

Discussão

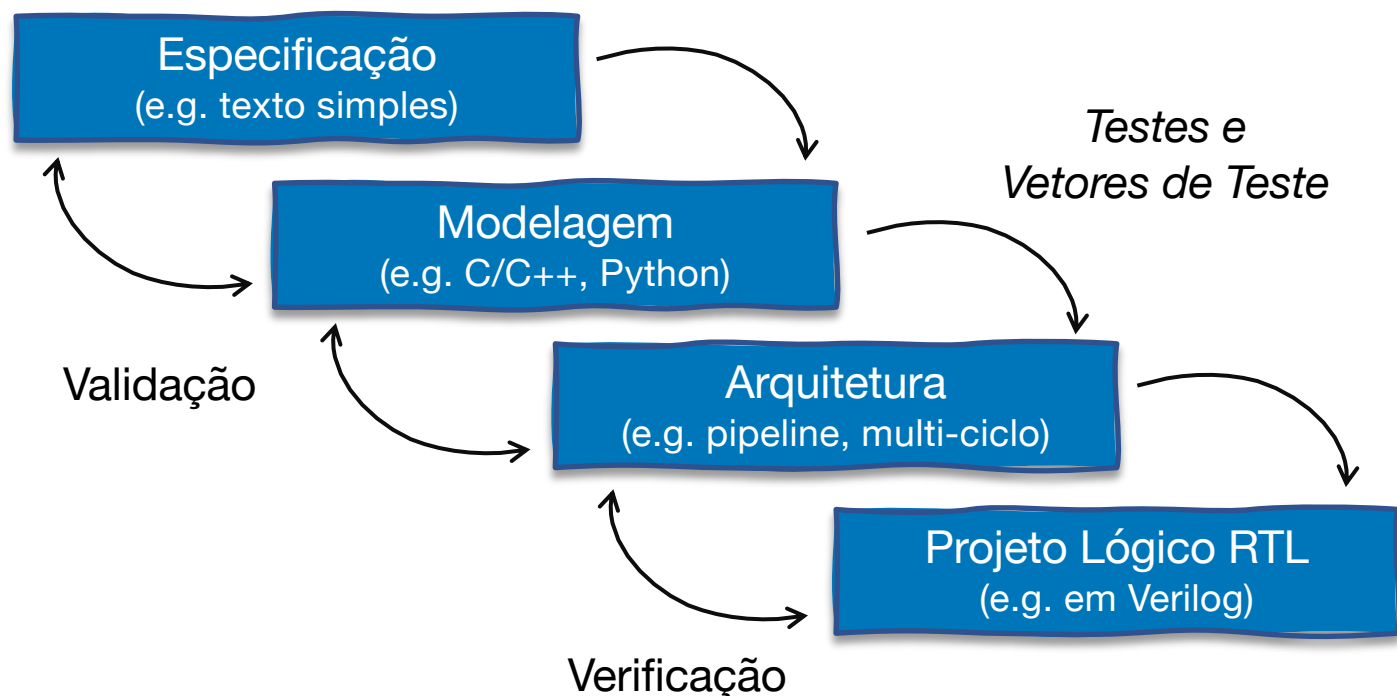
Projeto do Processador RISC231-M1

João Carlos Bittencourt

Projeto do Processador RISC231-M1

Especificação

Baseado no ISA MIPS R2000



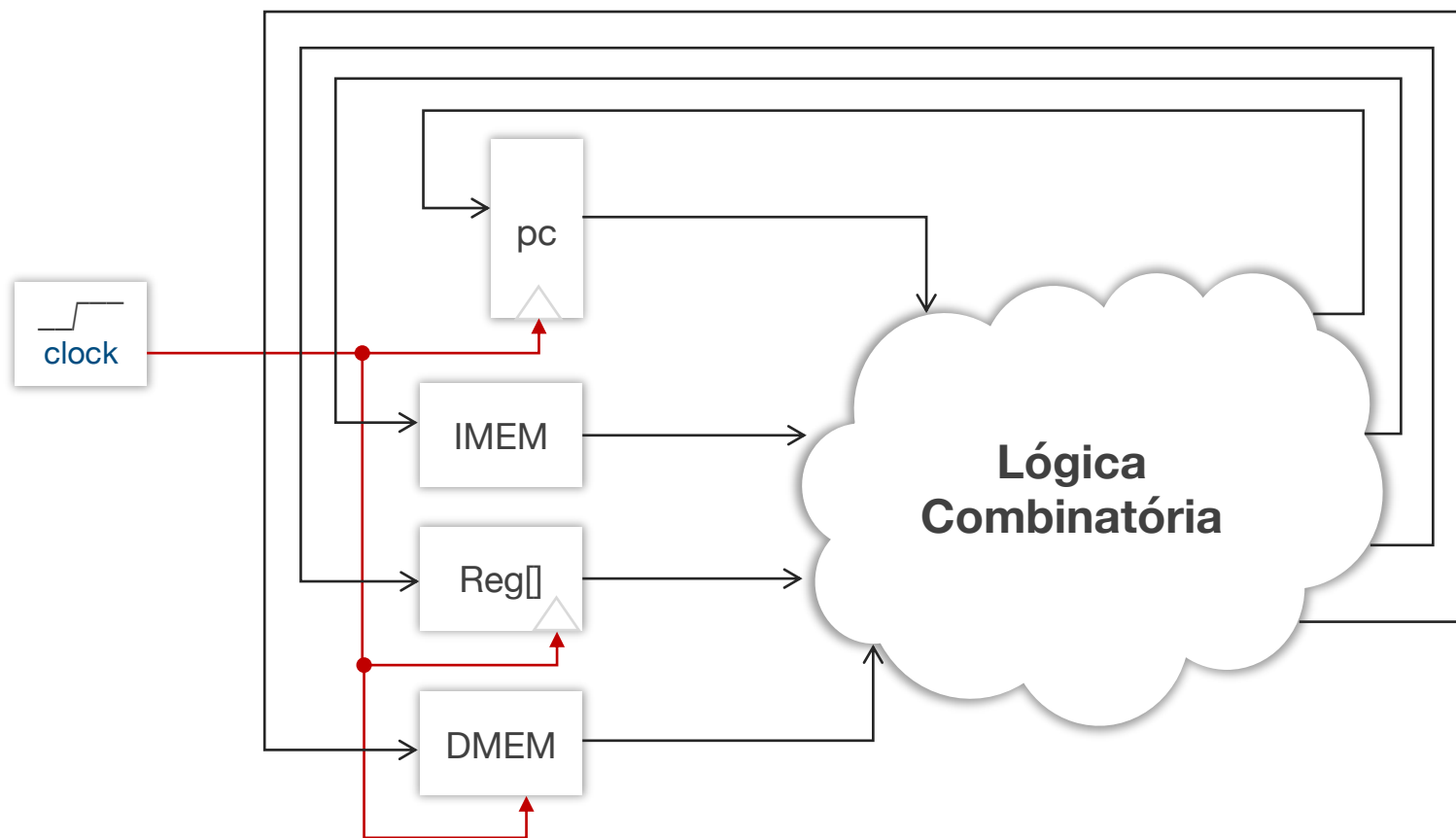
Testes e Vetores de Teste

Fornecidos como parte do roteiro de laboratório e projeto final

Arquitetura

Processador monociclo

RISC231-M



Em cada ciclo de clock o computador executa uma instrução

O estado atual alimenta a entrada da lógica combinatória.

Na borda do clock, todos os elementos de estados são atualizados com as saídas da lógica combinatória.

Elementos de Estado da ISA do RISC231

Cada instrução lê e atualizada esses elementos de estado durante a execução:

Banco de Registradores

Formado por 32 registradores de 32 bits

Três operandos: rs, rt, e rd

Registrador na posição 0 é sempre igual a 0 (ignora a escrita)

Contador de Programas

Armazena o endereço da instrução sendo executada.

Memórias

Armazena instruções e dados em um espaço de endereçamento de 32 bits

Usaremos memórias separadas para instrução e dados (como uma cache de dados/instrução)

Instruções são lidas da memória de instruções

Instruções Load/Store acessam a memória de dados

Estrutura do Caminho de Dados

Problema

Um único bloco de lógica combinatória “monolítico” que “executa uma instrução” (realiza todas as operações, começando com a leitura da instrução e terminando com o acesso aos registradores) é muito grande e ineficiente

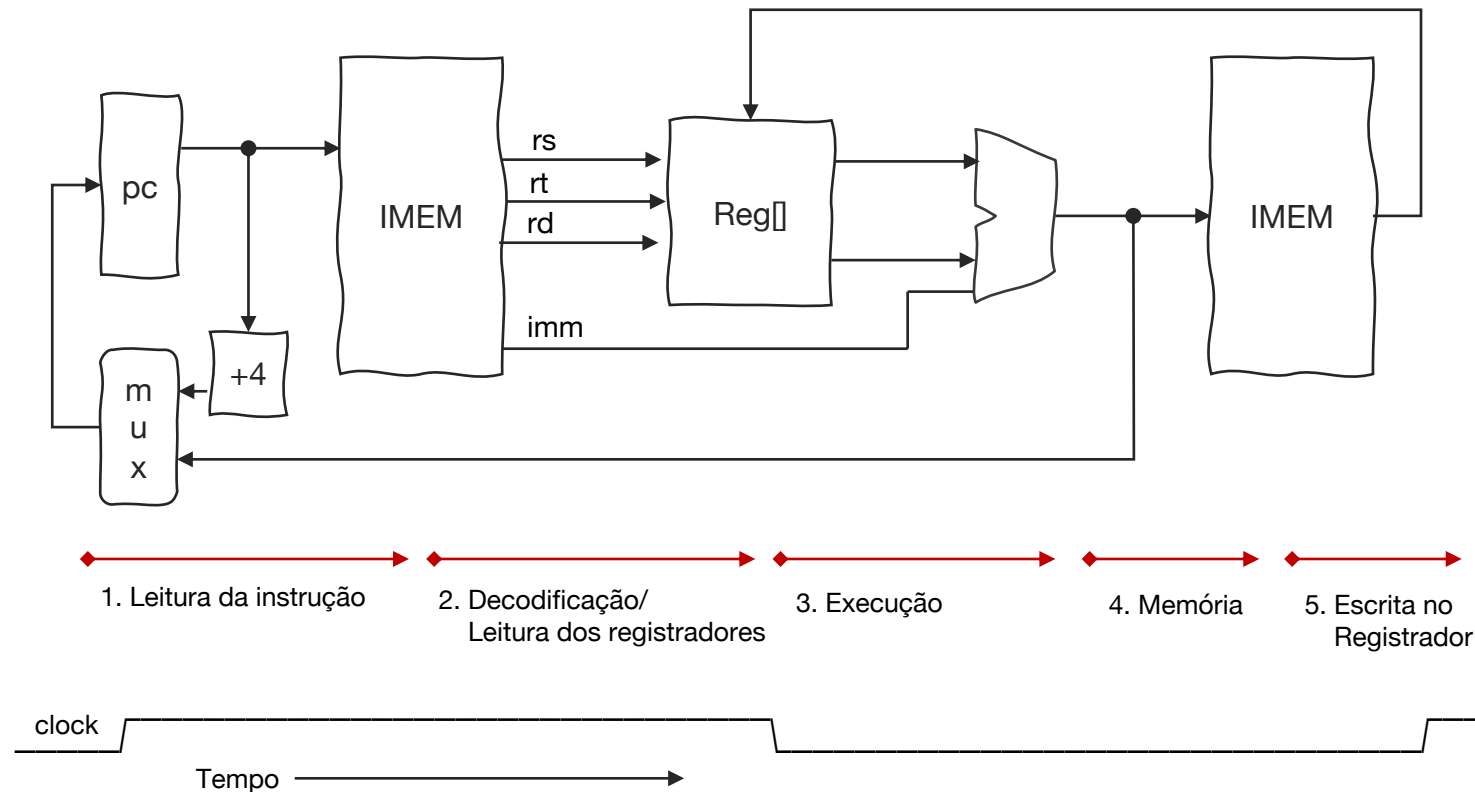
Solução

Dividir o processo de “execução de uma instrução” em estágios, e então conectar os estágios para criar um único caminho de dados

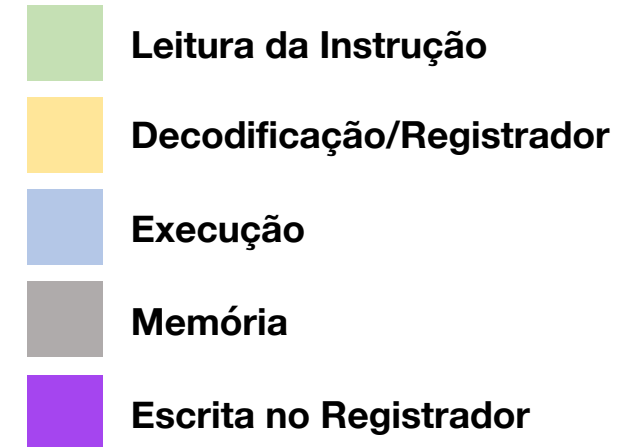
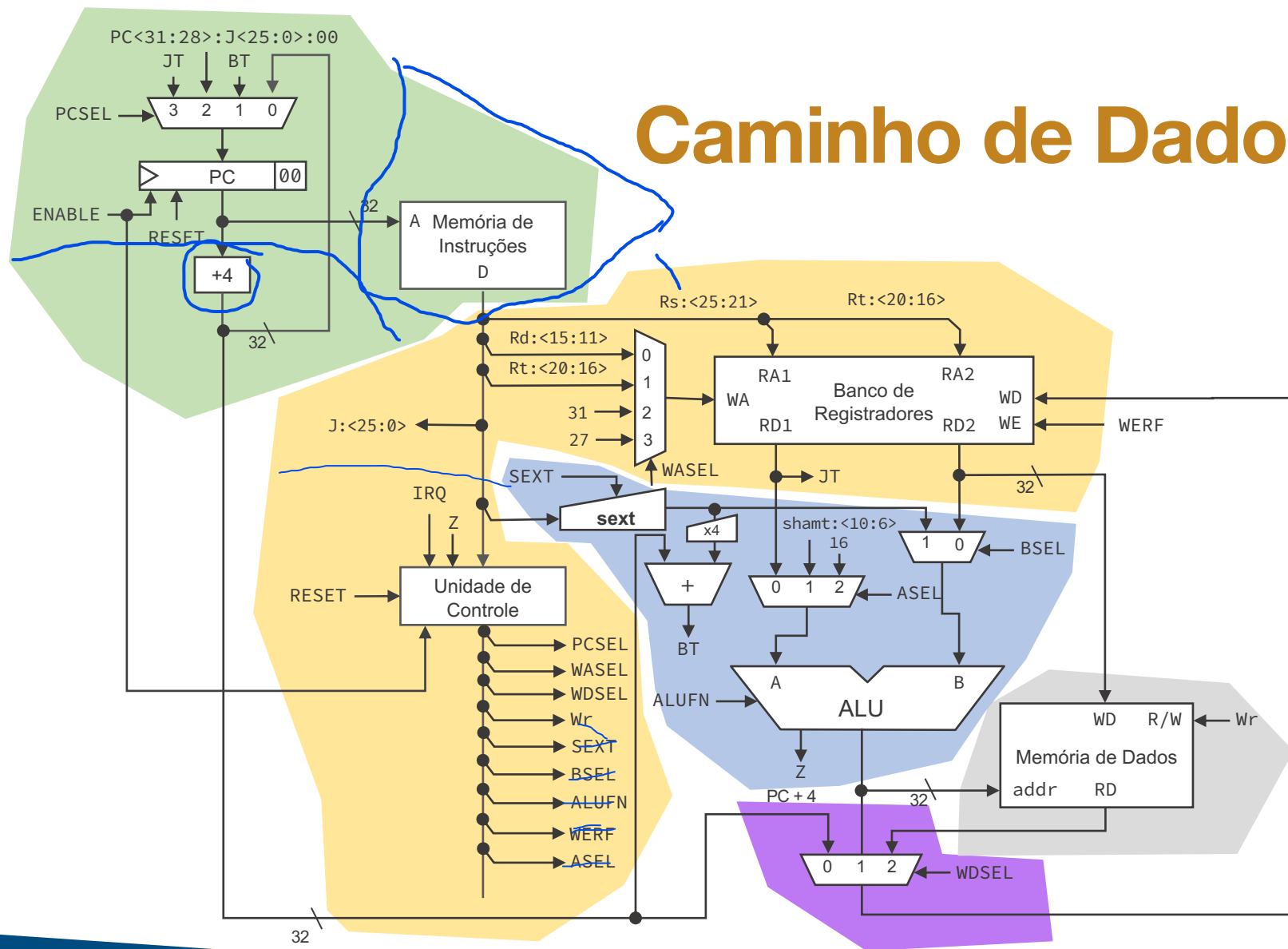
Estágios menores são mais simples de projetar

Mais fácil de otimizar (modificar) um estágio sem tocar nos outros (modularidade)

Caminho de Dados de Cinco Estágios

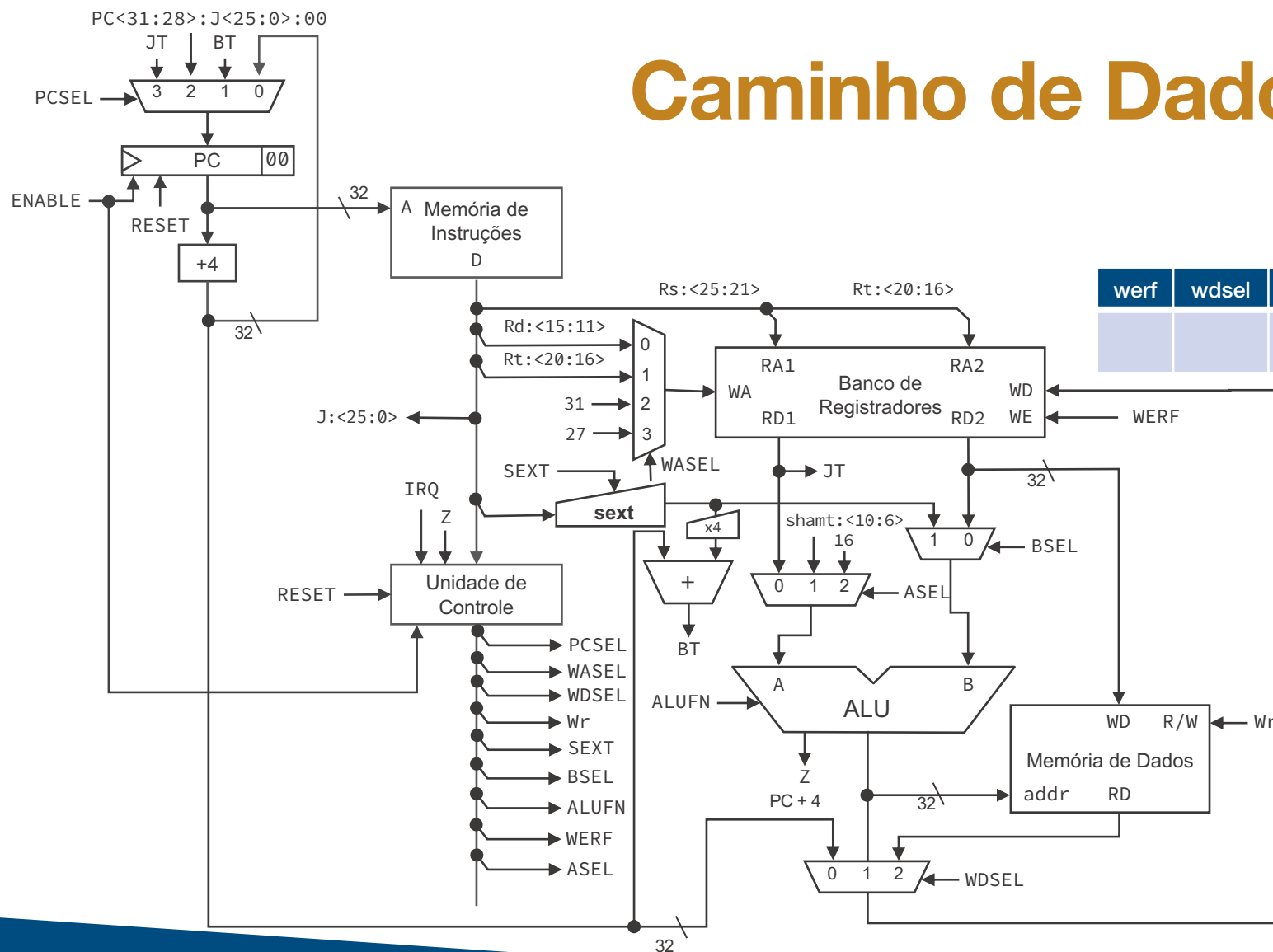


Caminho de Dados do RISC231



Caminho de Dados do RISC231

Instrução: add

[illegible]

Atividade Ativa
