

Estrutura de um CPU Básico

Paper técnico - versão académica

Autor: Francisco Gonçalves
Co-autoria editorial: Augustus
Data: 08-02-2026

Resumo

Este paper apresenta, de forma sistemática, a arquitectura de um CPU básico, descrevendo os seus blocos funcionais essenciais, o ciclo de instrução, o caminho de dados, o modelo de memória e os mecanismos elementares de controlo de execução. A análise enfatiza a relevância pedagógica do modelo clássico para a compreensão de arquitecturas modernas, incluindo pipeline, execução paralela e múltiplos núcleos.

Palavras-chave: CPU, ALU, unidade de controlo, registos, barramentos, ciclo fetch-decode-execute, ISA.

1. Introdução

Um processador central (CPU - Central Processing Unit) é a entidade computacional responsável por interpretar e executar instruções de programas. Mesmo em processadores contemporâneos, com elevada complexidade microarquitetural, subsistem os mesmos fundamentos conceptuais: controlo, processamento aritmético-lógico, armazenamento interno de alta velocidade e comunicação com memória e periféricos.

2. Arquitectura de referência de um CPU básico

Num modelo elementar, de inspiração von Neumann, instruções e dados partilham o mesmo espaço de memória. O CPU integra, no mínimo, os seguintes blocos: Program Counter (PC), Instruction Register (IR), Unidade de Controlo (UC), Unidade Lógica e Aritmética (ALU), banco de registos, registo de estado (flags) e interface de memória.

3. Blocos funcionais

3.1 Unidade de Controlo (UC)

A UC coordena o funcionamento interno do processador. Ao descodificar o opcode da instrução em IR, gera sinais de controlo para seleccionar fontes e destinos de dados, accionar operações da ALU, controlar leituras/escritas em memória e actualizar o PC.

3.2 Unidade Lógica e Aritmética (ALU)

A ALU executa operações aritméticas (ADD, SUB, INC, DEC), lógicas (AND, OR, XOR, NOT) e operações de deslocamento (SHL, SHR). O resultado actualiza, quando aplicável, flags como Zero (Z), Carry (C), Sign (S) e Overflow (V).

3.3 Registos internos

Os registos constituem a memória mais rápida da máquina. Entre os registos essenciais num CPU básico destacam-se PC, IR, ACC (em arquitecturas com acumulador), SP (Stack Pointer), FLAGS/PSW e registos de propósito geral (R0...Rn).

3.4 Barramentos

A comunicação interna e externa ao CPU utiliza três barramentos lógicos: dados, endereços e controlo. A largura destes barramentos condiciona o volume de dados por transferência e o espaço endereçável de memória.

3.5 Clock

O relógio define a cadência temporal das micro-operações. Cada instrução é decomposta em passos sincronizados por ciclos de clock.

4. Ciclo de instrução

O ciclo clássico fetch-decode-execute organiza a execução sequencial de instruções. Na fase de fetch, a instrução é lida da memória para o IR; na fase de decode, a UC interpreta a operação e prepara sinais; na fase de execute, a ALU e/ou a interface de memória realizam a acção pretendida, com escrita do resultado e eventual actualização do PC.

5. Caminho de dados (datapath)

O datapath é o conjunto de elementos que transportam e transformam dados: banco de registos, multiplexadores, ALU, registos intermédios e barramentos internos. A UC programa este percurso por meio de sinais de selecção.

6. Memória, latência e hierarquia

No modelo básico, assume-se memória principal única. Em implementações reais, é introduzida uma hierarquia (registos, cache L1/L2/L3, RAM, armazenamento secundário) para reduzir a latência média de acesso e elevar desempenho efectivo.

7. Interrupções e excepções

Interrupções permitem reagir a eventos assíncronos (E/S, temporizador, sinais externos). O CPU preserva contexto mínimo (tipicamente PC e FLAGS), transfere controlo para a rotina de serviço e retoma posteriormente o fluxo interrompido.

8. ISA mínima de um CPU básico

Uma ISA elementar inclui instruções de transferência (LOAD, STORE, MOV), aritmética e lógica (ADD, SUB, AND, OR, XOR), controlo de fluxo (JMP, JZ, JNZ, CALL, RET) e controlo de sistema (NOP, HLT, INT).

9. Modelo quantitativo de desempenho

O tempo de CPU pode ser aproximado por: $\text{Tempo} = \text{Número de instruções} \times \text{CPI} \times \text{período do clock}$. A optimização pode actuar em qualquer termo da expressão: redução de instruções, redução de CPI ou aumento da frequência efectiva.

10. Limitações e evolução

Um CPU básico executa instruções sobretudo de forma sequencial, com paralelismo limitado. A evolução microarquitectural introduz pipeline, execução superscalar, predição de desvios, execução fora de ordem, vectorização e multicores, mantendo porém os mesmos fundamentos estruturais.

11. Conclusão

A compreensão da estrutura de um CPU básico é determinante para engenharia informática, sistemas embebidos e optimização de software. O modelo clássico continua a ser o alicerce conceptual sobre o qual assentam as arquitecturas contemporâneas.

Referências

1. PATTERSON, David A.; HENNESSY, John L. Computer Organization and Design: The Hardware/Software Interface. Morgan Kaufmann.
2. STALLINGS, William. Computer Organization and Architecture. Pearson.
3. TANENBAUM, Andrew S.; AUSTIN, Todd. Structured Computer Organization. Pearson.
4. MANO, M. Morris; CILETTI, Michael D. Digital Design. Pearson.
5. HARRIS, David; HARRIS, Sarah. Digital Design and Computer Architecture. Morgan Kaufmann.