

Aula 05 - Memórias e Endereçamento 1/2

Prof. Sergio R. M. Canovas

PCS - Departamento de Engenharia de Computação e Sistemas Digitais
Escola Politécnica da Universidade de São Paulo

Setembro, 2020

Agenda

Introdução

Terminologia

ROMs

RAMs

1 Introdução

2 Terminologia

3 ROMs

4 RAMs

Ao final da aula você saberá:

Introdução

Terminologia

ROMs

RAMs

- O que são memórias e os conceitos de endereço e palavra de dados;
- O que são memórias ROM e RAM;
- Notações de tamanhos de memórias e quantidades em potências de 10 e potências de 2;
- Definições precisas da terminologia sobre memórias;
- Como utilizar ROMs e RAMs em operações de leitura e escrita;
- Tipos de implementação de memórias ROM e RAM.

Introdução

Introdução

Terminologia

ROMs

RAMs

Introdução

Introdução

Introdução

Terminologia

ROMs

RAMs

- A principal vantagem dos sistemas digitais sobre os analógicos é a capacidade de armazenar facilmente grandes quantidades de dados por períodos longos ou curtos [3];
- Esta capacidade é obtida com o uso de memórias;
- Existem diversos tipos de memórias e formas de endereçá-las, mas vamos iniciar com uma abstração simples;

Introdução

Introdução

Terminologia

ROMs

RAMs

- Uma **memória** pode ser vista como um vetor unidimensional para armazenamento de dados, que oferece acesso a cada posição do vetor;
- Cada posição do vetor contém um dado e possui um **endereço** de acesso, que corresponde ao índice do vetor;

Introdução

Introdução

Terminologia

ROMs

RAMs

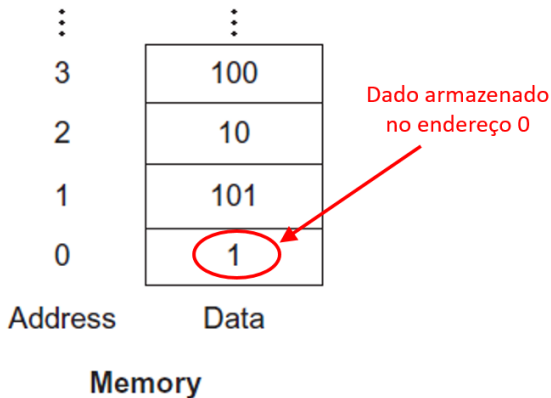


Figura: Endereçamento - Fonte: [2].2 - Figura 2.2 - adaptado

Introdução

Introdução

Terminologia

ROMs

RAMs

- A memória armazena o conteúdo de cada posição do vetor sem perdê-lo com o tempo. Cada tecnologia tem um funcionamento específico;
- Trata-se de um dos cinco componentes clássicos de um computador [2].1:
 - Dispositivos de entrada;
 - Dispositivos de saída;
 - **Memória**;
 - Unidade de controle;
 - Fluxo de dados (*datapath*);

Introdução

Introdução

Terminologia

ROMs

RAMs

- A memória possui um certo **tamanho da palavra de dados**, que corresponde ao número de bits de dados que são armazenados em cada posição endereçada, que é fixo;
- Exemplo: se a palavra de dados for de 8 bits, cada posição do vetor conterá 8 bits de dados (1 byte);

Introdução

Introdução

Terminologia

ROMs

RAMs

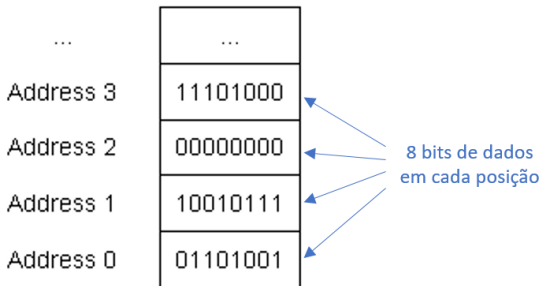


Figura: Fonte: <https://www.learncpp.com/cpp-tutorial/introduction-to-fundamental-data-types/comment-page-2/>

Introdução

Introdução

Terminologia

ROMs

RAMs

- Mesmo com um tamanho de palavra maior (64 bits, por exemplo), é comum a utilização de endereçamento byte a byte;

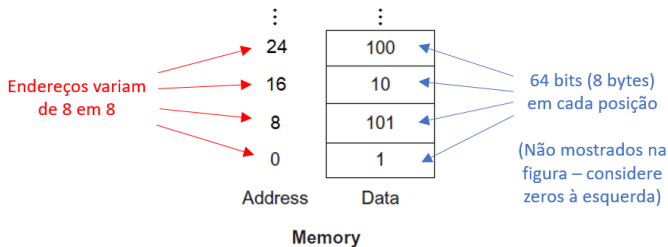


Figura: Endereçamento - Fonte: [2].2 - Figura 2.3 - adaptado

Introdução

Introdução

Terminologia

ROMs

RAMs

- Memórias possuem tamanho finito, chamado **capacidade**;
- Uma forma de expressar a capacidade de uma memória é denotar $NP \times TP$ [2].11, onde:
 - NP = Número de palavras;
 - TP = Tamanho de cada palavra em bits;
- Exemplo: **4096 x 20** denota uma capacidade de 4096 palavras de 20 bits cada, totalizando 81.920 bits;

Introdução

Introdução

Terminologia

ROMs

RAMs

Decimal term	Abbreviation	Value	Binary term	Abbreviation	Value	% Larger
kilobyte	KB	10^3	kibibyte	KiB	2^{10}	2%
megabyte	MB	10^6	mebibyte	MiB	2^{20}	5%
gigabyte	GB	10^9	gibibyte	GiB	2^{30}	7%
terabyte	TB	10^{12}	tebibyte	TiB	2^{40}	10%
petabyte	PB	10^{15}	pebibyte	PiB	2^{50}	13%
exabyte	EB	10^{18}	exbibyte	EiB	2^{60}	15%
zettabyte	ZB	10^{21}	zebibyte	ZiB	2^{70}	18%
yottabyte	YB	10^{24}	yobibyte	YiB	2^{80}	21%

FIGURE 1.1 The 2^x vs. 10^y bytes ambiguity was resolved by adding a binary notation for all the common size terms. In the last column we note how much larger the binary term is than its corresponding decimal term, which is compounded as we head down the chart. These prefixes work for bits as well as bytes, so *gigabit* (Gb) is 10^9 bits while *gibibits* (Gib) is 2^{30} bits.

Figura: Fonte: [2].1 - Figura 1.1

Introdução

Introdução

Terminologia

ROMs

RAMs

- O endereço serve para determinar qual palavra de dados, isto é, qual posição do vetor, desejamos acessar;
- Imaginando uma memória somente-leitura (ROM) como uma caixa preta, colocamos o endereço na entrada de endereço e o conteúdo da palavra de dados armazenada na posição correspondente será disponibilizado na saída de dados;

Introdução

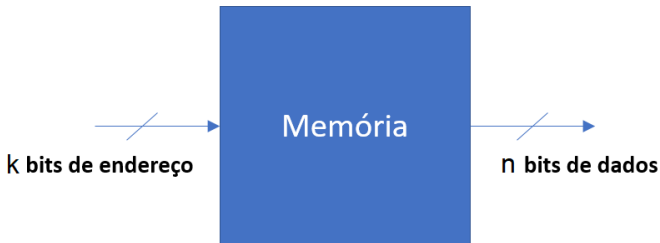
Introdução

Terminologia

ROMs

RAMs

- Lembrando que nos circuitos digitais representamos os números em binário;



Introdução

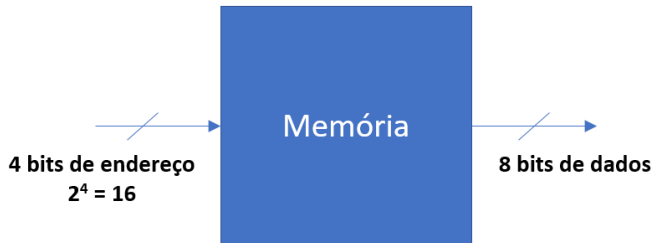
Introdução

Terminologia

ROMs

RAMs

- Exemplo: memória 16 x 8;



Introdução

Introdução

Terminologia

ROMs

RAMs

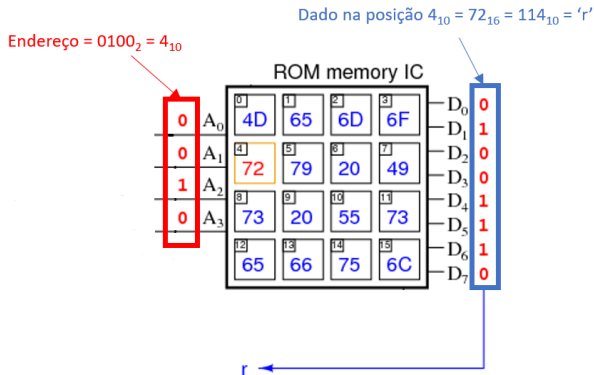


Figura: Fonte:

https://thumbs.gfycat.com/ForkedIdleHackee-size_restricted.gif - adaptado

Introdução

Introdução

Terminologia

ROMs

RAMs

- Vejamos um exemplo de circuito integrado de memória RAM (as terminologias serão revistas a seguir):
 - Circuito integrado 6116;
- Diferentemente da ROM, uma RAM permite a escrita de dados durante o funcionamento do circuito de forma fácil;

Introdução

Introdução

Terminologia

ROMs

RAMs

11 bits para
endereçamento.

Qual o tamanho
do "vetor"
armazenado?

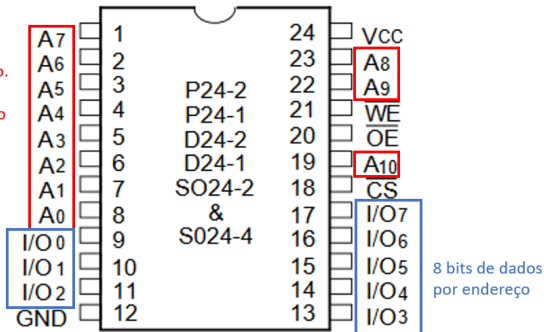


Figura: Fonte:

<http://www.princeton.edu/mae412/HANDOUTS/Datasheets/6116.pdf>

Introdução

Introdução

Terminologia

ROMs

RAMs

- $2^{11} = 2048 = 2\text{Ki}$ posições;
- 8 bits de dados = 1 byte = 1B;
- Portanto, trata-se de uma memória de 2KiB, mais especificamente $2\text{Ki} \times 8$;
- Observe que os pinos de dados nesse *chip* servem tanto para leitura quanto para escrita;
- As entradas \overline{CS} (*Chip Select*), \overline{OE} (*Output Enable*) e \overline{WE} (*Write Enable*) controlam a leitura/escrita;

Introdução

Introdução

Terminologia

ROMs

RAMs

- Uma vez que escrevemos um dado em uma posição (endereço) da memória RAM, enquanto ela dispor de alimentação elétrica, o dado pode ser consultado ao se fazer uma leitura naquele mesmo endereço;
- Em um computador, a memória RAM armazena programas na forma de sequências de instruções (ex. Word) e dados utilizados por esses programas (ex. o documento Word no qual o usuário está trabalhando).

Terminologia

Introdução

Terminologia

ROMs

RAMs

Terminologia

Terminologia

Introdução

Terminologia

ROMs

RAMs

A terminologia a seguir é baseada em [3].11:

- **Célula de memória:** dispositivo ou circuito utilizado para armazenar um único bit;
- **Capacidade:** quantidade de bits que podem ser armazenados pela memória. Já vimos as formas de denotar, mas em geral é expressa apenas em número de bytes (ex.: 8GiB de RAM, HD de 1TiB, etc.);
- **Densidade:** capacidade da memória dividida pela área física ocupada pelas células;

Terminologia

Introdução

Terminologia

ROMs

RAMs

- **Operação de leitura:** operação na qual a palavra binária armazenada em uma posição (endereço) é obtida pelo circuito ou sistema que está utilizando a memória;
- **Operação de escrita:** operação na qual uma nova palavra é colocada em uma determinada posição (endereço). Também chamada de **armazenamento** (*store*). A nova palavra substitui a palavra anterior, o mesmo endereço não armazena duas palavras simultaneamente;
- **Tempo de acesso:** tempo necessário para realizar uma operação de leitura, isto é, o tempo entre a memória receber uma nova entrada de endereço e os dados se tornarem disponíveis na saída. O mesmo termo pode ser utilizado também para uma operação de escrita;

Terminologia

Introdução

Terminologia

ROMs

RAMs

- **RAM (*Random-Access Memory*)**: memória na qual a posição física de uma palavra não afeta o tempo de acesso, ou seja, o tempo de acesso pode ser considerado fixo para qualquer endereço;
- **SAM (*Sequential-Access Memory*)**: memória na qual o tempo de acesso não é constante, variando conforme o endereço. Exemplo: fita magnética. Processo relativamente lento;

Terminologia

Introdução

Terminologia

ROMs

RAMs

- **RWM (*Read/Write Memory*)**: memória que pode ser lida ou escrita de maneira igualmente fácil;
- **ROM (*Read-Only Memory*)**: apesar do nome, são memórias nas quais o número de operações de leitura durante a operação é muito maior que o número de operações de escrita. Algumas podem ser escritas somente uma vez, enquanto outras podem ser escritas mais de uma vez, mas a operação de escrita é mais complicada do que a leitura e não é realizada frequentemente;

Terminologia

Introdução

Terminologia

ROMs

RAMs

- **Memórias voláteis:** requerem alimentação elétrica para manter o armazenamento dos dados;
- **Memórias não-voláteis:** continuam armazenando os dados mesmo quando desligadas da alimentação elétrica;

Importante:

- Apesar das definições acima, o termo ROM é comumente utilizado para se referir a memórias não-voláteis;
- O termo RAM é comumente utilizado para se referir a memórias voláteis de acesso aleatório;
- Se formos rigorosos com os termos, uma ROM convencional também é uma RAM, pois o tempo de acesso a cada endereço é fixo, mas não nos referimos a elas como RAMs.

ROMs

Introdução

Terminologia

ROMs

RAMs

ROMs

ROMs

Introdução

Terminologia

ROMs

RAMs

- Uma ROM é essencialmente um dispositivo em que informação binária permanente é armazenada [1].5, sendo não-volátil;
- Uma ROM que não seja reprogramável (como seria uma EEPROM, por exemplo) não possui entradas de dados, pois não possuem operação de escrita;
- Já vimos no *slide* 15 que o diagrama de blocos de uma ROM possui k bits de entrada (endereço) e n bits de saída (palavra de dados);
- k bits de endereço podem endereçar 2^k palavras, de 0 a $2^k - 1$;

- Circuitos integrados de ROM possuem entradas de habilitação (*enable*) para possibilitar a associação de memórias no mesmo barramento (veremos adiante);
- Consideremos uma ROM 32×8 , ou seja, 32 palavras de 1 byte cada. Há 5 bits de entrada de endereço ($2^5 = 32$), referenciando as palavras armazenadas nas posições de 0 a 31;
- A construção interna desta ROM decodifica os 5 bits de entrada em 32 saídas distintas por meio de um decodificador;

- **Notação:** Em vez de se desenhar várias linhas de entrada na porta lógica, desenha-se uma única linha. Cada possível entrada da porta é representada por uma linha perpendicular seletivamente conectada, sendo que o “X” indica a existência de conexão;



(a) Conventional symbol



(b) Array logic symbol

Figura: Símbolo *array logic* - Fonte: [1].5 - Figura 5.5

ROMs

Introdução

Terminologia

ROMs

RAMs

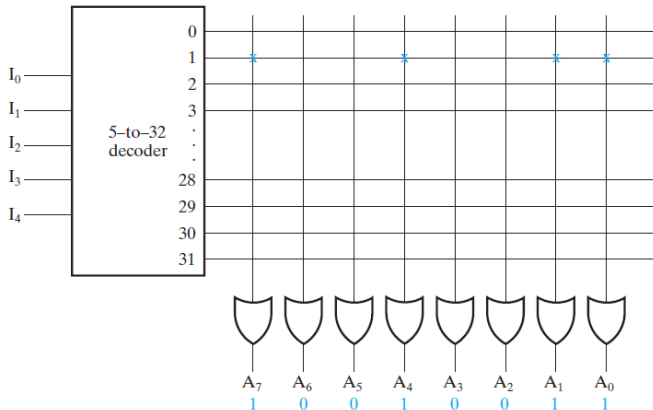


Figura: Lógica interna de uma ROM - Fonte: [1].5 - Figura 5.7 (b)

ROMs

Introdução

Terminologia

ROMs

RAMs

- Observe os “X” azuis na linha 1 da saída do decodificador;
- Cada “X” denota a existência de uma conexão. A ausência de “X” denota a não existência de uma conexão;
- Esta notação de porta lógica é conhecida como símbolo *array logic*;

ROMs

Introdução

Terminologia

ROMs

RAMs

- Cada saída do decodificador do *slide* 33 representa um endereço de memória;
- Cada uma das 32 saídas do decodificador são conectadas por meio de conexões programáveis a cada uma das 8 portas OR;
- Ou seja, cada porta lógica OR possui 32 conexões de entrada possíveis;
- Sendo assim, esta memória possui $32 \times 8 = 256$ conexões programáveis;
- De modo geral, uma ROM $2^k \times n$ possui um decodificador k -para- 2^k e n portas OR;

- Efetuar essas conexões de acordo com o padrão desejado corresponde a **programar** a memória;
- Para a entrada de endereço $00001_2 = 1_{10}$, por exemplo, apenas a saída 1 do decodificador assume valor lógico ALTO, e todas as outras assumem valor BAIXO;
- Desse modo, somente as conexões programadas na linha de saída 1 (quando o endereço de entrada for 00001_2) determinarão a saída de cada porta OR que, em conjunto, correspondem à palavra de saída;
- Pela figura, esse endereço está programado com o conteúdo 10010011_2 ;

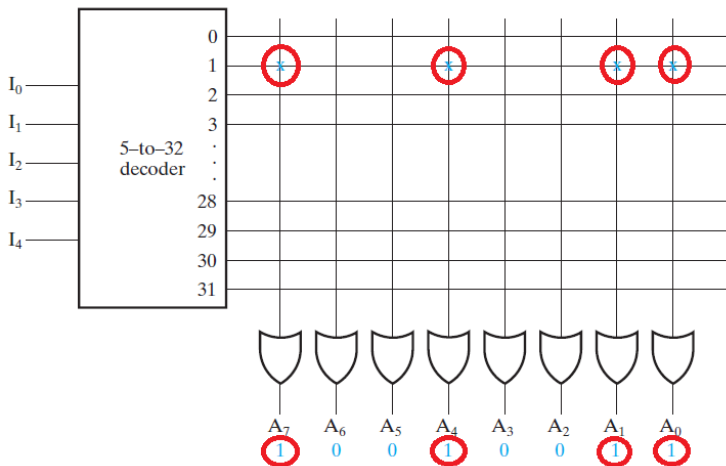
ROMs

Introdução

Terminologia

ROMs

RAMs



- Dependendo da tecnologia de programação de conexões utilizada, a memória apresenta diferentes nomes:
 - **ROM**: programável por máscara. Os dados são escritos no processo de fabricação e não podem ser reprogramados;
 - **PROM (*Programmable ROM*)**: existem fusíveis em cada ponto de conexão. Com o uso de tensões mais altas, eles são "queimados" e o caminho é destruído, sendo um processo irreversível. A gravação dos dados não sai de fábrica, mas o usuário só pode programar uma vez. Também pode ser implementada com antifusíveis, invertendo esta lógica de escrita;

- ■ **EPROM (*Erasable* PROM)**: uma vez programada, os dados podem ser apagados por exposição à luz ultravioleta e depois reprogramados. Possuem uma janela transparente no empacotamento;
- **EEPROM (*Electrically Erasable* PROM)**: pode ser apagada eletricamente, com o envio dos sinais apropriados em certas entradas;
- **Flash**: tipo de EEPROM com múltiplos modos de escrita e apagamento;
- Independentemente da tecnologia (cada uma mais adequada para uma aplicação ou uso), os dados estão armazenados na forma de um padrão de conexões abertas e fechadas nas entradas das portas OR.

RAMs

Introdução

Terminologia

ROMs

RAMs

RAMs

RAMs

Introdução

Terminologia

ROMs

RAMs

- Além de ROMs, computadores também utilizam memórias RAM;
- Lembrando que a sigla RAM é comumente usada para se referir a memórias voláteis de acesso aleatório;
- Esse tipo de memória aceita o armazenamento de novos dados durante a operação por meio de operações de escrita;
- Existem RAMs com capacidade variando de centenas a bilhões de bits;
- Independentemente do tamanho da palavra, o tamanho de uma RAM é usualmente apresentado pelo número total de bytes que ela pode armazenar (ex.: computador com 8GiB de RAM);

RAMs

Introdução

Terminologia

ROMs

RAMs

- A comunicação entre uma memória RAM e seu ambiente (ex.: o computador que a utiliza) acontece por meio de sinais de entrada e saída de dados, sinais de endereço e sinais de controle;
- Os sinais de dados e de endereço já foram vistos no exemplo da ROM, em que os primeiros eram apenas de saída;
- Os sinais de controle determinam a direção da transferência da informação (operação de leitura ou escrita);

RAMs

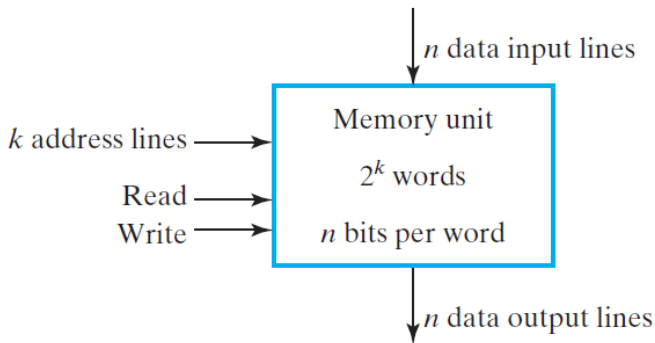


Figura: Diagrama de blocos de uma RAM - Fonte: [1].7 - Figura 7.1

- A entrada de controle *Read* causa a transferência da palavra de dados armazenada na posição relativa ao endereço para fora da memória;
- A entrada de controle *Write* causa o armazenamento da palavra de dados presente na entrada na posição indicada pelo sinal de endereço;

RAMs- Leitura

Introdução

Terminologia

ROMs

RAMs

- Os passos para uma **operação de leitura** são:
 - 1 Aplicar o endereço desejado em binário nas entradas de endereço. Isso selecionará uma certa posição da memória;
 - 2 Ativar a entrada *Read*. A memória irá tomar os bits armazenados na posição indicada pelo endereço e aplicá-los nos sinais de saída de dados, disponibilizando-os para o dispositivo que está utilizando a memória;
- O conteúdo armazenado na palavra selecionada não é alterado pela operação de leitura;

RAMs- Leitura

Introdução

Terminologia

ROMs

RAMs

- Em geral, *chips* de memória RAM provêem os sinais de controle de uma forma diferente da apresentada [1].7;
- Em vez de possuir duas entradas separadas *Read* e *Write*, possuem um único sinal *Read/Write* ou R/\overline{W} para determinar a operação;
- Também apresentam um sinal *Chip Select* (CS) para selecionar o *chip* que será lido ou escrito, permitindo a associação de vários *chips* de memória em um único barramento de endereço (veremos adiante);
 - Quando CS não está ativo, o *chip* em questão não está selecionado, e nenhuma operação é executada;
 - Quando CS está ativo, o sinal R/\overline{W} determina a operação a ser executada;

□ **TABLE 7-1**
Control Inputs to a Memory Chip

Chip Select CS	Read/ $\overline{\text{Write}}$ R/ \overline{W}	Memory Operation
0	×	None
1	0	Write to selected word
1	1	Read from selected word

Figura: Sinais de controle de uma memória RAM - Fonte: [1].7 - Tabela 7.1

RAMs- Leitura

Introdução

Terminologia

ROMs

RAMs

- Uma memória composta por vários *chips* também possui um sinal que habilita toda a memória (o CS é individual para cada *chip*). Este sinal é chamado de *Memory Enable*;
- Quem opera os sinais de controle da RAM é o dispositivo que a está utilizando. No computador, é a CPU que envia esses sinais;
- Vejamos um exemplo de ciclo de leitura com tempo de acesso de 65ns;

RAMs- Leitura

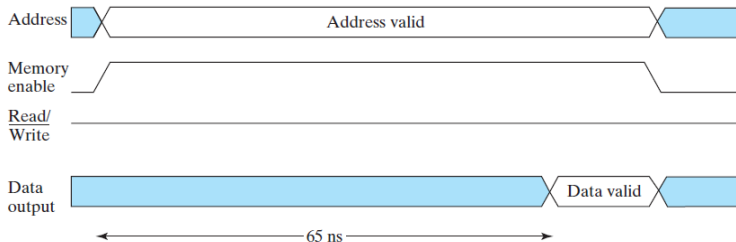


Figura: Ciclo de leitura de uma RAM - Fonte: [1].7 - Figura 7.3 (b) - adaptado

RAMs- Leitura

Introdução

Terminologia

ROMs

RAMs

- A CPU aplica o endereço e seta o *Memory Enable* para ALTO;
- O R/\overline{W} é mantido em ALTO, indicando uma operação de leitura;
- A memória leva 65ns para disponibilizar os dados armazenados no endereço selecionado nos sinais de saída de dados;

RAMs- Leitura

Introdução

Terminologia

ROMs

RAMs

- É importante observar que a memória não emprega o *clock* da CPU;
- O número de ciclos de *clock* que se deve aguardar pela disponibilização do dado é o menor inteiro cujo período total (número de ciclos x período do ciclo) é maior que o tempo de acesso;
- Em um *clock* de 50MHz (período de 20ns), 3 ciclos ainda não são suficientes para o tempo de acesso de 65ns, pois $3 \times 20ns = 60ns$, mas 4 ciclos são suficientes já que $4 \times 20ns = 80ns$ e $80 > 65$;
- Revendo a figura com os ciclos de *clock* da CPU, temos:

RAMs- Leitura

Introdução

Terminologia

ROMs

RAMs

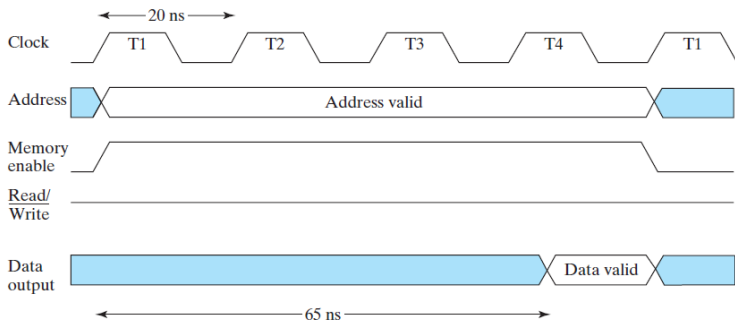


Figura: Ciclo de leitura de uma RAM - Fonte: [1].7 - Figura 7.3 (b)

RAMs- Leitura

Introdução

Terminologia

ROMs

RAMs

- No próximo pulso após T4, a CPU transfere os dados disponibilizados pela memória a um de seus registradores internos;

RAMs- Escrita

Introdução

Terminologia

ROMs

RAMs

- Os passos para uma **operação de escrita** são:
 - 1 Aplicar o endereço desejado em binário nas entradas de endereço. Isso selecionará uma certa posição da memória;
 - 2 Aplicar na entrada de dados os bits da palavra que se deseja armazenar);
 - 3 Ativar a entrada *Write*. Os bits que estavam na entrada serão copiados e armazenados na posição selecionada, substituindo a palavra que estava na mesma posição anteriormente;

RAMs- Escrita

Introdução

Terminologia

ROMs

RAMs

- Tomando agora um exemplo de ciclo de escrita com duração de 75ns:

RAMs- Escrita

Introdução

Terminologia

ROMs

RAMs

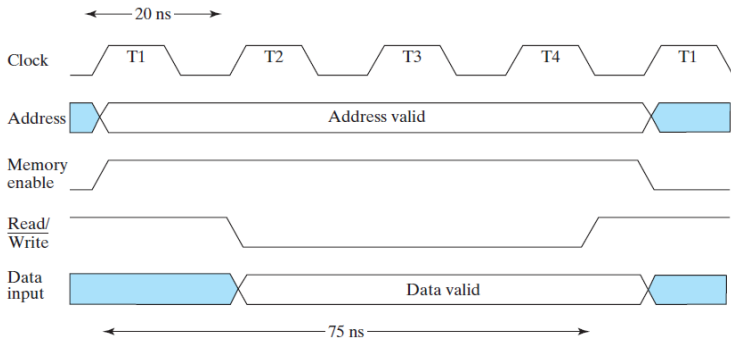


Figura: Ciclo de escrita de uma RAM - Fonte: [1].7 - Figura 7.3 (a)

RAMs- Escrita

Introdução

Terminologia

ROMs

RAMs

- A CPU aplica o endereço e seta o *Memory Enable* para ALTO na borda de subida do pulso T1;
- A palavra a ser armazenada (*Data input*) é colocada na borda de subida do próximo pulso de *clock*, T2;
- Também na borda de subida de T2, o R/\overline{W} é passado para BAIXO, indicando uma operação de escrita. **Isso é feito no ciclo seguinte a T1 para evitar sobrescrever outra palavra da memória: em T2, os sinais de endereço e a seleção da palavra na memória já estão estáveis;**
- Depois de um tempo suficiente para completar a operação, o sinal R/\overline{W} volta para ALTO;
- Os sinais de endereço e dados devem permanecer estáveis por mais um curto período pela mesma razão;

- Dependendo do tipo de célula utilizada em sua construção, a memória apresenta diferentes nomes:
 - **SRAM (*Static* RAM)**: as células de memória são do tipo *flip flops*, tipicamente compostas por 6 a 8 transistores [2].5. O tempo de acesso comum é de 0,5 a 2,5ns, extremamente rápidas mas relativamente caras por bit. Usadas para *cache*;

RAMs- SRAM

Introdução

Terminologia

ROMs

RAMs

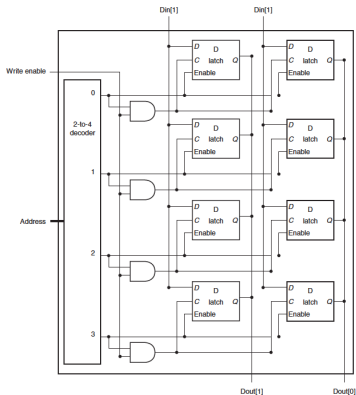


Figura: Estrutura de uma SRAM 4 x 2 - Fonte: [2].A - Figura A.9.3

RAMs- SRAM

Introdução

Terminologia

ROMs

RAMs

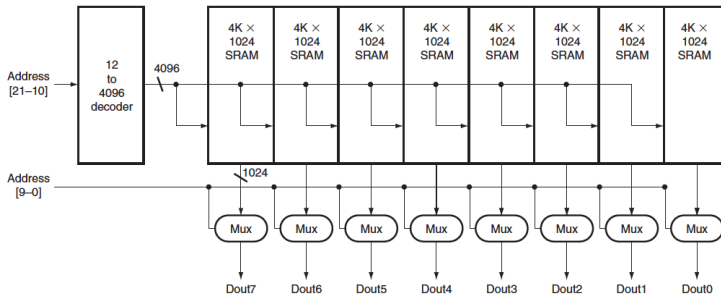


Figura: Organização típica de uma SRAM 4M x 8 como um array de 4K x 1024 arrays - Fonte: [2].A - Figura A.9.4

- ■ **DRAM (*Dynamic* RAM)**: cada bit é armazenado como carga em um capacitor. Um único transistor é usado para acessar a carga (ler/escrever o bit). É mais barata por bit que uma SRAM, mas mais lenta. O tempo de acesso típico é de 50 a 70ns. Usada como memória principal em computadores;
 - São chamadas de dinâmicas pois mesmo mantendo a alimentação elétrica, a carga de cada capacitor é perdida com o tempo. Para preservar os dados, é necessário ocorrer um *refresh* periódico, que consiste em ler o valor de cada palavra e reescrevê-lo na mesma posição a cada poucos milisegundos.

RAMs- DRAM

Introdução

Terminologia

ROMs

RAMs

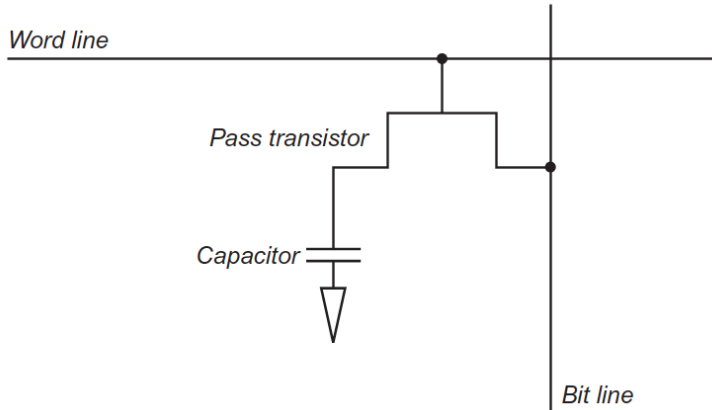


Figura: Célula DRAM de 1 transistor - Fonte: [2].A - Figura A.9.5

RAMs- DRAM

Introdução
Terminologia
ROMs
RAMs

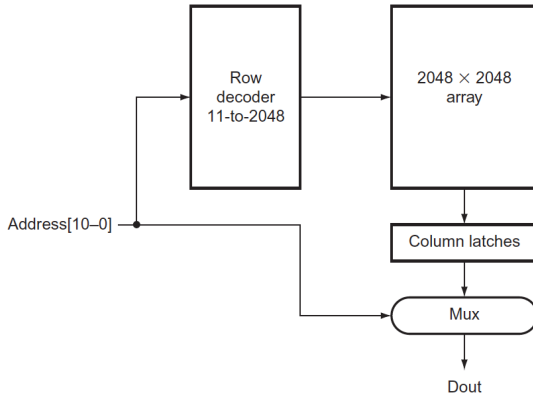


Figura: DRAM de 4M x 1 feita com array 2048 x 2048 - Fonte: [2].A - Figura A.9.6

Obrigado!



Universidade de São Paulo



DEPARTAMENTO DE ENGENHARIA DE
COMPUTAÇÃO E SISTEMAS DIGITAIS

PCS

Referências

Introdução

Terminologia

ROMs

RAMs



M. Mano, C. Kime, and T. Martin.
Logic and Computer Design Fundamentals.
Pearson Education, Incorporated, 2015.



D. Patterson and J. Hennessy.
Computer Organization and Design ARM Edition: The Hardware Software Interface.
The Morgan Kaufmann Series in Computer Architecture and Design. Elsevier Science, 2016.



R. Tocci, N. Widmer, G. Moss, and C. Martins.
Sistemas digitais: princípios e aplicações.
PRENTICE HALL BRASIL, 2007.