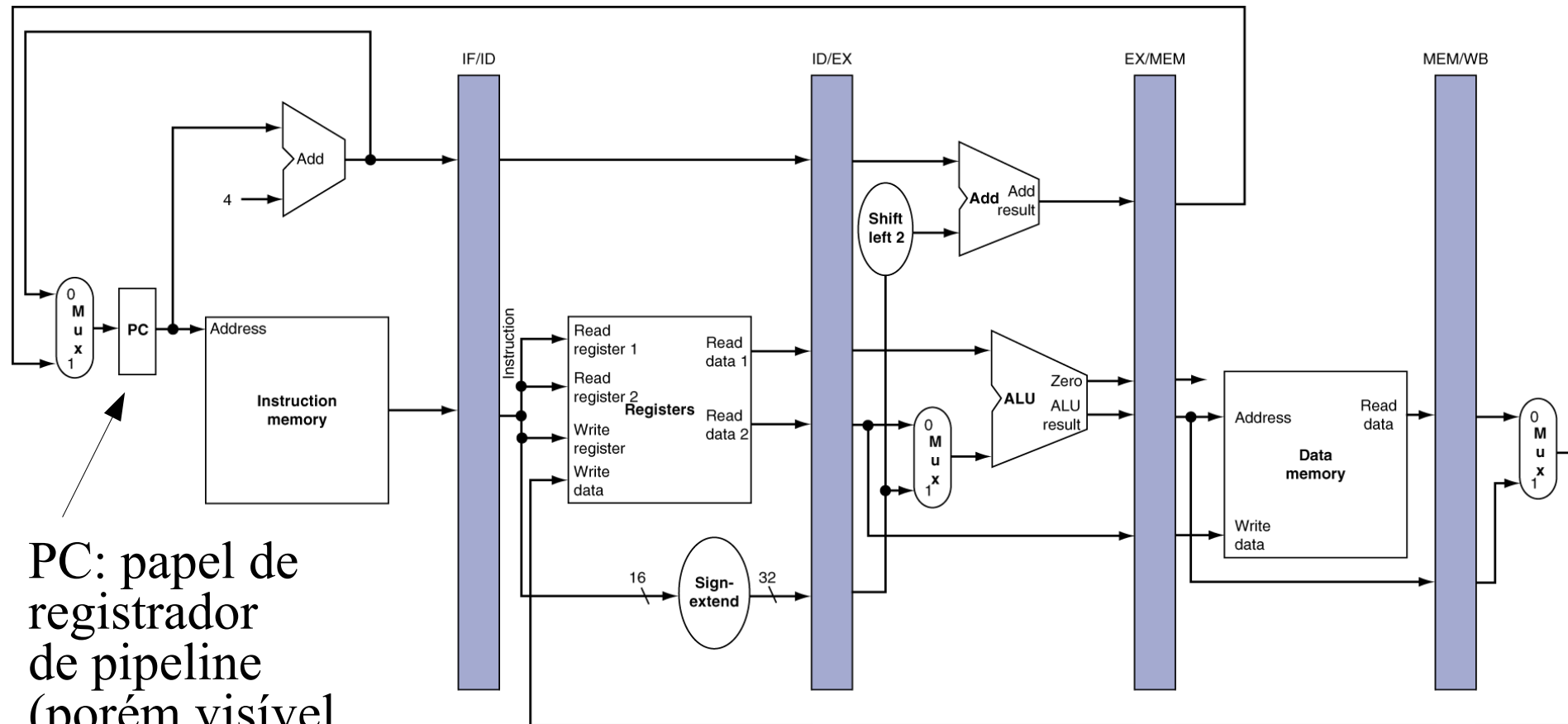


PC: papel de
registrador
de pipeline
(porém visível
ao programador)

Por essa imagem, qual o tamanho dos registradores de pipeline?

Registradores de Pipeline (pipes ou tubos)

Necessidade de registradores entre estágios → guardar dados e sinais de controle produzidos em ciclos anteriores



PC: papel de registrador de pipeline (porém visível ao programador)

Por essa imagem, qual o tamanho dos registradores de pipeline? IF/ID: 64 bits, ID/EX: 128 bits, EX/MEM: 97 bits, MEM/WB: 64 bits.

Operação do Pipeline

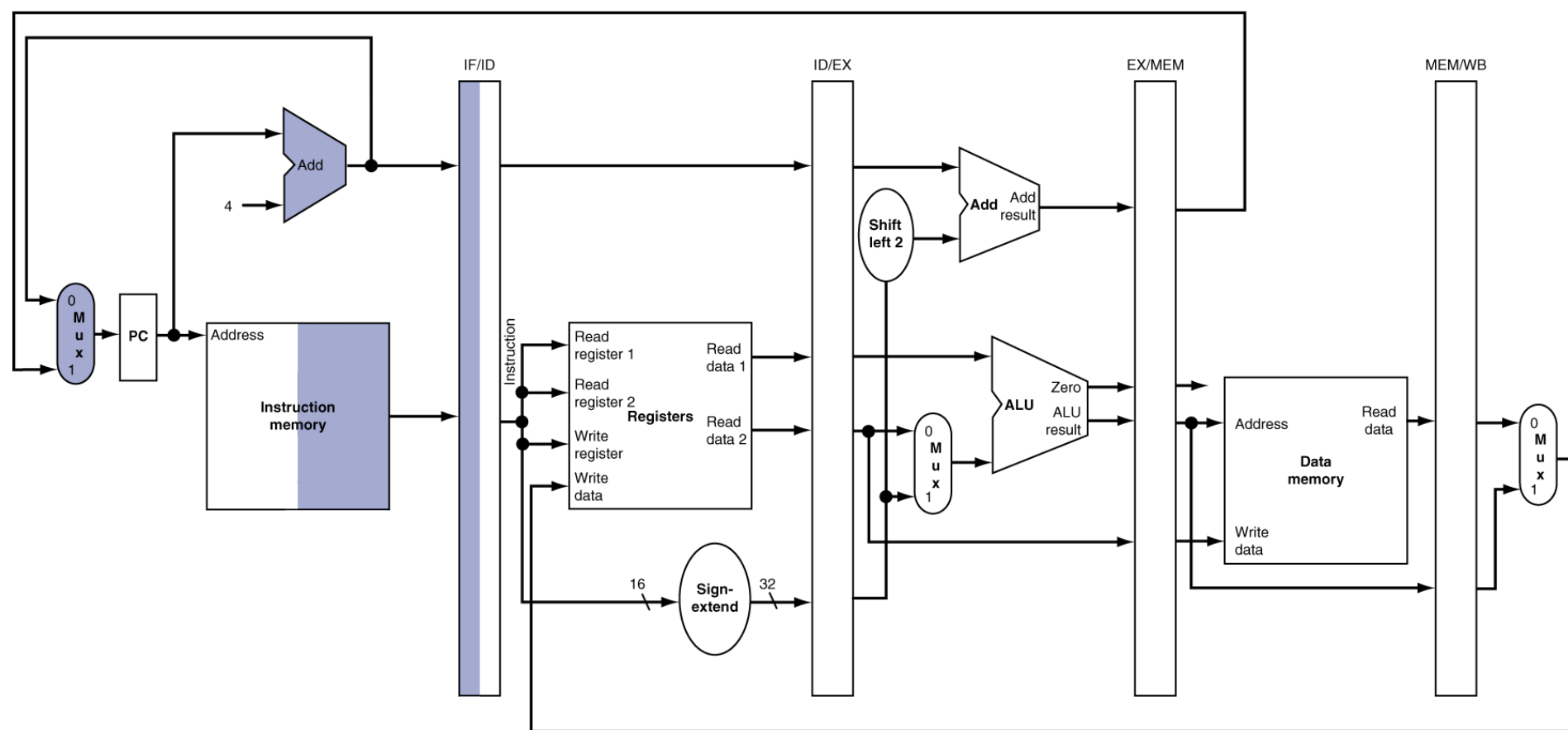
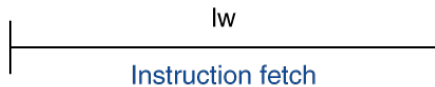
Fluxo de instruções, ciclo-a-ciclo, pelo caminho de dados com pipeline:

- Diagrama de pipeline para “Ciclo de clock único”
 - Mostra o uso do pipeline em um único ciclo de clock
 - Destaca os recursos usados
- Diagrama de pipeline para “Múltiplos ciclos de clock”
 - Ilustra operação ao longo do tempo

Vamos olhar os diagramas de “Ciclo de clock único” para *loads* e *stores*!

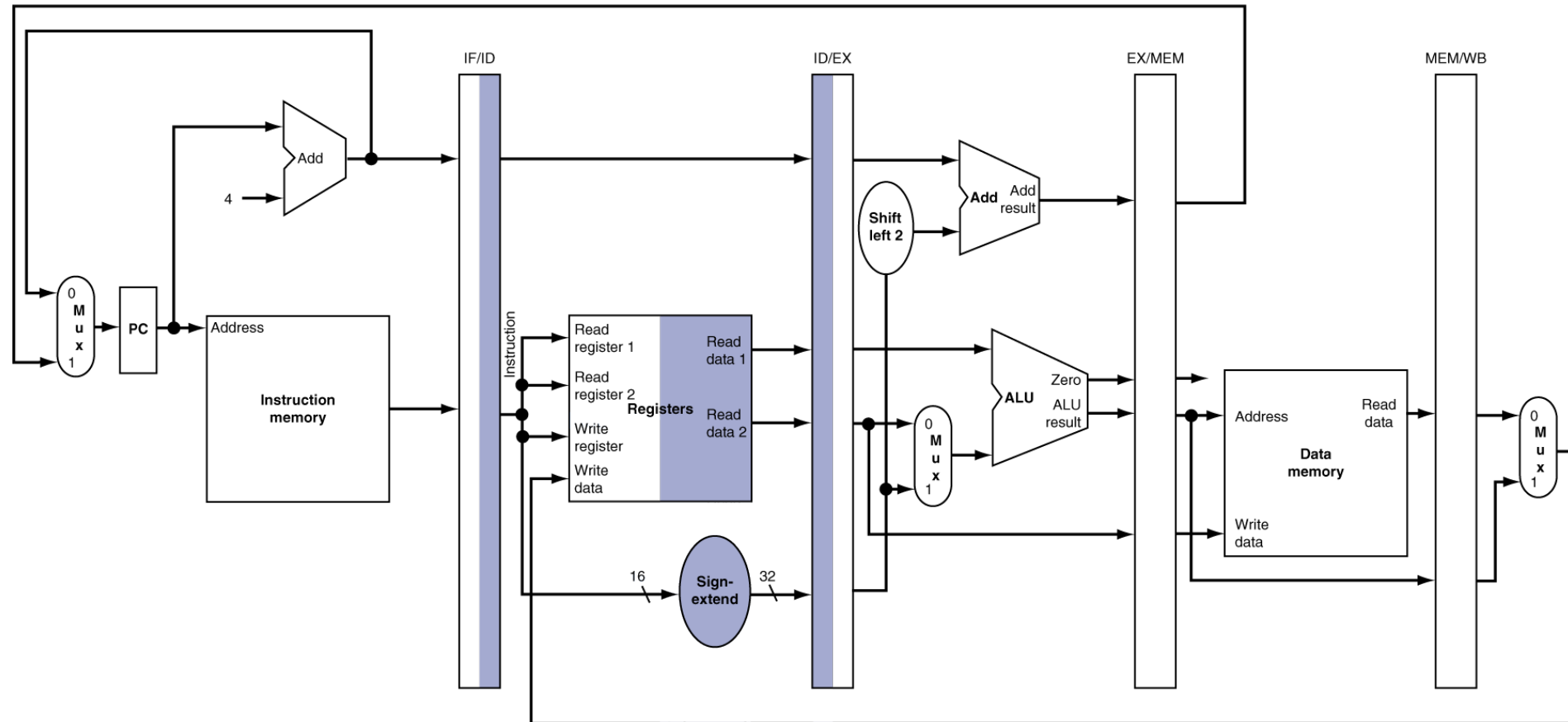
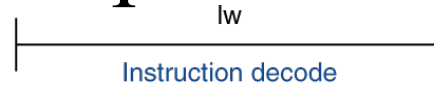
IF para Load e Store

- Leitura: metade direita destacada
- Escrita: metade esquerda destacada



Não se sabe qual instrução foi buscada → Passa-se informações potencialmente necessárias pelo pipeline. (Ex: $PC + 4 \rightarrow beq$)

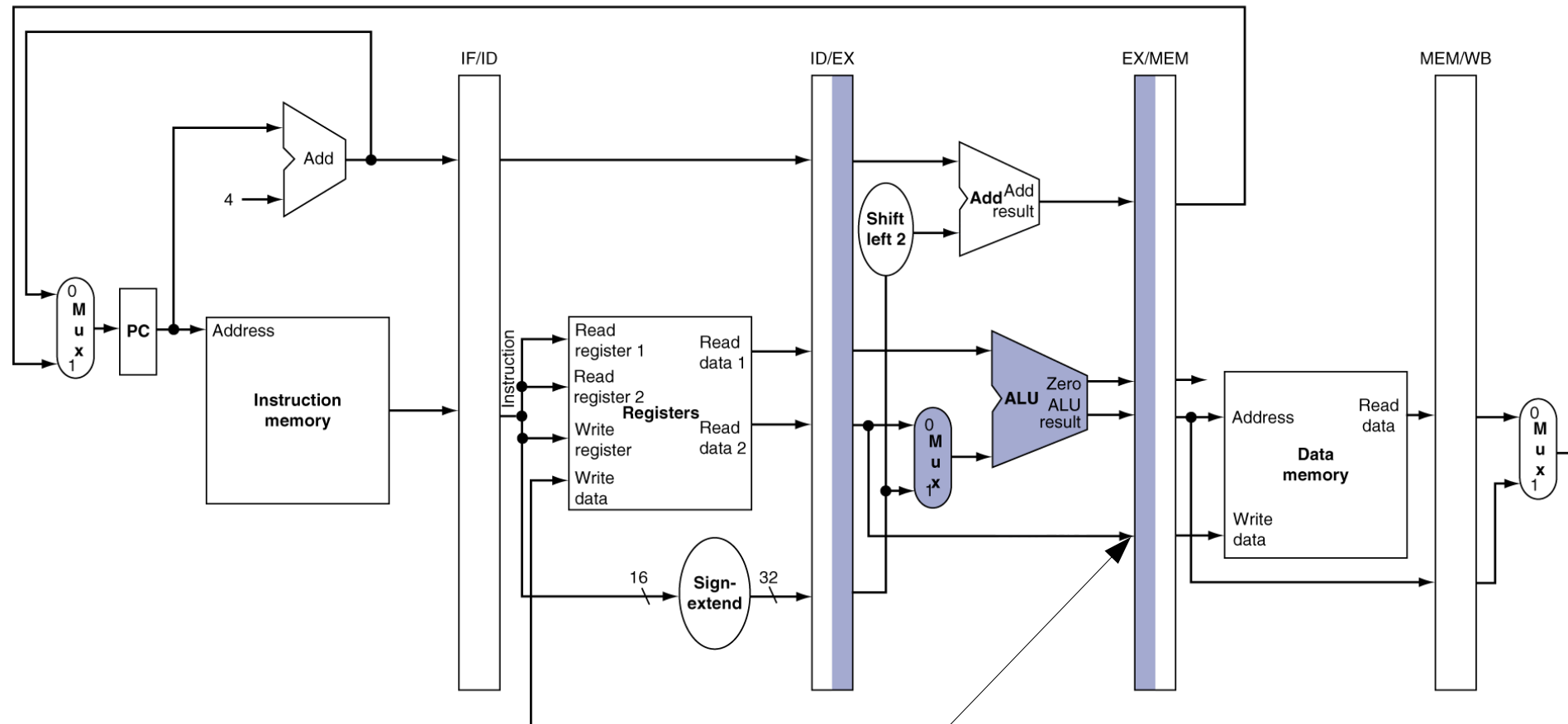
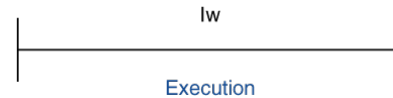
ID para Load e Store



Novamente, encaminha-se todos os dados que possam ser necessários em estágios posteriores da instrução.

Como poderíamos reduzir o tamanho do registrador ID/EX?
Quais as vantagens e desvantagens dessa mudança?

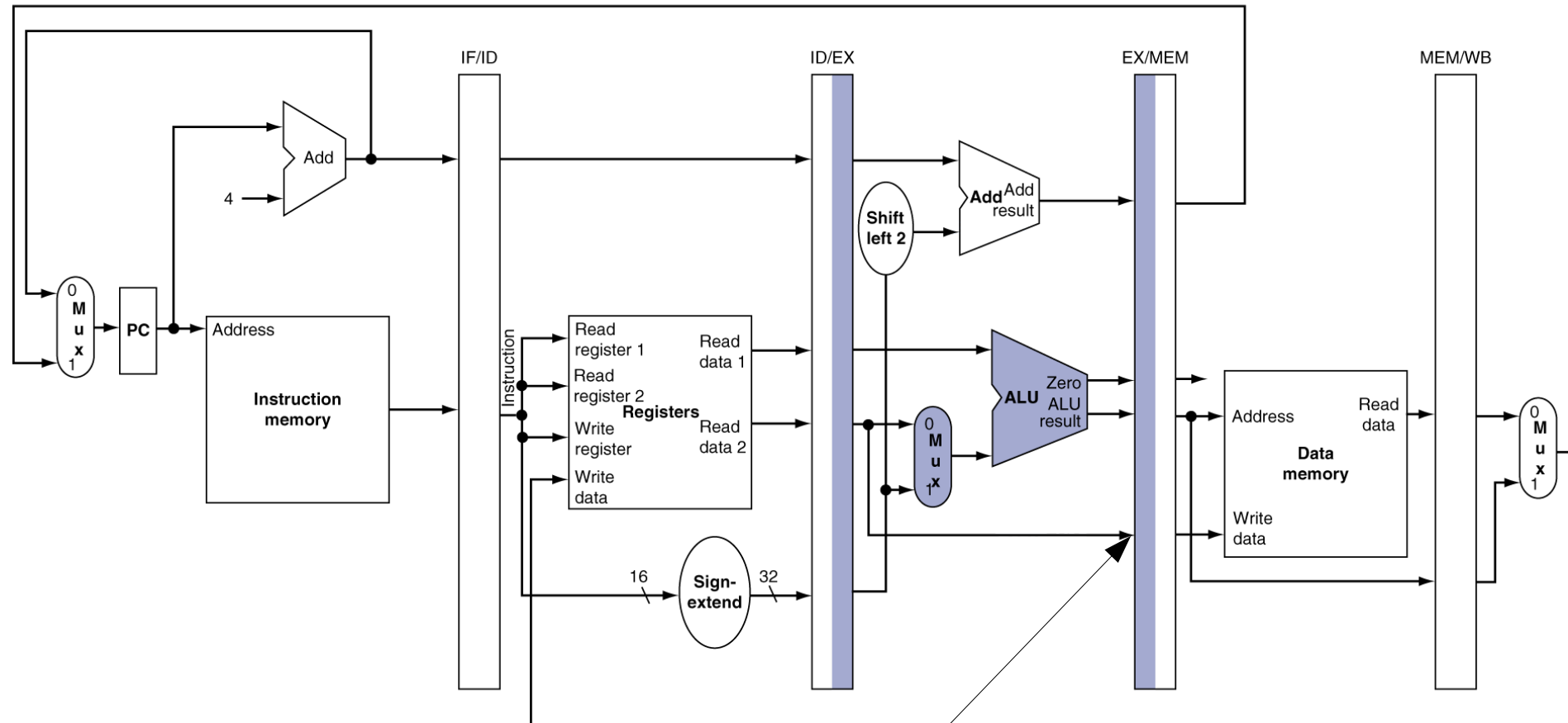
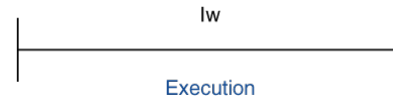
EX para Load



Novamente, encaminha-se todos os dados que possam ser necessários em estágios posteriores da instrução.

Qual a utilidade deste encaminhamento (dica: não é para o Load)?

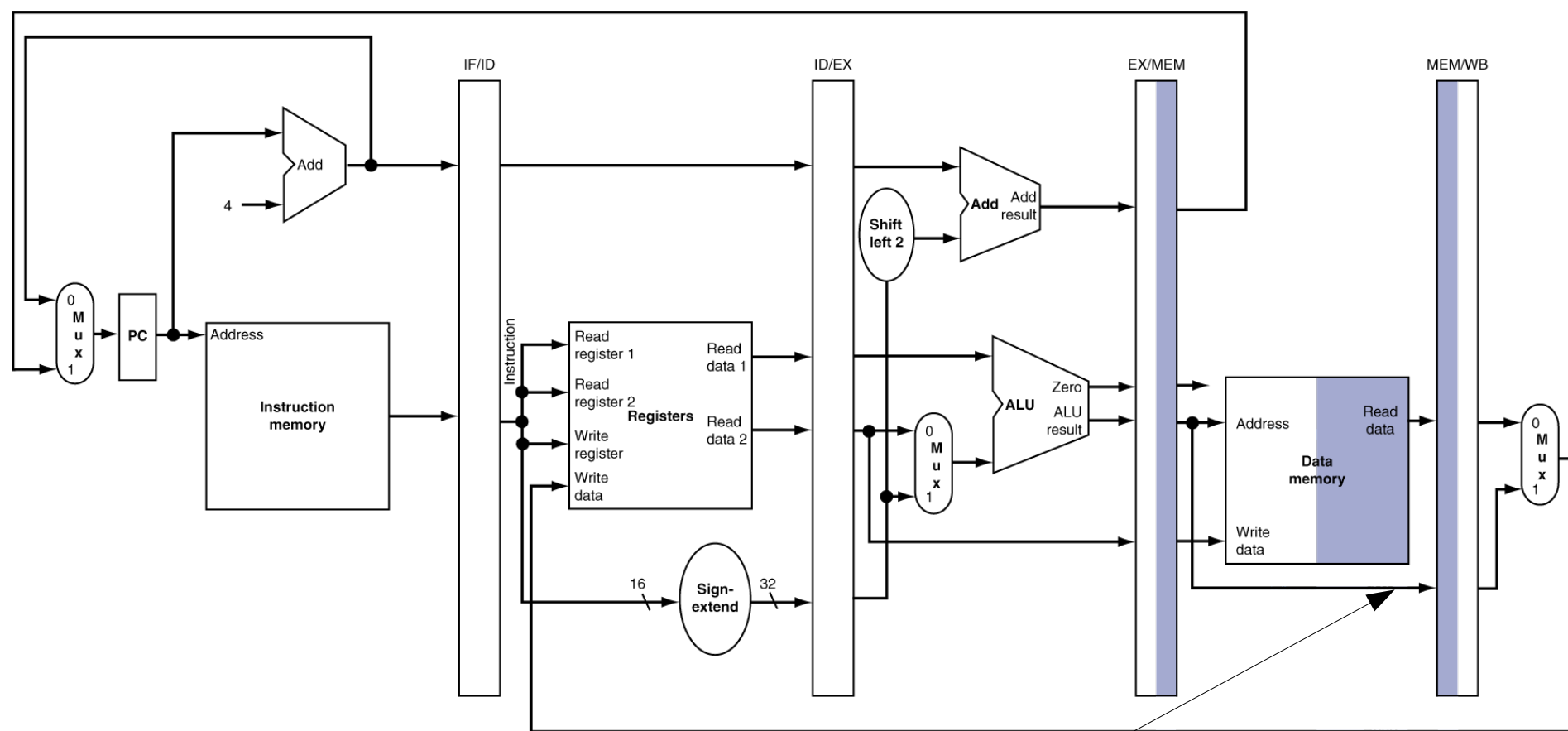
EX para Load



Novamente, encaminha-se todos os dados que possam ser necessários em estágios posteriores da instrução.

Qual a utilidade deste encaminhamento (dica: não é para o Load)? Store.

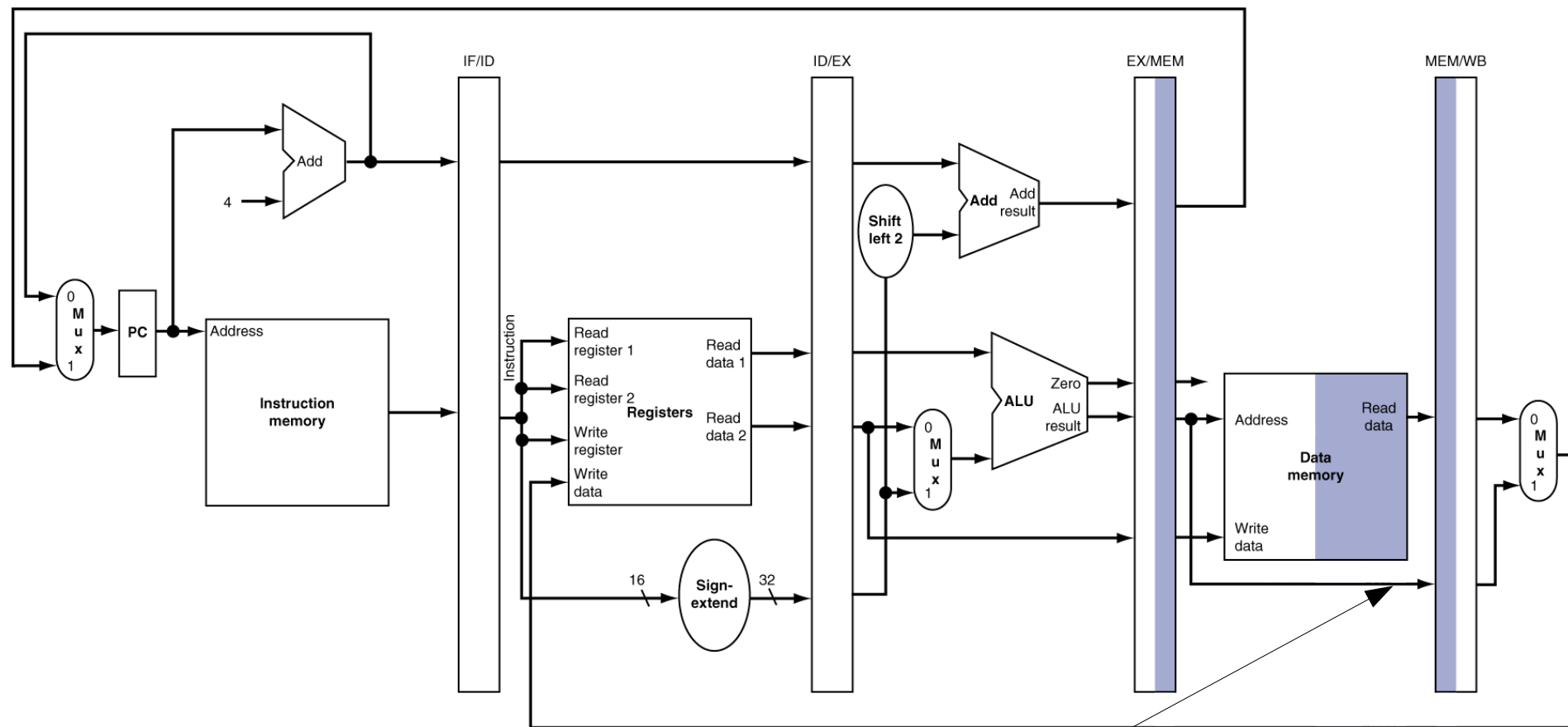
MEM para Load



Novamente, encaminha-se todos os dados que possam ser necessários em estágios posteriores da instrução.

Qual a utilidade deste encaminhamento (dica: não é para o Load)?

MEM para Load

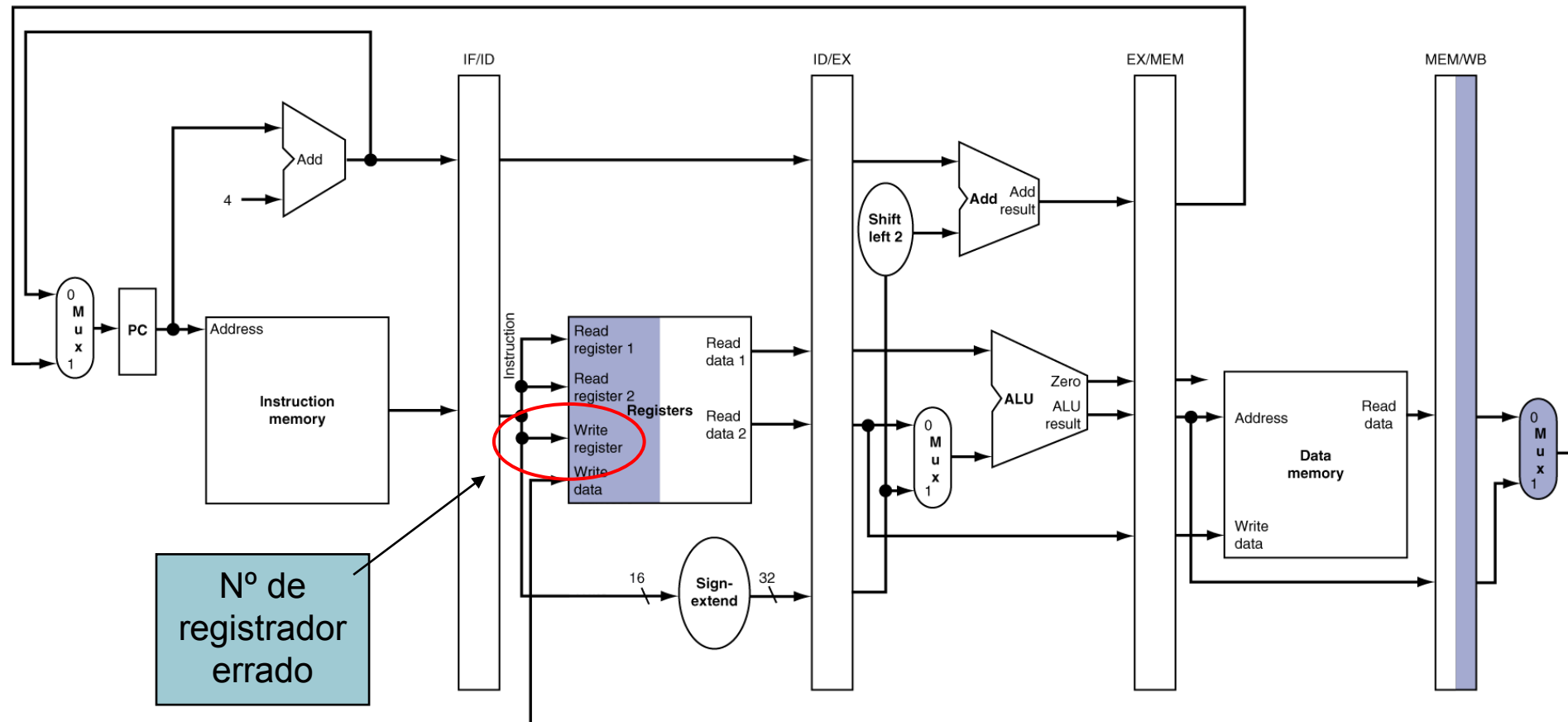


Novamente, encaminha-se todos os dados que possam ser necessários em estágios posteriores da instrução.

Qual a utilidade deste encaminhamento (dica: não é para o Load)? Tipo-R.

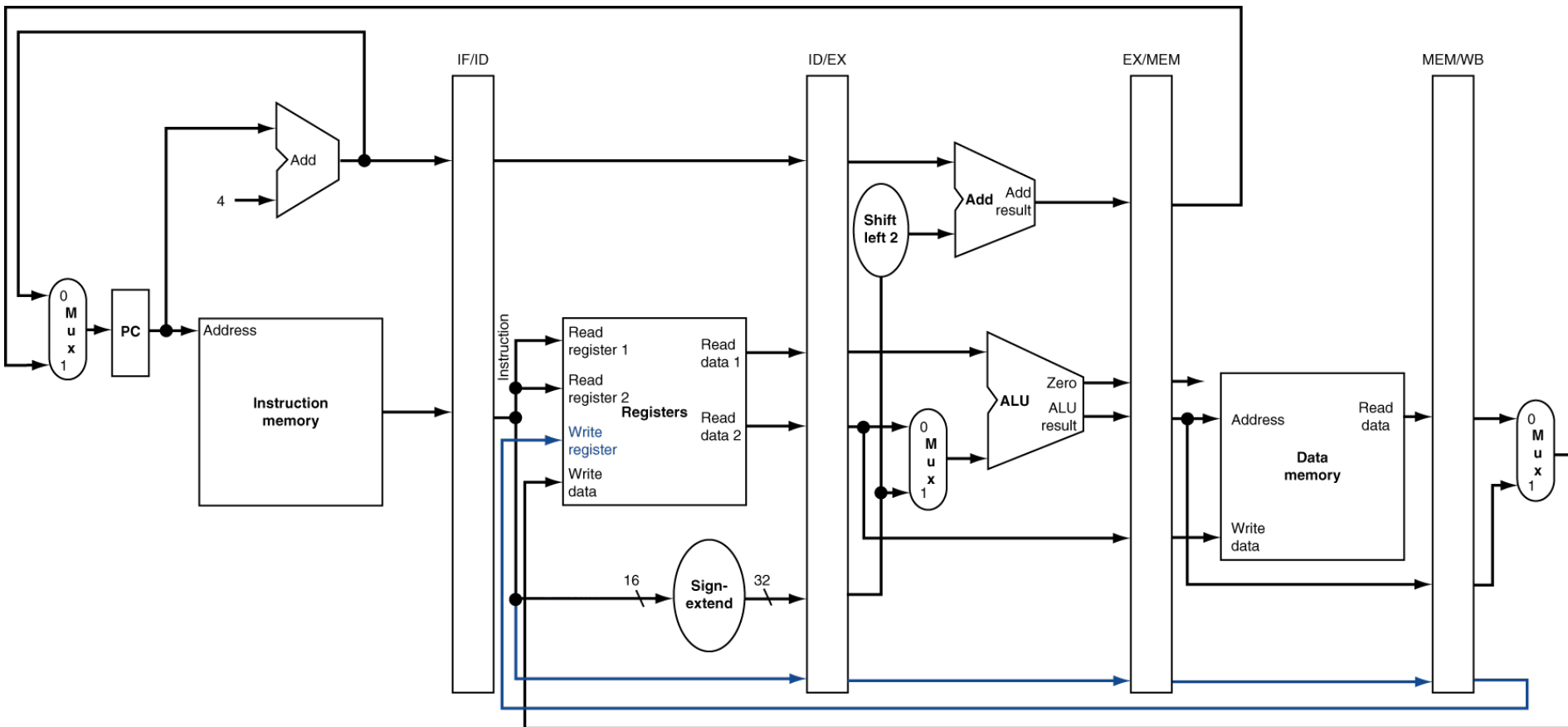
WB para Load

lw
Write back



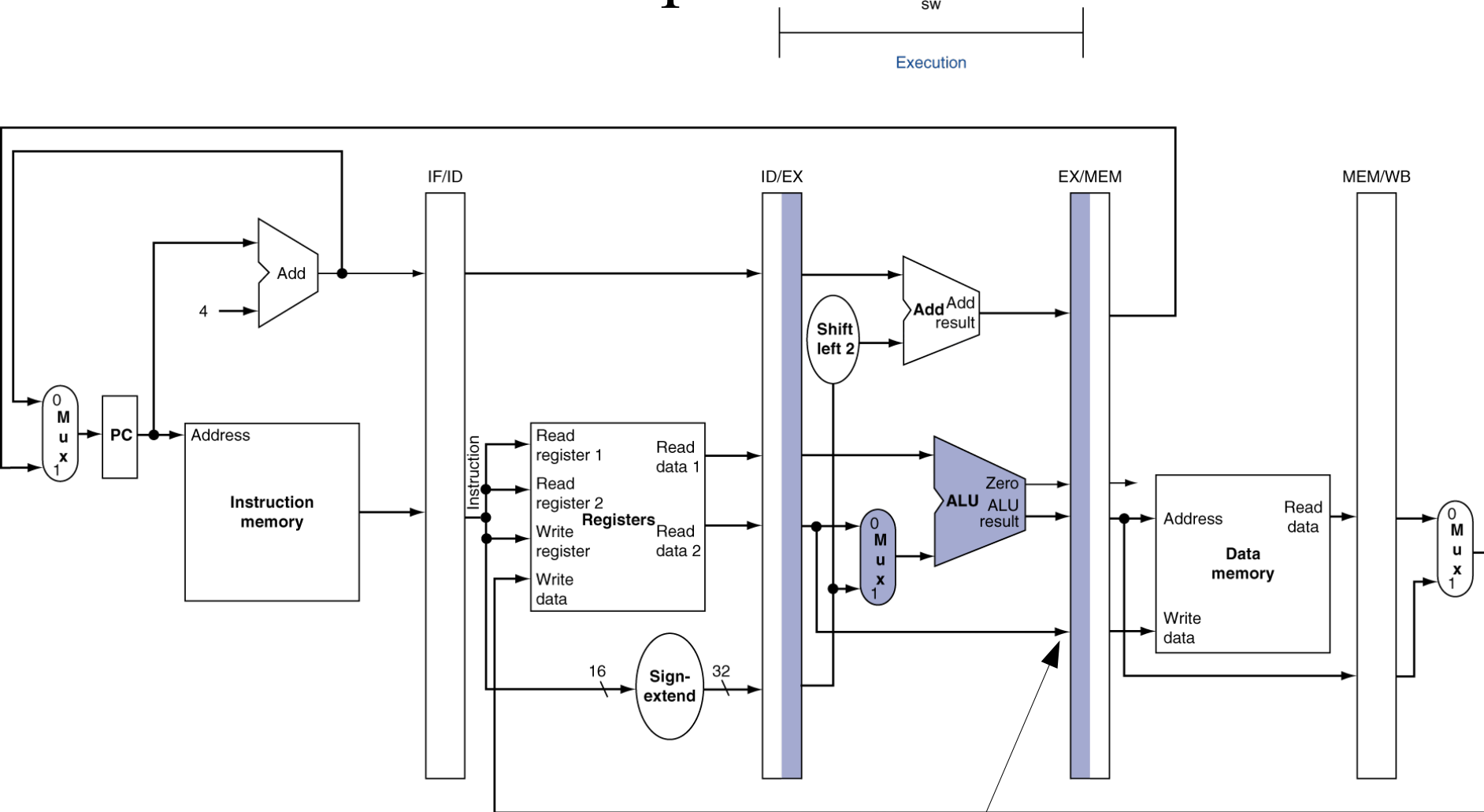
Como corrigir esse caminho de dados?

Datapath corrigido para o Load



Lembre-se do mantra: encaminhar todos os dados que possam ser necessários em estágios posteriores da instrução!

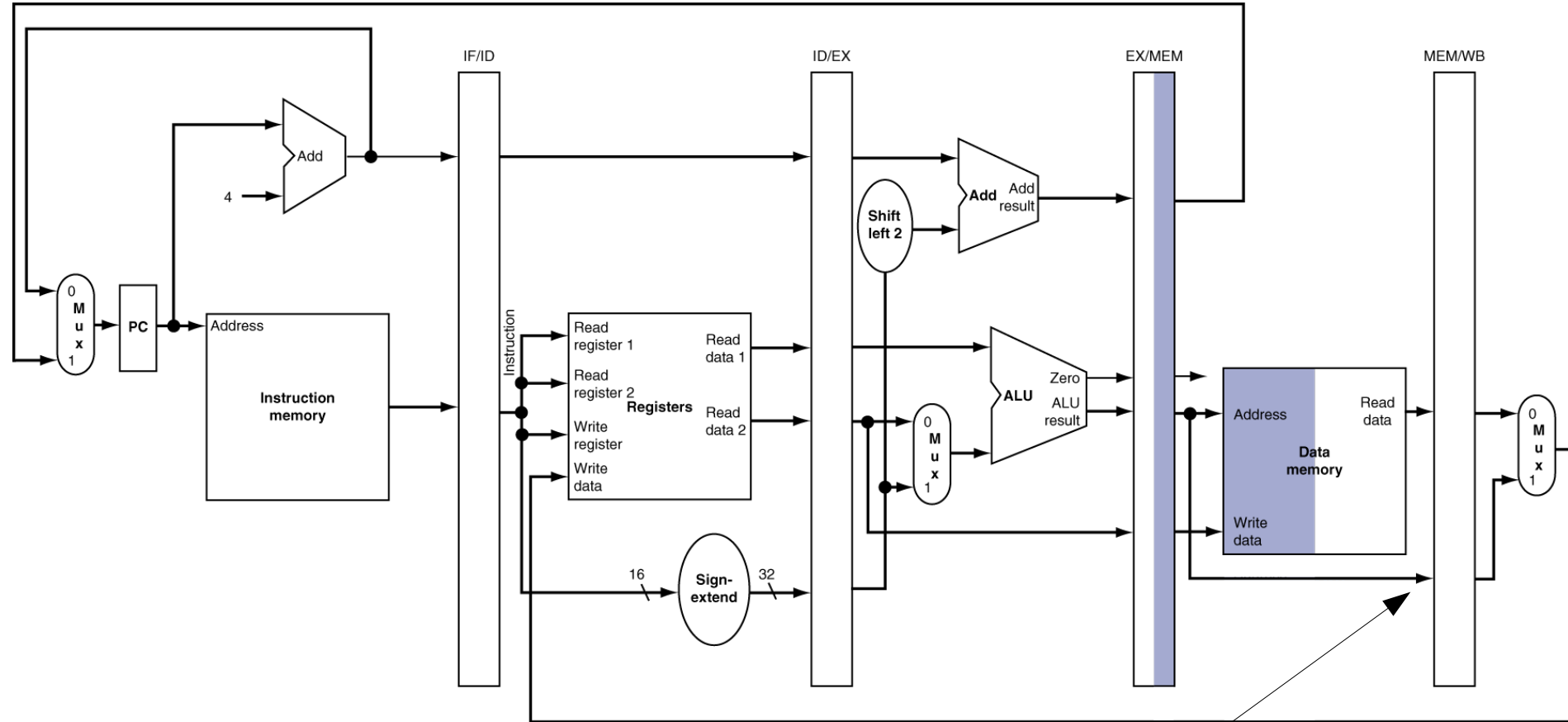
EX para Store



Lembrou-se do mantra, né?

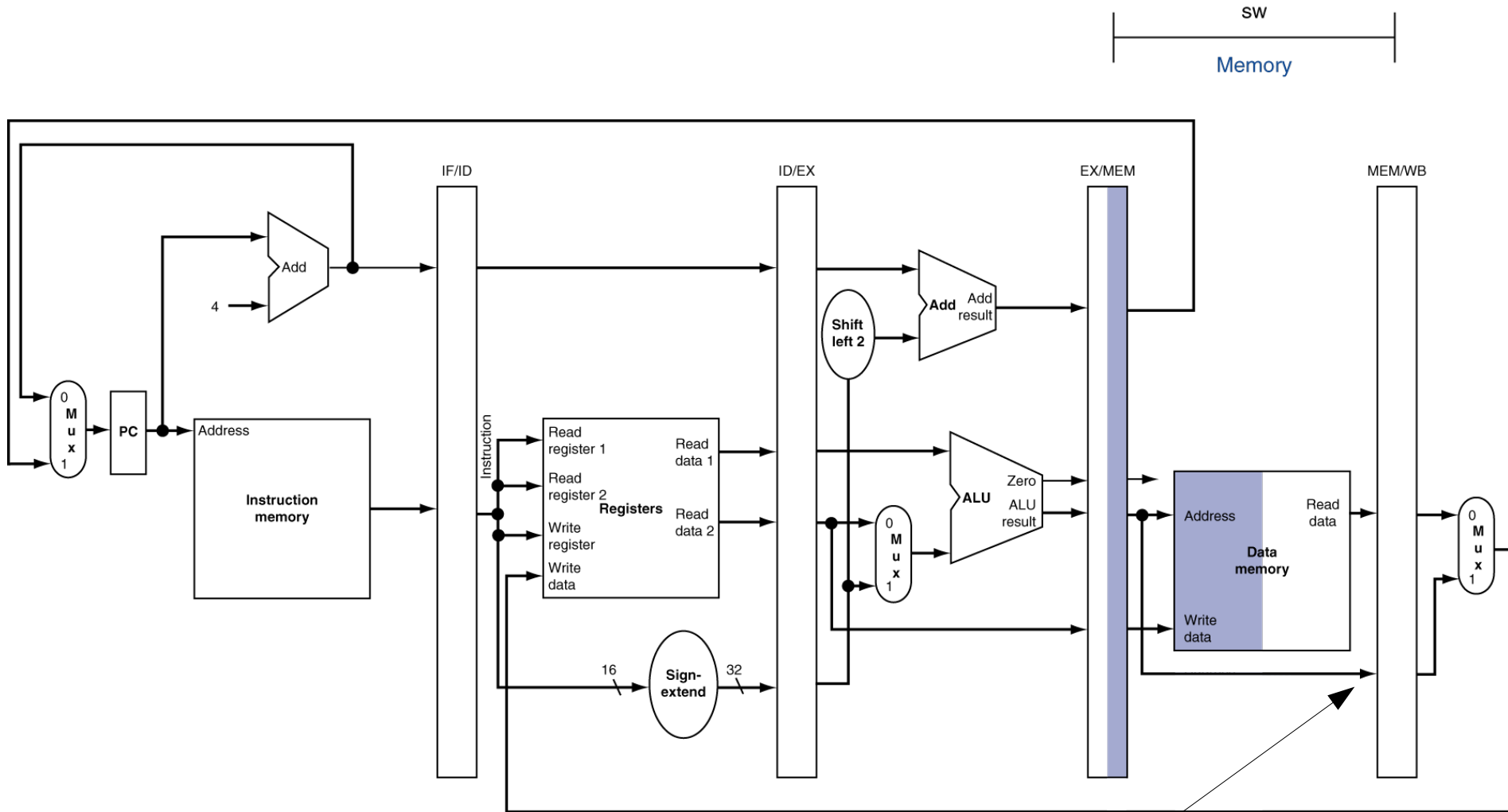
MEM para Store

SW
Memory



E nesse caso?
Porque o mantra está sendo usado?

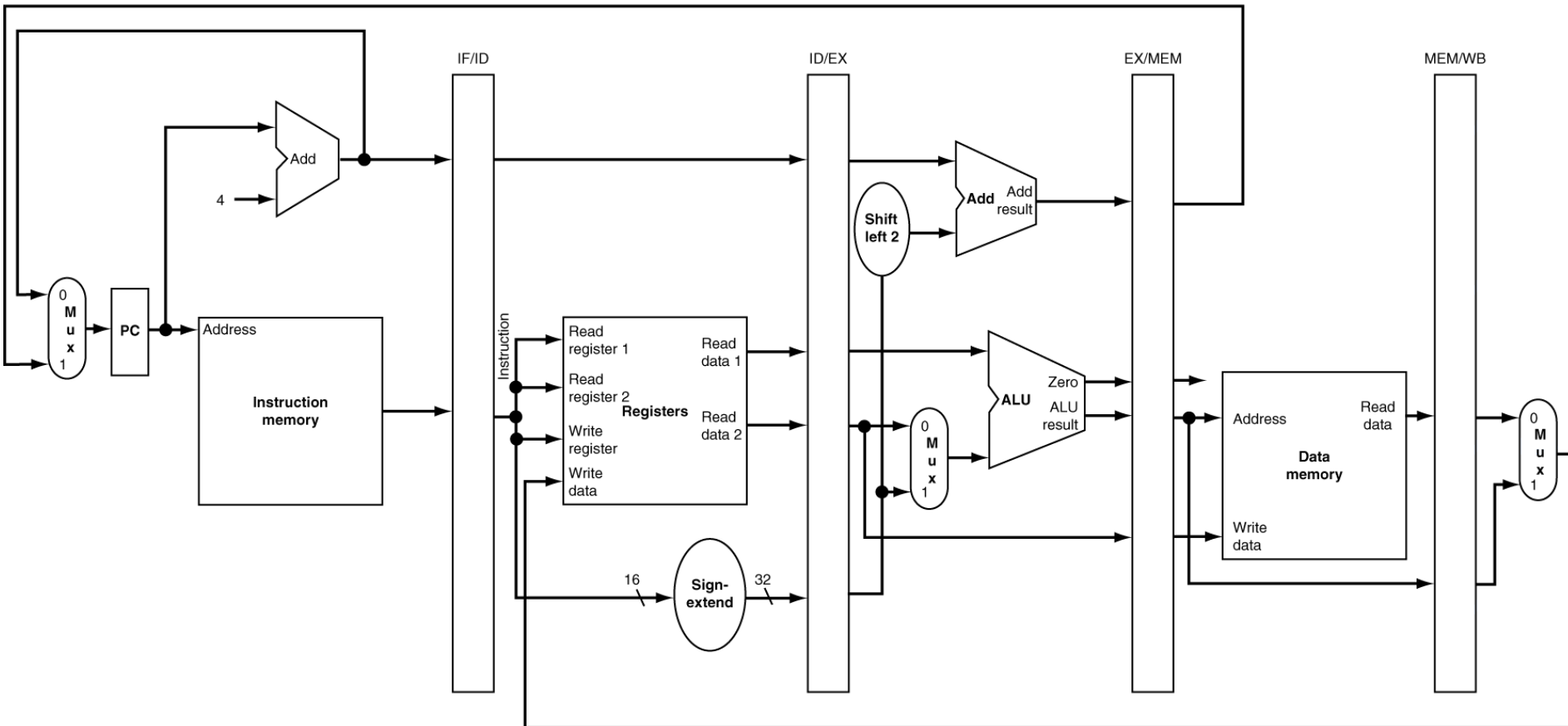
MEM para Store



E nesse caso?
Porque o mantra está sendo usado?
Por causa das instruções tipo-R.

WB para Store

SW
Write-back



Nada acontece!

- Como as instruções posteriores já estão em progresso, não há como acelerá-las.
- Uma instrução passa por um estágio mesmo que não haja nada a fazer!