

Organização de Computadores – 2005/2006

Barramentos e Processadores

Paulo Ferreira
paf@dei.isep.ipp.pt

Março de 2006

Barramentos	2
Definição	3
Especificações	4
Exemplo	5
Perguntas	6
Memória	7
Tamanho	8
Explicação	9
Arquitecturas	10
Arquitetura de Harvard	11
Memória	12
Von Neumman	13
Harvard	14
Von Neumman	15
Harvard	16

Perguntas

- Uma lâmpada acende ao mesmo tempo que outra apaga?
- Haverá algum intervalo de tempo no qual estão duas lâmpadas acesas?
- Haverá algum intervalo de tempo no qual estão todas as lâmpadas apagadas?
- Isto poderá trazer problemas nos circuitos que usam este como entrada. . .
- Podemos adicionar linhas adicionais que dizem se os dados estão válidos ou não e especificar um protocolo para o funcionamento do barramento
- Além das linhas para «transmitir» os bits, necessitamos de outras para «sincronizar» o funcionamento destas

ORGC

Barramentos e Processadores – slide 6

Memória

- Como ligar um processador à memória?
- Exemplo: Z80 64kbytes de memória = 512kbits de memória
 - 1 linha de ligação a cada bit de memória não é prático. . .
- Vamos ligar apenas um barramento de 8 bits à memória
- Cada vez que acedemos à memória apenas conseguimos ler ou escrever 1 byte
- Para escolher qual a posição de memória a que queremos aceder usamos sinais adicionais chamados endereços porque endereçam a memória

ORGC

Barramentos e Processadores – slide 7

Tamanho

- Podemos dizer que a memória é um vector de elementos em que:
 - Cada elemento tem tantos bits como o barramento de dados
 - O número de elementos que pode existir depende do número de bits do barramento de endereços
- A quantidade de memória máxima que um processador suporta depende de:
 - $ABus$ – número de bits do barramento de endereços
 - $DBus$ – número de bits do barramento de dados
- $Mem = DBus * 2^{ABus}$

ORGC

Barramentos e Processadores – slide 8

Explicação

- $Mem = DBus * 2^{ABus}$
- A memória é um «rectângulo» com dois lados:
 - O lado mais pequeno tem a largura do barramento de dados
 - O lado maior depende do número de posições de memória que conseguimos endereçar isto é: 2^{ABus}
- Para fazer com que a memória suportada por um processador seja o dobro da suportada por outro podemos:
 1. Adicionar apenas mais um bit ao barramento de endereços
 2. Duplicar a largura do barramento de dados
- A primeira opção é mais simples mas a segunda dá uma maior performance

ORGC

Barramentos e Processadores – slide 9

Arquitecturas

slide 10

Arquitectura de Harvard

http://www.gsd.harvard.edu/library/map/building_architects.html

- Alvar Aalto – Poetry Room (in Lamont Library)
- Walter Gropius – Harkness Commons
- Graham Gund – The Gate Lodge at the Johnson Gate
- Le Corbusier – with Sert, Jackson & Gourley – Carpenter Center
- H. H. Richardson – Austin Hall and Sever Hall
- Jose Luis Sert – Holyoke Center, Center for World Religions, and Undergraduate Science Center
- James Stirling – Sackler Museum
- Hugh Stubbins – Loeb Drama Center, Pusey Library
- :-) :-) :-) :-) :-) :-)

ORGC

Barramentos e Processadores – slide 11

Memória

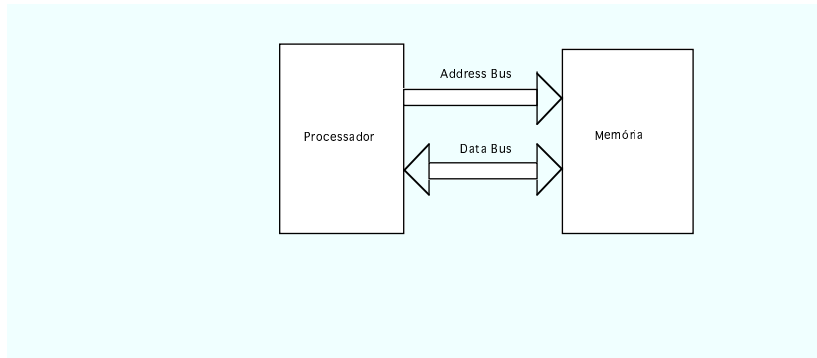
- Para que serve?
 - Armazenar os programas
 - Armazenar os dados
- Programa e dados são coisas diferentes
- Mas são guardados na memória...

ORGC

Barramentos e Processadores – slide 12

Von Neumman

- Na realidade de outros (Eckert & Mauchly)...
- Há só uma memória que serve para dados e programa

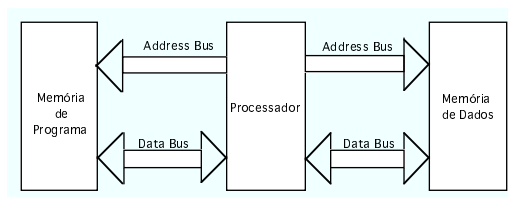


ORGC

Barramentos e Processadores – slide 13

Harvard

- Memórias separadas para dados e programa



ORGC

Barramentos e Processadores – slide 14

Von Neumman

- Vantagens:
 - Simplicidade
 - Um compilador trata como dados aquilo que mais tarde será código
- Desvantagens:
 - Performance
- Usos:
 - Quase todos os processadores usados hoje em dia

ORGC

Barramentos e Processadores – slide 15

Harvard

■ Vantagens:

- Performance (acesso simultâneo a instruções e dados)

■ Desvantagens:

- Duas memórias + dois barramentos
- Diferença entre código e dados

■ Usos:

- Microcontroladores (código em ROM e dados em RAM)
- Caches internas dos processadores

ORGC

Barramentos e Processadores – slide 16