

MEMÓRIAS

Um sistema digital é capaz de armazenar uma grande quantidade de dados em períodos de tempo curtos ou longos, sendo esta sua principal vantagem sobre os sistemas analógicos, tornando-os bastante versáteis e adaptáveis a uma grande variedade de situações.

Por exemplo, as instruções armazenadas na memória interna (principal) de um computador digital informam ao computador o que fazer em todas as situações possíveis, de modo que ele possa executar todas as tarefas com o mínimo de intervenção humana.

CONCEITOS BÁSICOS - TERMINOLOGIA

Computador: É todo dispositivo capaz de processar dados e controlar processos externos, automaticamente, de acordo com instruções sequenciais fornecidas previamente.

Dados: Podem ser números, letras, palavras, símbolos gráficos quaisquer ou ainda, sinais elétricos que traduzem a ocorrência de eventos físicos do seu mundo exterior. Processar dados significa manipular ou tratar esses dados de modo a alcançar um resultado desejado.

Instrução: Denominamos *instrução* a cada ordem fornecida ao computador para uma operação específica com os dados.

Memória: É todo dispositivo capaz de armazenar informações. Em uma memória, toda a informação é armazenada na forma binária.

Célula de memória: É um dispositivo ou circuito elétrico capaz de armazenar um único bit (0 ou 1). Como célula de memória podemos citar o flip-flop, um capacitor, um ponto magnético em fita ou disco, etc.

Palavra de memória: É um conjunto de bits que representa instruções ou dados. Por exemplo, um registrador composto por 8 flip-flops pode ser considerado uma memória com capacidade de armazenar uma palavra de 8 bits.

Byte: É o conjunto de 8 bits.

Endereço: É um número que identifica a posição de uma palavra na memória, sendo expresso sempre em número binário, mas, em alguns casos por conveniência pode ser expresso em octal e hexadecimal.

A figura a seguir mostra a tabela de uma pequena memória composta por 8 palavras, cada uma delas com um endereço específico.

Endereço

000	Palavra 0
001	Palavra 1
010	Palavra 2
011	Palavra 3
100	Palavra 4
101	Palavra 5
110	Palavra 6
111	Palavra 7

Operação de leitura: É a operação em que uma palavra binária armazenada em posição específica na memória (endereço) é identificada e transferida para outro dispositivo qualquer do sistema. Essa operação muitas vezes é denominada *busca*.

Operação de escrita: É a operação na qual uma palavra é colocada em determinada posição da memória. Essa operação muitas vezes é denominada *armazenamento*.

CONSIDERAÇÕES SOBRE MEMÓRIAS PRINCIPAL E EXTERNA

O flip-flop é um circuito eletrônico capaz de armazenar 1 bit de informação.

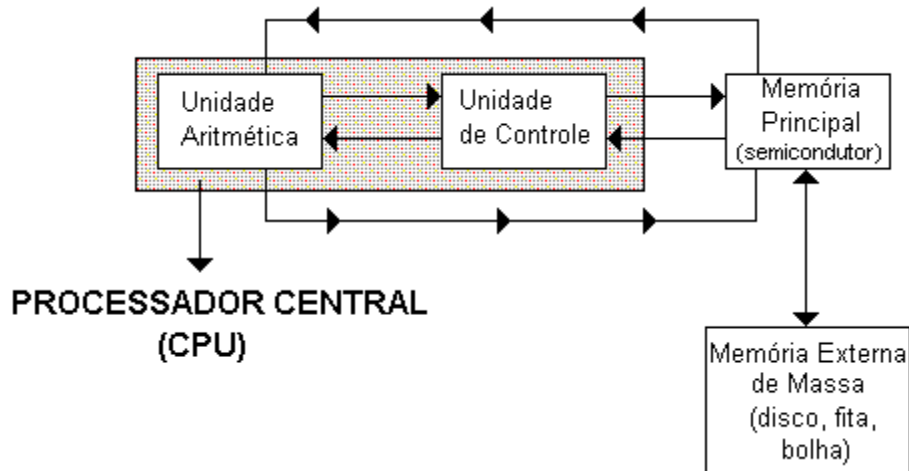
Grupos de flip-flops denominados registradores são capazes de armazenar *informação estruturada*, composta por dados ou instruções e tais *informações* podem ser recebidas ou transferidas para outros dispositivos de armazenamento.

Os registradores são elementos de memória de alta velocidade empregados no processo de armazenamento de informação durante a execução de instruções pela unidade de controle da máquina, havendo uma constante movimentação de informações entre os registradores e demais dispositivos componentes do sistema.

Avanços tecnológicos tornaram possível a colocação de um número extremamente elevado de flip-flops dentro de um único *chip*, aliando-se a isso uma velocidade de acesso cada vez maior a um custo mais acessível.

As informações em sistemas digitais podem ser armazenadas em memórias construídas com semicondutores segundo a tecnologia bipolar ou unipolar (MOS) ou ainda, memórias que utilizam capacitores como elementos básicos de armazenamento em substituição aos flip-flops.

As memórias a semicondutor são usadas como memória principal de um computador, onde é exigida uma velocidade de operação bastante alta. A memória principal é a parte integrante do próprio computador e sem ela o computador simplesmente não funciona. Pode ser também denominada memória residente ou memória interna, estando em constante comunicação com o processador enquanto as instruções básicas estiverem sendo executadas. A figura abaixo mostra o diagrama simplificado de um computador.



No entanto, devido ao alto custo por bit, as memórias a semicondutor são empregadas apenas como memória principal.

A memória externa é dispensável para o funcionamento do computador, entretanto, amplia sobremaneira a capacidade e a flexibilidade do mesmo. Mais do que isso, torna-se imprescindível sempre que tornar-se necessário armazenar dados e programas para consultas futuras.

As fitas magnéticas e os discos magnéticos estão entre os dispositivos de memória de massa mais populares, apresentando um custo por bit bem menor do que os dispositivos utilizados na implementação da memória principal.

O mais novo componente da família dos dispositivos de memória de massa é constituído pelos que empregam a tecnologia de *bolhas magnéticas (MBM)*, dispositivos a semicondutor que utilizam o princípio do magnetismo para armazenar milhões de bits em um único chip. As MBM são muito lentas, não podendo, portanto, ser utilizadas na implementação de dispositivos de memória principal.

CAPACIDADE DE UMA MEMÓRIA

É muito importante especificar quantos bits podem ser armazenados em uma determinada memória. Tomemos como exemplo uma memória que possa armazenar 4.096 palavras de 20 bits. Isto representa uma capacidade total de armazenamento de 81.920 bits (4.096 x 20), onde 4.096 é o número de palavras e 20 é a quantidade de bits por palavra.

$$\begin{array}{ccc}
 & \text{Quantidade de} & \\
 & \text{bits por palavra} & \\
 & \downarrow & \\
 \text{4.096} & \times & \text{20} = \text{81.920} \\
 \uparrow & & \uparrow \\
 \text{Número de} & & \text{Quantidade total} \\
 \text{palavras} & & \text{de bits}
 \end{array}$$

Utiliza-se comumente representar o número de palavras da memória como múltiplo de 1.024, sendo comum a designação 1K (1 kilo) para representar 1.024 bits, que é igual a 2^{10} .

Por exemplo, uma memória que tenha uma capacidade de armazenamento de 8K x 20 é na verdade uma memória de 8.192 x 20. Memórias de grande capacidade de armazenamento utiliza a designação 1M (1 mega), que representa 2^{20} que é igual a 1.048.576 bits.

Dessa forma uma memória com capacidade de 2M x 8, possui uma capacidade de 2.097.152 x 8.

Vejamos alguns exemplos:

1) Um chip de memória é especificado tendo a capacidade de 4K x 8. Quantas palavras podem ser armazenadas nesse chip? Qual é o tamanho da palavra? Quantos bits no total esse chip pode armazenar?

Solução:

$4K = 4 \times 1.024 = 4.096$ palavras

Cada palavra tem 8 bits ou 1 byte

O número total de bits é 32.768 (4.096×8)

2) Qual das memórias armazena mais bits?

a) 2M x 8

b) 1M x 16

Solução:

a) $2 \times 1.048.576 \times 8 = 16.777.216$

b) $1 \times 1.048.576 \times 16 = 16.777.216$

Portanto, as duas memórias tem a mesma capacidade de armazenamento de bits.

CONCEITOS SOBRE DISPOSITIVOS DE MEMÓRIAS

Tempo de acesso: O tempo de acesso mede a velocidade de acesso à memória. É a quantidade de tempo necessária à efetivação de uma operação de leitura ou, mais especificamente, é o tempo decorrido entre o momento da recepção pela memória de um novo endereço, e o instante em que a informação daquele endereço fica disponível.

Memória volátil: Qualquer tipo de memória que necessite de energia elétrica para reter informações armazenadas é denominada memória volátil. Quando a energia elétrica for interrompida toda a informação armazenada será perdida.

Muitas memórias a semicondutor são voláteis, enquanto que todas as memórias magnéticas são não voláteis.

Memória de acesso randômico (RAM): Memória onde a localização física real de uma palavra de memória não tem efeito sobre o tempo que se leva para ler ou escrever nesta posição, isto é, o tempo de acesso é constante para qualquer endereço da memória. A grande maioria das memórias e semicondutor e todas as de núcleo magnético são randômicas.

Memória de acesso sequencial (SAM): Memória onde o tempo de acesso não é constante, mas depende do endereço. Para encontrar determinada palavra, passa-se por todos os endereços situados entre aquele onde se realizou o último acesso e o objeto do acesso atual. Isto produz tempos de acesso bem maiores do que os dispositivos de acesso randômico.

Exemplos de SAM: fitas magnéticas, discos magnéticos e as memórias de bolhas magnéticas. Para melhor ilustrar a diferença entre uma SAM e uma RAM, considere a situação de uma fita cassete com 60 minutos de música gravada. Para ouvir determinada música deve-se avançar ou recuar a fita até encontrar a música desejada, e com isto, perde-se um tempo considerável dependendo de onde se encontra a música desejada.

Isto representa então uma analogia com a memória SAM. No caso da memória RAM a analogia pode ser feita com uma máquina automática de música, onde apertando-se um botão consegue-se ouvir a música desejada.

Memória de Leitura/Escrita (RWM): Qualquer memória que possa ser lida ou escrita com facilidade.

Memória de leitura (ROM): São memórias a semicondutor onde a taxa de operações de leitura é infinitamente maior do que a escrita. Tecnicamente uma ROM pode ser gravada ou programada apenas uma vez, o que na maioria das vezes é feita na fábrica. Depois disso, a informação somente poderá ser lida.

Dispositivos de memória estática: Nessas memórias as informações armazenadas permanecerão armazenadas enquanto houver energia elétrica aplicada à memória, sem que haja necessidade da informação ser rescrita periodicamente na memória.

Dispositivos de memória dinâmica: Nessas memórias as informações armazenadas não permanecerão armazenadas, mesmo com a presença da energia elétrica necessária para alimentar o circuito, a não ser que as informações sejam rescritas periodicamente na memória com determinada frequência. Esta operação é denominada *refresh* (recarga).

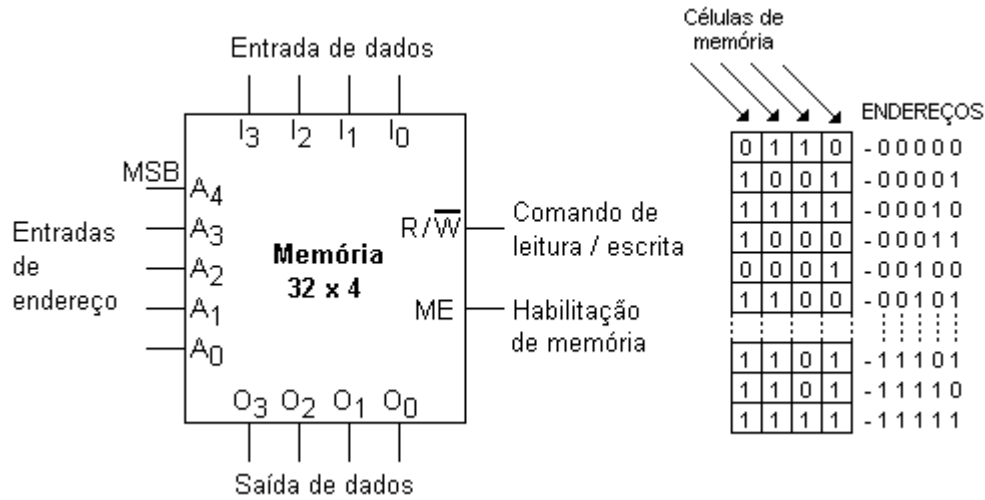
OPERAÇÕES BÁSICAS DA MEMÓRIA

As memórias de uma forma geral apresentam operações básicas, apesar dos diversos tipos de implementação, cujas operações básicas são:

- a) selecionar o endereço que está sendo acessado para leitura ou escrita;
- b) selecionar a operação a ser realizada, leitura ou escrita;
- c) fornecer os dados de entrada para operação de escrita;
- d) manter estáveis as informações de saída da memória. resultantes de uma operação de escrita, durante um tempo determinado;

e) habilitação ou desabilitação da memória, para fazê-la responder ou não ao endereço na entrada e ao comando leitura/escrita.

A figura a seguir mostra como exemplo, o diagrama simplificado de uma memória 32 x 4 (32 palavras de 4 bits), isto é, cada palavra tem o tamanho de 4 bits.



Como o tamanho da palavra é de 4 bits, existe nesta memória 4 linhas de entrada de dados e 4 linhas de saída de dados.

Durante a operação de escrita, os dados a serem armazenados na memória devem ser colocados nas linhas de entrada de dados (I₀ a I₃) e durante a operação de leitura, a palavra lida aparece nas linhas de saída de dados (O₀ a O₃).

A memória mostrada no exemplo acima possui 32 posições diferentes de armazenamento e, portanto, 32 endereços diferentes, começando por 00000 e terminando em 11111 (0 a 31 decimal).

Desta forma essa memória deve ter 5 entradas de endereçamento ($2^5 = 32$). Em geral são necessárias N linhas de entrada de endereço para uma memória com capacidade de 2^N palavras.

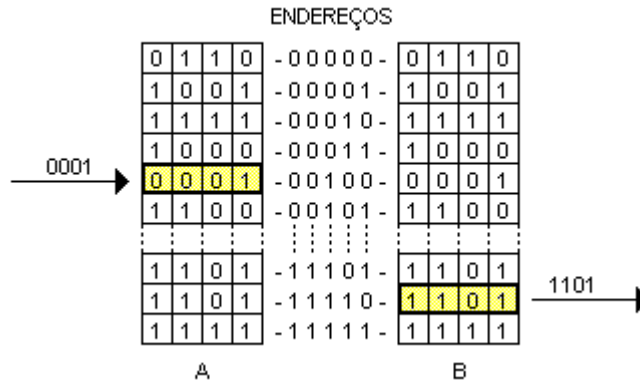
Voltando ao exemplo acima, observa-se que cada posição possui 4 células de memória que armazenam 0s e 1s, formando assim uma palavra em determinadas posições. Dessa forma, no endereço 00001 está armazenada a palavra 1001; no endereço 11110 está armazenada a palavra 0111 e assim por diante.

O comando de leitura/escrita determina qual das operações a memória deverá executar. Alguns sistemas utilizam linhas separadas para leitura e escrita. Quando se usa uma única linha temos as seguintes condições:

$$\text{LEITURA: } R / \overline{W} = 1$$

$$\text{ESCRITA: } R / \overline{W} = 0$$

A figura a seguir ilustra o processo de leitura e escrita na memória 32 x 4, onde em A temos a escrita de uma palavra de dados 0001 na posição da memória cujo endereço é 00100, enquanto que em B temos a leitura de uma palavra de dados 1101 na posição de memória, cujo endereço é 11110.



Muitos sistemas de memória tem meios de desabilitar toda a memória ou parte dela, de tal forma a fazer que um determinado conjunto de posições desabilitadas não responda a nenhum tipo de operação.

No nosso exemplo a memória possui uma entrada que permite a habilitação e desabilitação, podendo ser ativa em nível alto ou em nível baixo.

No nosso exemplo a habilitação ocorre em nível alto. Um nível baixo aplicado a uma memória desabilita-a por completo.

Esse procedimento é usado quando diversos módulos de memória são combinados para formar uma memória maior.

Vejamos alguns exemplos:

a) Descreva as condições de cada uma das linhas de entrada e saída, quando precisamos ler o conteúdo do endereço 11110.

Solução:

- Entradas de endereço: 11110
- Entradas de dados: não utilizadas
- R / W' : nível alto
- Habilitação da memória : nível alto
- Saídas de dados: 1101

b) Descreva as condições de cada uma das linhas de entrada e saída, quando precisarmos escrever a palavra 1001 no endereço 10001.

Solução:

- Entradas de endereço: 10001
- Entradas de dados: 1001

R /W' : nível baixo
Habilitação da memória: nível alto
Saídas de dados: não utilizadas (normalmente em alta impedância)

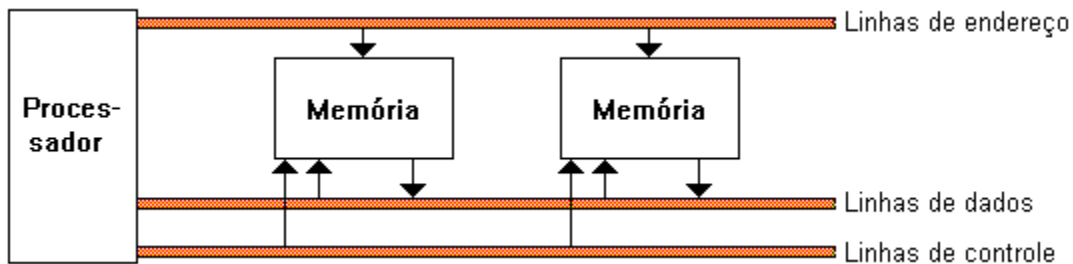
- c) Uma memória tem uma capacidade de 8K x 4. Pergunta-se:
- Quantas linhas para entrada e saída de dados esta memória deve ter?
 - Quantas linhas de endereço deve ter?
 - Qual é a sua capacidade total em bytes?

Solução:

- 4 linhas de entrada e 4 linhas de saída, pois o tamanho da palavra é de 4 bits.
- A memória armazena 8.192 palavras, devendo então existir 8.192 endereços de memória. Como $8.192 = 2^{13}$, a mesma precisa de 13 bits para especificar qualquer um dos seus endereços.
- Cada byte corresponde a 8 bits. Logo, tem 4.096 bytes de capacidade.

CONEXÕES DA MEMÓRIA COM O PROCESSADOR

Geralmente a memória principal (interna) de um computador é formada por CIs (chips) de RAMs e de ROMs, cujas interfaces com o processador são realizadas por três grupos de sinais ou barramentos conforme ilustra a figura abaixo:



Observa-se que na figura acima estão distintos os barramentos de endereço, de dados e de controle, sendo que cada um desses barramentos é constituído de várias linhas.

O número de linhas varia de computador para computador.

Os três barramentos permitem então que o processador possa ler e escrever dados na memória.

Quando um computador está executando um programa, o processador busca constantemente informações das posições de memórias que contém as instruções representando as operações a serem realizadas.

O processador deve também escrever (armazenar) dados em posições de memória, conforme solicitado pelas instruções.

As operações de leitura e escrita do processador na memória devem seguir os seguintes passos:

Operação de escrita:

a) O processador gera o endereço da posição da memória onde o dado deve ser armazenado. Ela coloca este endereço nas linhas do barramento de endereços.

b) O processador coloca no barramento de dados o dado a ser armazenado.

c) O processador ativa as linhas correspondentes aos sinais de controle apropriados para a escrita.

d) As memórias decodificam o endereço para determinar qual posição está sendo selecionada para a posição de armazenamento.

e) O dado no barramento de dados é transferido para a posição de memória selecionada.

Operação de leitura:

a) O processador gera o endereço da posição de memória de onde o dado deve ser retirado; o endereço é então colocado no barramento (linhas) de endereço.

b) O processador ativa as linhas correspondentes aos sinais de controle apropriados para a leitura.

c) As memórias decodificam o endereço para determinar qual posição está sendo selecionada para a leitura.

d) As memórias colocam no barramento de dados o dado que está armazenado na posição selecionada, de onde ele será transferido para o processador.

Desta forma, quanto aos barramentos de um sistema de computador podemos resumir:

BARRAMENTO DE ENDEREÇOS: É um barramento unidirecional, que leva o endereço que aparece na saída do processador para a memória.

BARRAMENTO DE DADOS: É um barramento bidirecional por onde trafegam os dados, tanto no sentido da memória para o processador como do processador para a memória. Neste barramento trafegam tanto instruções básicas como dados.

BARRAMENTO DE CONTROLE: É um barramento bidirecional por onde trafegam os sinais de controle, principalmente no sentido do processador para a memória.

ROM (READ-ONLY MEMORIES)

ROM (Read-Only Memories), que traduzindo para o português significa memória de apenas leitura, é uma memória a semicondutor destinada a armazenar informações em caráter permanente e que em raríssimos casos são mudadas.

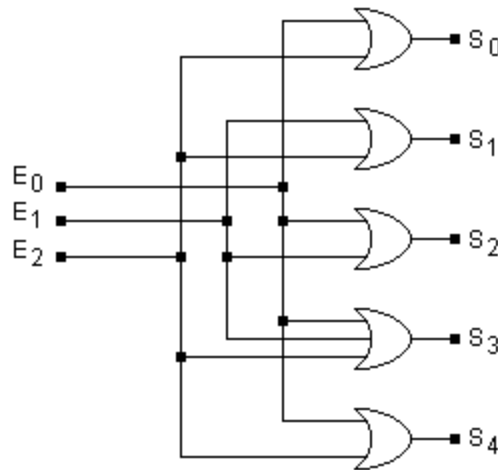
Durante sua operação normal nenhum dado poderá ser escrito na ROM, sendo utilizada então apenas para leitura de dados que estiverem armazenados.

Para alguns tipos de ROM os dados são gravados durante o processo de fabricação enquanto que, para outros, os dados são gravados eletricamente.

O processo de gravação de dados em uma ROM é denominado programação ou queima. Algumas ROMs não podem mais ter seus dados alterados, enquanto que outras podem ter seus dados apagados e regravados.

As ROMs tem características não voláteis e por isso são empregadas para guardar dados que não mudarão durante a operação de um sistema, uma vez que, após cessada a alimentação elétrica os dados não se perdem.

Uma ROM pode ser implementada, conforme ilustra a figura abaixo:



ENDEREÇOS			PALAVRAS				
E ₀	E ₁	E ₂	S ₀	S ₁	S ₂	S ₃	S ₄
0	0	1	1	1	0	1	1
0	1	0	0	1	1	1	0
1	0	0	1	0	1	1	1

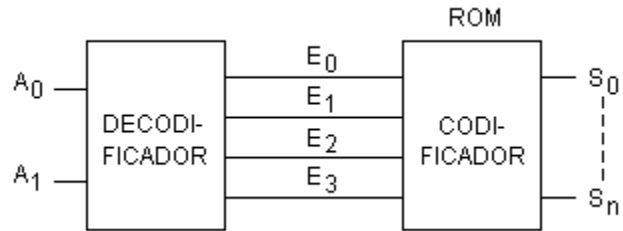
Na realidade, trata-se de um codificador onde E₀ a E₂ são as entradas.

Essas entradas combinadas equivaleriam ao endereço e, nas saídas teríamos as palavras S₀ a S₄.

Então para o circuito apresentado, o endereço 100 por exemplo, equivale a palavra 10111.

Convém lembrar que no codificador somente uma das entradas pode ser levada a nível lógico 1 de cada vez, o que torna muito limitada uma ROM com codificadores.

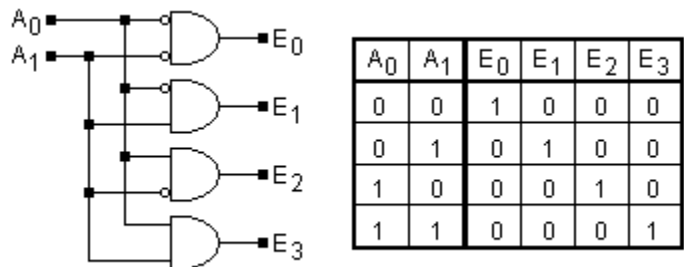
Utilizando um decodificador e um codificador conforme mostra a figura abaixo, podemos através das combinações das entradas estabelecer uma saída, onde neste caso, as entradas poderão estar todas submetidas a nível 1.



A vantagem desse circuito é que a capacidade de armazenamento é sensivelmente aumentada com um número de linhas de endereçamento mais reduzido.

Para n linhas de endereçamento teremos 2^n entradas no codificador e conseqüentemente 2^n palavras.

O decodificador implementado e tabela correspondente são mostrados abaixo:

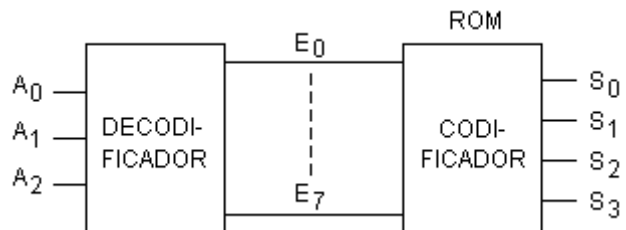


Com base no decodificador e codificador, poderemos projetar uma ROM.

Supondo que queiramos projetar uma ROM que atenda as exigências da tabela mostrada abaixo:

ENDEREÇOS			PALAVRAS			
A_0	A_1	A_2	S_0	S_1	S_2	S_3
0	0	0	1	0	1	0
0	0	1	1	1	0	0
0	1	0	0	1	0	0
0	1	1	0	0	0	1
1	0	0	0	1	1	0
1	0	1	0	0	1	1
1	1	0	1	1	1	1
1	1	1	0	1	1	1

Teremos então o diagrama mostrado a seguir:



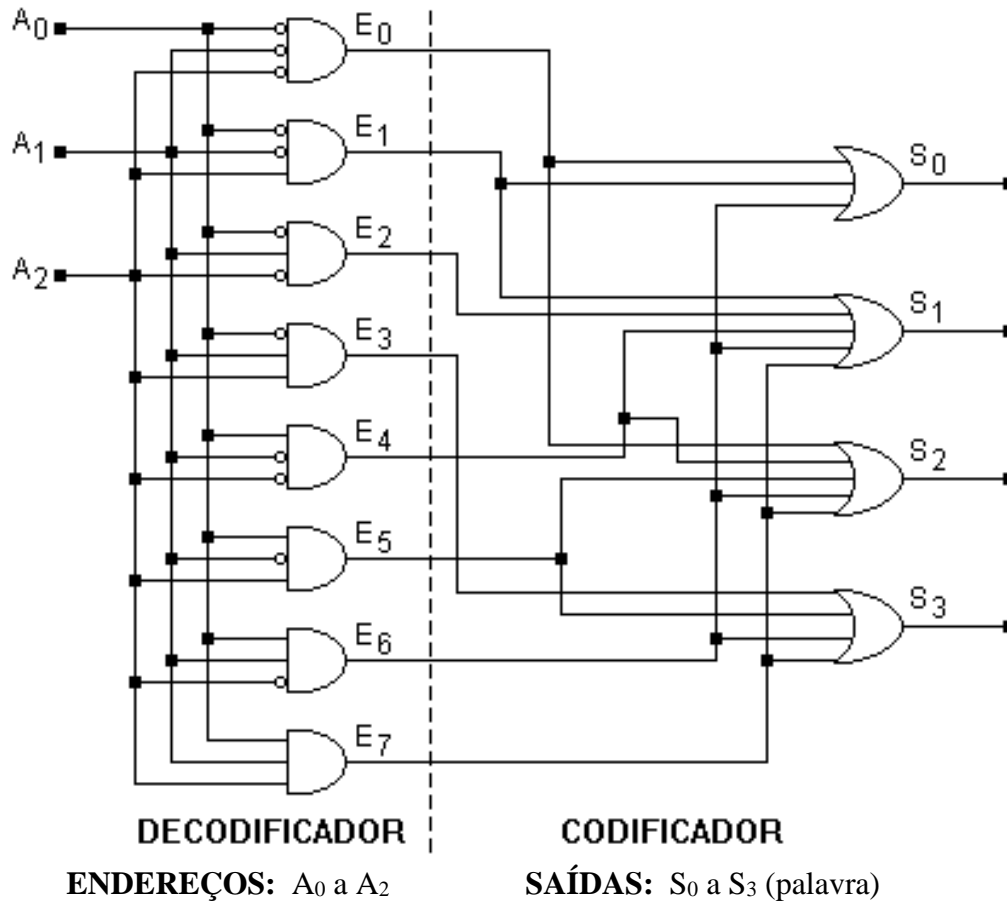
No decodificador teremos 8 saídas (E_0 a E_7), pois o mesmo possui 3 possibilidades de entrada. A tabela correspondente ao decodificador é mostrada abaixo:

A_0	A_1	A_2	E_0	E_1	E_2	E_3	E_4	E_5	E_6	E_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

A tabela correspondente ao codificador é mostrada a seguir:

E_0	E_1	E_2	E_3	E_4	E_5	E_6	E_7	S_0	S_1	S_2	S_3
1	0	0	0	0	0	0	0	1	0	1	0
0	1	0	0	0	0	0	0	1	1	0	0
0	0	1	0	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	0	0	1	1	0
0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	0	1	0	1	1	1	1
0	0	0	0	0	0	0	1	0	1	1	1

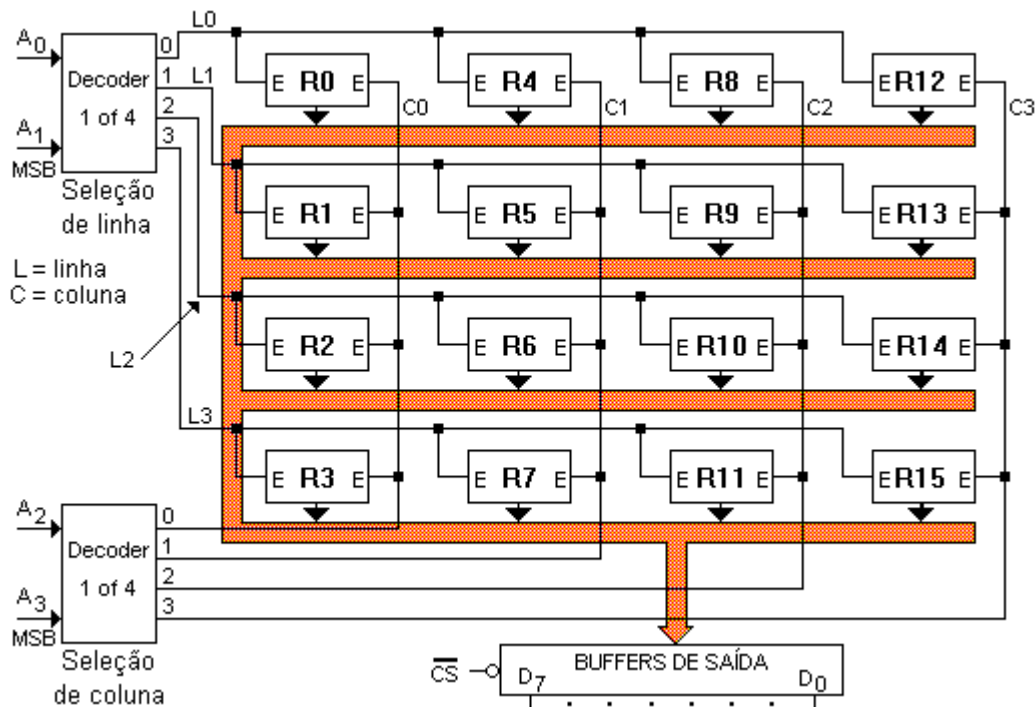
A ROM implementada é mostrada a seguir:



Analisando o circuito implementado, podemos observar que as condições impostas para o projeto estão totalmente satisfeitas.

ARQUITETURA DE UMA ROM

A figura a seguir mostra a arquitetura interna de uma ROM 16 x 8, onde verifica-se a existência de 2 decodificadores: 1 para seleção de linha e 1 para seleção de coluna:



Observa-se no diagrama acima a existência de 16 registradores (R0 a R15) que armazenam todas as informações que foram programadas na ROM.

Cada registrador tem um número de células de memória igual ao tamanho da palavra, que no caso, armazena uma palavra de 8 bits, além de 2 entradas para habilitação (E), ativas em nível alto, sendo uma interligada à linha e outra interligada à coluna.

Os registradores são arranjados na forma de uma matriz quadrada (muito comum na maioria dos chips), identificáveis através das linhas e colunas por eles ocupadas na matriz.

Por exemplo, o registrador 7 está na linha 3, coluna 1; o registrador 11 está na linha 3, coluna 2 e assim por diante.

O código de endereço aplicado, $A_3A_2A_1A_0$, determina qual dos registradores da matriz deve ser habilitado.

Vejamos alguns exemplos:

a) Qual dos registradores será habilitado para uma entrada de endereço 1110?

Solução:

$$A_3A_2A_1A_0 = 1110$$

$$A_3A_2 = 11 \implies \text{coluna 3}$$

$$A_1A_0 = 10 \implies \text{linha 2}$$

Teremos então o registrador 14 devidamente habilitado

b) Qual é o endereço que habilitará o registrador 6?

Solução:

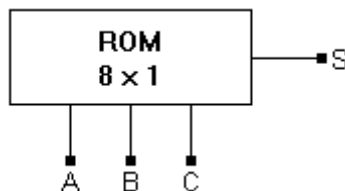
As entradas de habilitação deste registrador são conectadas aos sinais da linha 2 e coluna 1 respectivamente.

Logo, $A_3A_2A_1A_0 = 0110$

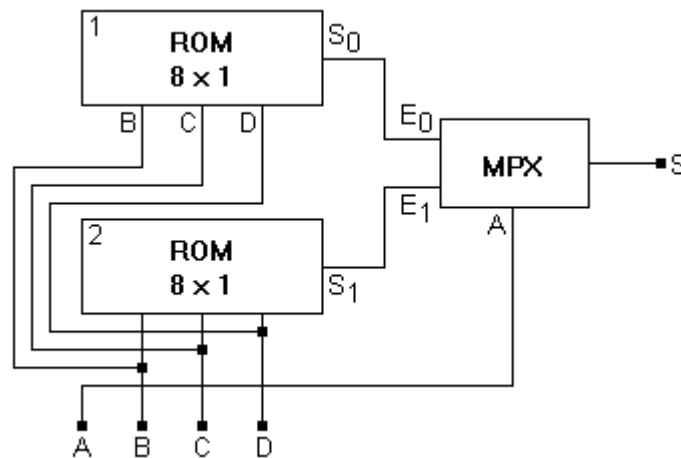
O registrador habilitado pelas entradas de endereço deverá colocar seus dados no barramento de dados, que serão entregues ao *buffer de saída* desde que CS' esteja em nível baixo, caso contrário as saídas do buffer estarão em alta impedância e as linhas D_0 a D_7 estarão em flutuação.

AMPLIAÇÃO DA CAPACIDADE DE UMA ROM

Consideremos uma ROM organizada na forma 8×1 , conforme ilustra a figura abaixo:



Se dispusermos de 2 ROMs com capacidade 8×1 , poderemos ampliar sua capacidade para 16×1 , conforme ilustra a figura abaixo:



As entradas de endereço ABCD dividem-se em duas partes: a mais significativa, no caso A, é aplicada ao multiplex, enquanto BCD (menos significativa) é aplicada à ROM.

De acordo com o valor assumido por A, o multiplex selecionará qual saída, S_0 ou S_1 será conectada às entradas E_0 ou E_1 do multiplex.

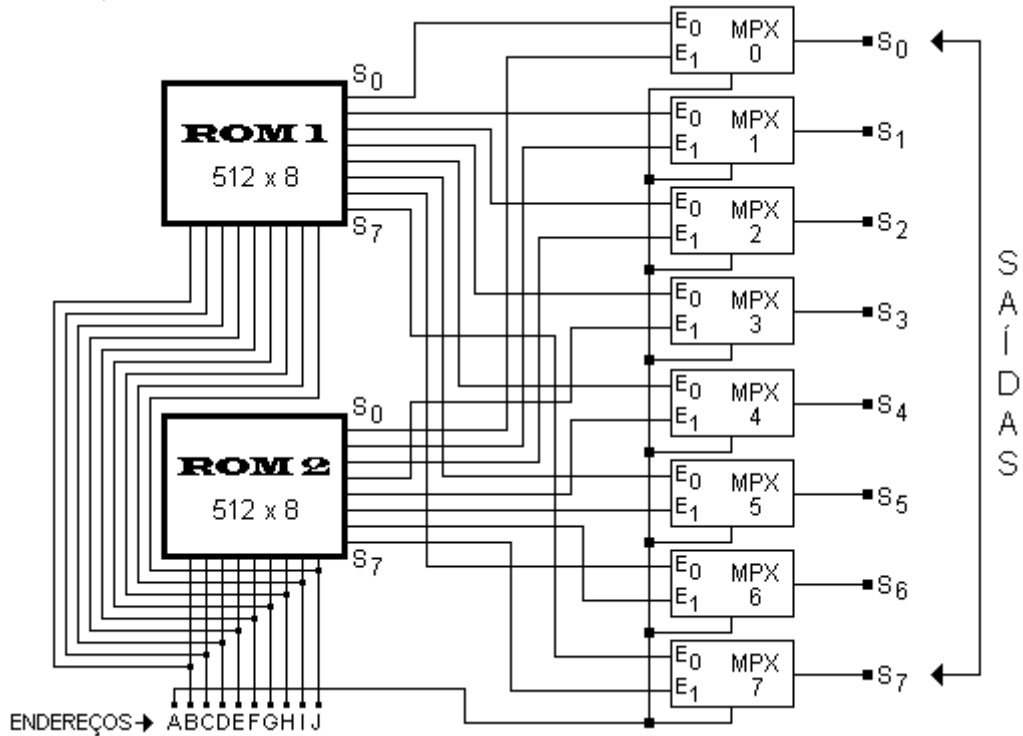
O exemplo acima mostra que não existe grandes dificuldades na ampliação de uma ROM.

Vejamos outro exemplo mais complicado da ampliação da capacidade de uma ROM, a partir de 2 ROMs organizadas na forma 512×8 , com 4096 bits cada.

Com isto será possível ampliar sua capacidade de tal forma a organizá-la na forma 1024 x 8, totalizando 8.192 bits.

Devemos então dispor de 8 multiplex, uma vez que, o tamanho da palavra para essa ROM é de 8 bits.

Desta forma, teremos:



Os endereços de A até J admitem 1024 possibilidades (2^{10}).

A é o endereço mais significativo, conectado aos MPX (0 a 7), que permitirá obter na saída as localidades correspondentes às ROMs 1 ou 2.

As entradas E₀ de cada multiplex estão ligadas na ROM 1, enquanto que as entradas E₁ estão ligadas na ROM 2.

A ROM 1 abrange as localidades 0 a 511, que correspondem aos endereços 0000000000 a 0111111111, enquanto que a ROM 2 abrange as localidades 512 a 1023, que correspondem aos endereços 1000000000 a 1111111111.

Quando A = 0, teremos nas saídas dos multiplex os valores da ROM 1, enquanto que quando A = 1, nas mesmas saídas teremos os valores da ROM 2.

Para o endereço 0100011001 por exemplo, teremos a localidade 281, correspondente a ROM 1; para o endereço 1000101011 teremos a localidade 555, correspondente a ROM 2.

TIPOS DE ROM

MROM - ROM PROGRAMADA POR MÁSCARA:

A ROM programada por máscara tem suas posições de memória gravadas pelo fabricante de acordo com as especificações do cliente.

Uma máscara (negativo fotográfico) é utilizada para realizar as conexões elétricas do chip; para cada conjunto de informações utiliza-se um tipo de máscara.

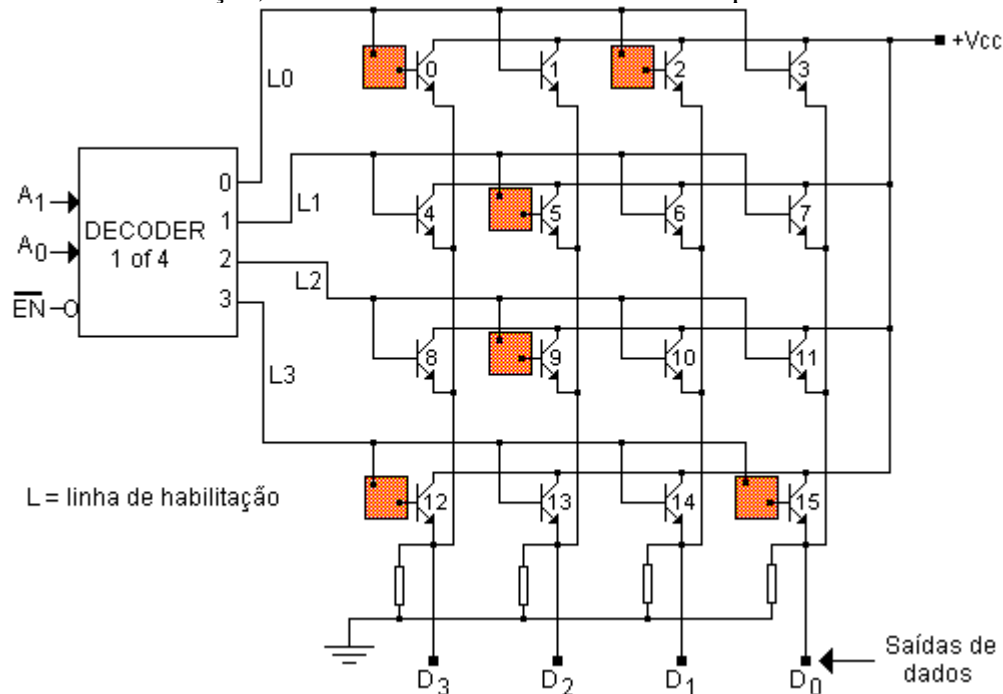
Vantagens: alternativa mais econômica, desde que sejam produzidas em larga escala.

Desvantagens: não podem ser apagadas e reprogramadas caso ocorra a mudança de um determinado projeto que exija modificações nos dados armazenados, pois, neste caso, a ROM com dados antigos não pode ser reaproveitada.

A figura abaixo mostra a estrutura interna de uma MROM de pequena capacidade, constituída de 16 células arranjadas em 4 linhas e 4 colunas de células, construídas com transistores bipolares.

Dessa forma, cada célula é constituída por um transistor bipolar. Cada linha constitui um registrador de 4 bits. Uma conexão aberta na base do transistor armazena um bit “0” enquanto que, uma conexão da base à saída do decodificador armazena um bit “1”.

A condição de cada uma das conexões da base é controlada através de uma máscara fotográfica durante o processo de fabricação, de acordo com os dados fornecidos pelo cliente.



Um decodificador é utilizado na decodificação das entradas de endereço A₁A₀ para selecionar as linhas correspondentes aos registradores. A saída do decodificador é ativa em nível alto, fornecendo o sinal de seleção para a linha correspondente ao endereço selecionado, desde que a entrada EN' esteja em

nível baixo. Caso EN' esteja em nível alto, todas as saídas do decodificador estarão inativas, não havendo qualquer tensão em suas bases.

A tabela abaixo ilustra melhor essa situação:

ENDEREÇOS			SAÍDAS			
A ₁	A ₀	EN'	D ₃	D ₂	D ₁	D ₀
0	0	0	0	1	0	1
0	1	0	1	0	1	1
1	0	0	1	0	1	1
1	1	0	0	1	1	0

Além da tecnologia bipolar pode-se ainda utilizar a tecnologia CMOS ou NMOS para fabricar MROMs.

Dentre as MROMs mais populares pode-se citar:
 74187 - organizada como 256 x 4, tecnologia bipolar
 7488A - organizada como 32 x 8, tecnologia bipolar

Uma memória MROM muito popular, fabricada segundo a tecnologia NMOS, é a TMS47256 / TMS47C256, organizada na forma 32K x 8.

PROMs - ROMs PROGRAMÁVEIS:

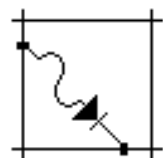
Em virtude do alto custo das MROMs, a não ser que sejam utilizadas em aplicações que envolvam grande produção de chips, foram desenvolvidas para aplicações mais modestas ROMs que podem ser gravadas pelo usuário.

Esse tipo de ROM é conhecido como PROM FUSE, isto é, uma ROM programável (do inglês *PROGRAMMABLE READ ONLY MEMORY FUSE*).

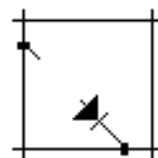
Uma forma bastante simples de implementar uma ROM programável é através de diodos.

Cada célula de memória é constituída por um diodo, após o qual é conectado um fusível.

Se quisermos que determinado ponto não seja um diodo, basta fazer circular por esse ponto uma corrente suficiente para romper o fusível correspondente, conforme ilustra a figura abaixo:



Célula antes da queima do fusível



Célula após a queima do fusível

Pode-se dessa forma gravar as palavras que se deseja armazenar. No entanto, depois de gravada uma palavra, não mais poderá ser alterada ou desgravada. É necessário antes de gravar esse tipo de ROM fazer um mapeamento, localizando os pontos para gravação, pois um eventual erro irá inutilizar o chip.

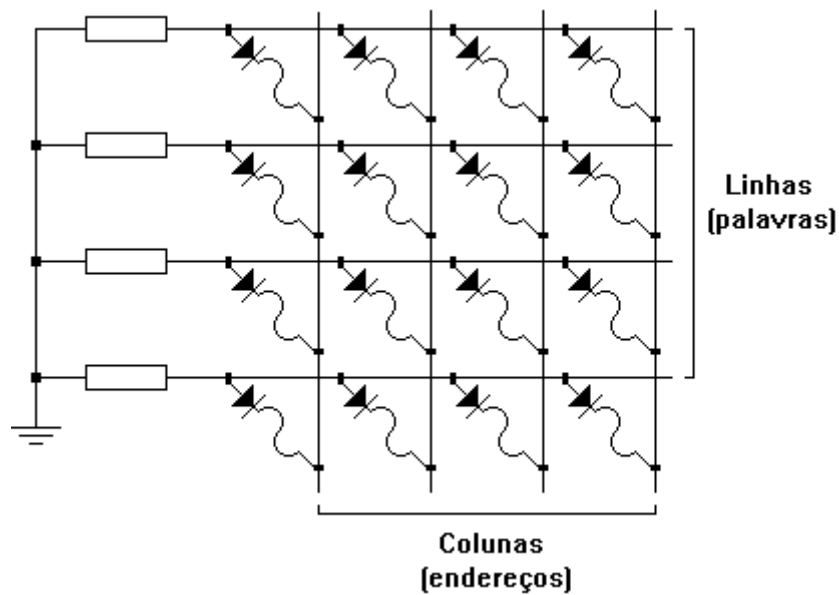
Se após gravada, houver a necessidade de modificar os dados armazenados, isto não mais será possível, havendo então necessidade de utilizar outro chip.

Pode-se também implementar uma PROM através de transistores, segundo a tecnologia bipolar, CMOS ou NMOS.

Neste caso, a ligação entre a base a linha de dados será efetuada através de um fusível, que poderá ser queimado, com uma corrente adequada.

A vantagem das PROMs com a tecnologia CMOS ou NMOS é sua capacidade bem superior com relação às bipolares.

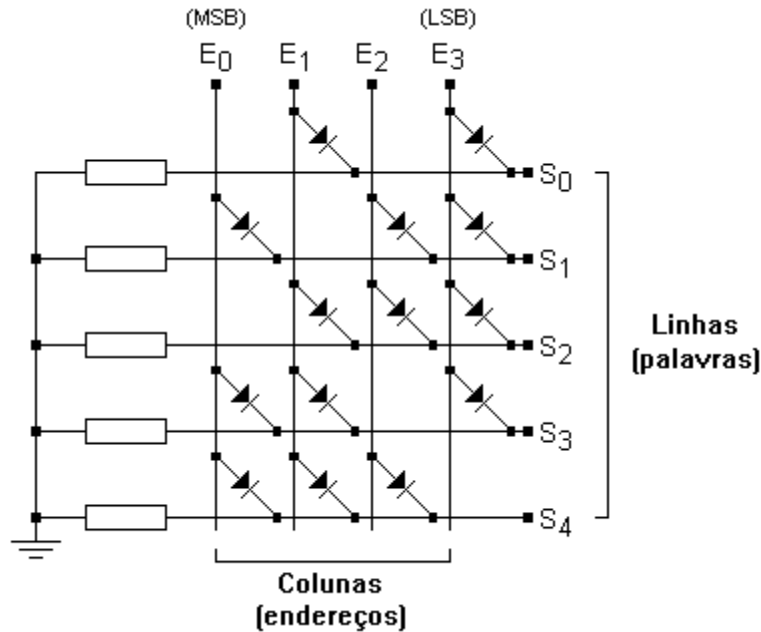
A figura a seguir mostra uma PROM implementada com diodos:



A tabela a seguir mostra a situação de uma PROM com diodos após a gravação pelo usuário, segundo suas necessidades:

ENDEREÇOS				PALAVRAS				
E ₀	E ₁	E ₂	E ₃	S ₀	S ₁	S ₂	S ₃	S ₄
1	0	0	0	0	1	0	1	1
0	1	0	0	1	0	1	1	1
0	0	1	0	0	1	1	0	1
0	0	0	1	1	1	1	1	0

O diagrama da PROM devidamente programada e gravada é mostrado a seguir:



Dentre as PROMs mais populares podemos citar:
 74186 - tecnologia bipolar, capacidade de 64 bytes
 TBP28S166 - tecnologia bipolar, capacidade 2K x 8
 TMS27PC256 - tecnologia CMOS, capacidade 32K x 8

EPROMs - ROM programável / apagável:

Uma EPROM pode ser programada pelo usuário, com a vantagem de poder ser apagada e reprogramada quantas vezes forem necessárias, sendo fabricadas segundo a tecnologia MOS, CMOS ou NMOS.

Uma vez programada uma EPROM comporta-se como uma memória *não volátil*, pois reterá seus dados indefinidamente.

O processo de programação de uma EPROM envolve a aplicação de níveis especiais de tensão situados entre 10 e 25V às entradas do chip, em intervalos com tempo determinado, da ordem de 50ms por posição de memória.

Em seu estado natural (sem qualquer programação) todas as células armazenam o bit 1.

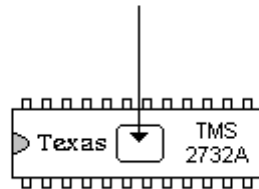
Dessa forma, durante a programação os pinos de dados e endereços da EPROM são usados para selecionar quais células são programadas como “0” e quais serão deixadas em “1”.

Uma vez que uma célula da EPROM tenha sido gravada, é possível apagá-la expondo-a à luz ultravioleta, aplicada através de uma janela no chip, requerendo para isso uma exposição que varia de 15 a 30 minutos.

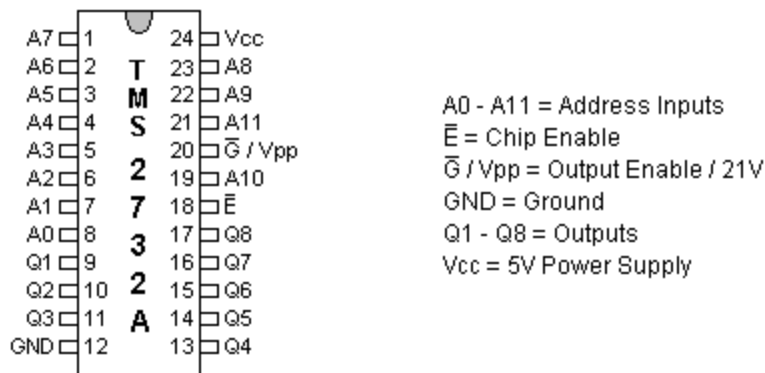
Entretanto, não é possível apagar células selecionadas, pois ao expor o chip à radiação ultravioleta todas as células se apagarão, voltando a nível “1”.

A figura a seguir ilustra uma EPROM:

Janela para apagamento
por ação da radiação ultravioleta



A figura abaixo ilustra o chip 2732A, (32768-BIT ERASABLE PROGRAMMABLE READ-ONLY MEMORY), apagável pela ação de radiação ultravioleta, que é uma EPROM organizada na forma 4K x 8.



Esse chip de memória possui 12 bits de endereços, pois $2^{12} = 4096$ e 8 saídas de dados.

Possui ainda 2 entradas de controle: E' é a entrada de habilitação usada para colocar o chip no estado *standby*, onde seu consumo de potência é reduzido.

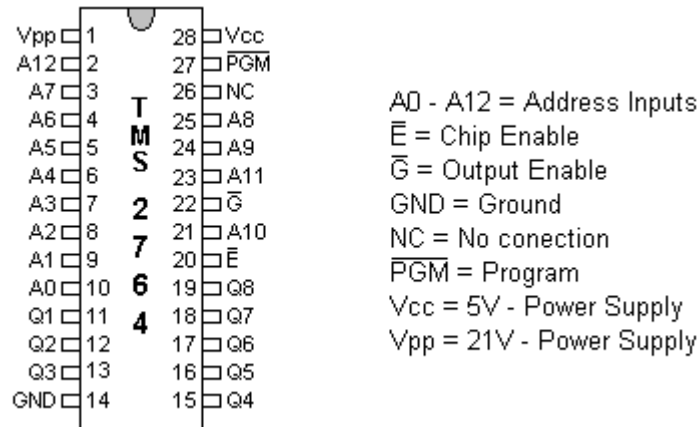
\bar{G} / V_{pp} é uma entrada dupla, onde: G' é a habilitação de saída usada para controlar os buffers de saída, de forma que o mesmo possa ser ligado ao barramento de um microprocessador sem provocar *bus contention* (disputa pelo barramento); Vpp é a tensão especial de programação, durante o processo de programação da EPROM.

A tabela abaixo mostra os 6 modos de operação do TMS2732A:

FUNCTION PINS	M O D E					
	Read	Deselect	Power Down (Standby)	Program	Program Verification	Inhibit Programming
\bar{E} (18)	V _{IL}	X	V _{IH}	V _{IL}	V _{IL}	V _{IH}
\bar{G} / V_{pp} (20)	V _{IL}	V _{IH}	X	21V	V _{IL}	21V
V _{cc} (24)	5V	5V	5V	5V	5V	5V
Q1 - Q8 (9 to 11 13 to 17)	Q	HI-Z	HI-Z	D	Q	HI-Z

V_{IL} e V_{IH} correspondem aos níveis baixo e alto respectivamente, compatíveis TTL; HI-Z corresponde ao estado de alta impedância; X significa don't care; D significa dado e Q significa saída.

O chip TMS2764 é uma EPROM, também apagável por radiação ultravioleta (65536-BIT ERASABLE PROGRAMMABLE READ-ONLY MEMORY), organizada como 8K x 8, conforme ilustra a figura abaixo:



É preciso ficar atento quanto à denominação dos chips. A letra “P” inserida entre os números 2732 e 2764, indica que trata-se de uma memória PROM. Desta forma, os chips TMS27P32A e TMS27P64 são memórias PROM.

As EPROMs apagáveis por ultravioleta, apresentam duas grandes desvantagens: a primeira é o fato de a mesma precisar ser retirada do soquete para ser apagada e reprogramada; a segunda é que a memória inteira é apagada com a aplicação de radiação ultravioleta.

Para contornar esse inconveniente foi desenvolvido outro tipo de EPROM, que pode ser apagada por pulsos elétricos.

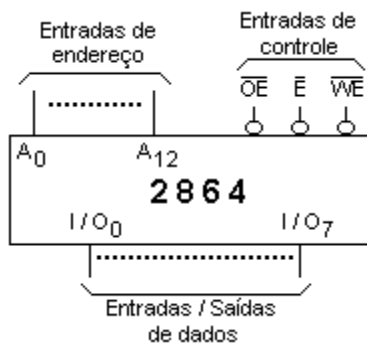
A grande vantagem da memória por apagamento com pulsos elétricos com relação a memória por apagamento com radiação ultravioleta é que o apagamento e a reprogramação dos dados pode ser feita seletivamente ao invés da memória toda, além do que, para ser programada ou reprogramada, o chip não precisa ser removido do circuito.

Enquanto que uma EPROM por ultravioleta leva entre 15 a 20 minutos para ser apagada, uma EPROM por impulsos elétricos tem esse tempo reduzido para alguns milissegundos, o que constitui outra grande vantagem.

O primeiro chip EPROM apagável por pulsos elétricos foi o Intel 2816 com capacidade de 2K x 8. A partir daí esse tipo de memória foi bastante aperfeiçoado.

Um chip mais moderno, o 2864 com capacidade 8K x 8, contém internamente um circuito que gera as tensões necessárias para apagamento e reprogramação, de modo a necessitar apenas a alimentação de V_{cc} .

A figura a seguir mostra a configuração de um chip 2864:



O chip 2864 é bastante versátil, podendo ser comparado às RAMs estáticas em termos de operação, constituindo uma vantagem por ser *não volátil*. No entanto em comparação às RAMs estáticas, a desvantagem é que possui circuitos mais complexos e tempo de acesso maior.

Analisando o digrama do chip 2864, pode-se observar que os pinos I/O (dados), podem ser utilizados tanto para a entrada como para a saída, dependendo dos níveis nas entradas de controle, conforme ilustra a tabela abaixo:

MODO	E'	OE'	WE'	SAÍDAS
Leitura	baixo	baixo	alto	DATA _{OUT}
Escrita	baixo	alto	baixo	DATA _{IN}
Standby	alto	X	X	HI-Z

APLICAÇÕES DA ROM

As ROMs podem ser usadas em qualquer tipo de aplicação que exija armazenamento de dados de características *não volátil*.

FIRMWARE (MICROPROGRAMA): Uma das mais importantes aplicações da memória ROM está no armazenamento de micro programas de um computador.

Alguns computadores armazenam também em ROM seu sistema operacional e em alguns casos, interpretadores de linguagem.

Muitos produtos de consumo como, jogos eletrônicos, sistemas eletrônicos de injeção de combustíveis em automóveis, etc. usam ROMs.

Os programas de computador que estão armazenados em ROMs denominam-se *firmware*, pelo fato de seus dados não estarem sujeitos à mudanças, ao contrário daqueles armazenadas em RAMs (*software*), que são modificados com facilidade.

BOOTSTRAP - (MEMÓRIA DE PARTIDA FRIA): Alguns computadores não possuem seu sistema operacional armazenado em ROM. Ao invés disso, usam memória de massa para sua inicialização (normalmente discos magnéticos).

Para que o computador saiba o que fazer após sua ligação, um programa muito pequeno denominado *programa de partida fria ou bootstrap* armazenado em uma ROM é executado logo que o mesmo é ligado.

As instruções do programa levam o processador a inicializar o sistema, fazendo com que parte residente do sistema operacional seja transferida da memória de massa para a memória interna.

TABELAS DE DADOS: As ROMs são muito utilizadas para armazenar dados que não mudam nunca, como por exemplo: tabela de implementação de funções trigonométricas e tabelas de códigos.

O chip MM4220BM fabricado pela NATIONAL SEMICONDUCTOR por exemplo, armazena valores da função seno para ângulos de 0 a 90°.

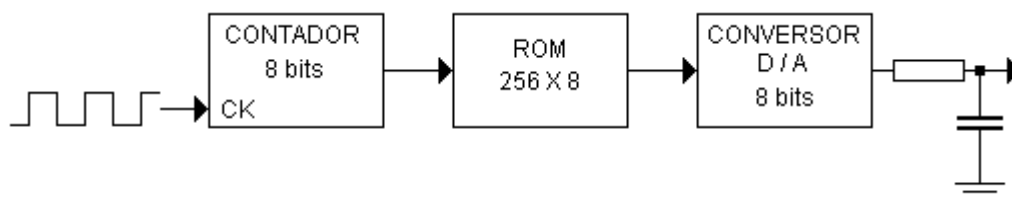
CONVERSORES DE DADOS: A conversão de dados é necessária, quando o microprocessador está entregando dados em binário puro e tais dados devem ser entregues por exemplo, a um display de 7 segmentos.

Uma forma de implementar essa conversão é a utilização de uma ROM programada para tal fim. O chip 74185 é uma ROM TTL que armazena os dados necessários para efetuar a conversão binário-BCD para uma entrada de 6 bits.

GERADORES DE CARACTERES: Os caracteres alfanuméricos são formados por um grupo de pontos, que podem estar brilhando ou não, dependendo do caractere a ser mostrado. Esse grupo de pontos normalmente é arranjado como uma matriz de pontos 5 x 7 ou 7 x 9, sendo esses pontos representados por um código binário (0 e 1).

Uma ROM geradora de caracteres armazena os códigos do padrão de pontos de cada caractere em um endereço que corresponde ao código ASCII. Por exemplo, o padrão de pontos para o caractere A deve estar armazenado no endereço 1000001, onde 1000001 é o código ASCII para a letra A.

GERADORES DE FUNÇÕES: O gerador de funções é um circuito que deve fornecer em sua saída formas de onda diversas (senoidal, quadrada, triangular, etc). Seu diagrama é mostrado abaixo:

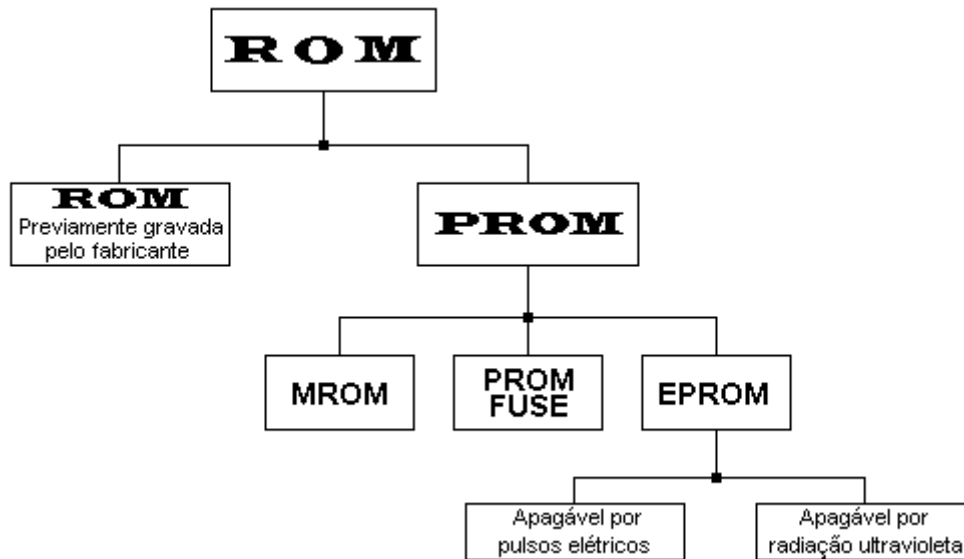


Na ROM é armazenada uma tabela com 256 valores diferentes de 8 bits, cada um deles correspondente ao valor de um ponto da curva que deva ser representada.

Os endereços da ROM são gerados ciclicamente por um contador de 8 bits, que pulsa continuamente pela ação do clock.

Quando o contador tiver percorrido todos os 256 endereços da ROM, esta terá gerado em sua saída todos os 256 pontos necessários para que o conversor D/A produza a forma de onda desejada.

A figura a seguir sintetiza tudo o que foi dito até agora sobre as ROMs e suas divisões:



RAM - (RANDOM ACCESS MEMORY) SRAM - (STATIC RANDOM ACCESS MEMORY)

As memórias RAM, também conhecidas como memórias de escrita e leitura (RWM) são usadas em computadores para armazenamento temporário de dados.

ARQUITETURA DE UMA RAM

A desvantagem das RAMs é que por serem *voláteis* os dados nela armazenados se perdem quando o computador é desligado, isto é, armazenam dados por tempo indeterminado, enquanto a alimentação estiver sendo aplicada ao chip.

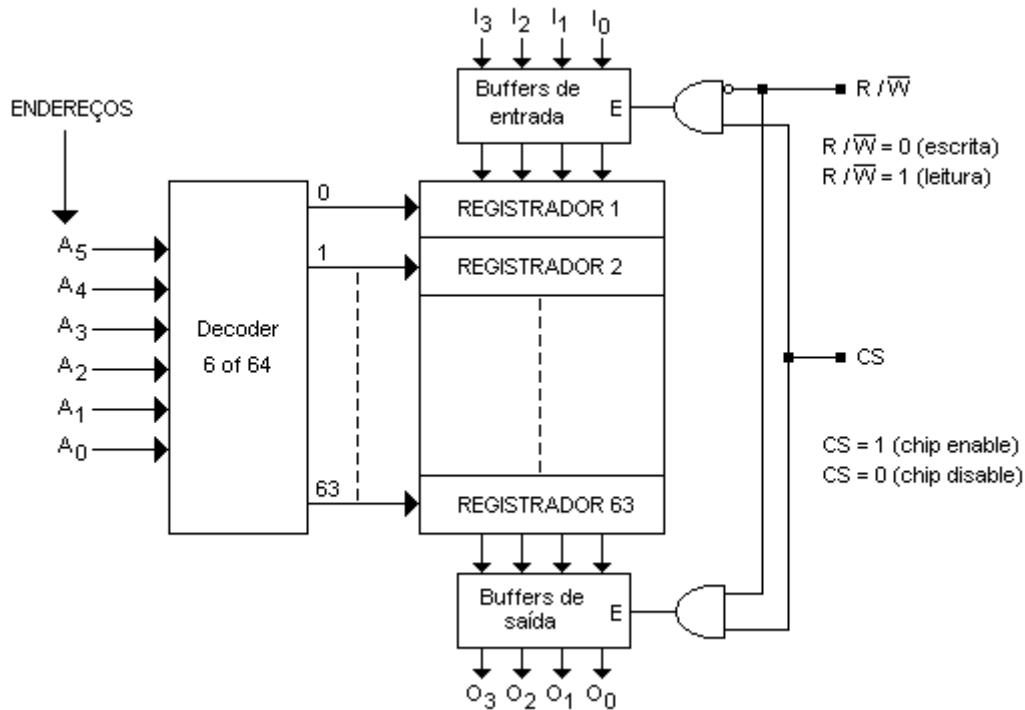
Por esse motivo, essas RAMs são denominadas **RAMs ESTÁTICAS (SRAM)**.

A grande vantagem das RAMs é que podem ser lidas e escritas rapidamente.

Tomemos como exemplo uma RAM que armazena 64 palavras de 4 bits cada, organizada na forma 64 x 4, conforme ilustra a figura a seguir:

Para selecionar uma das 64 posições de memória para leitura e escrita o código de endereços é aplicado à entrada de um decodificador (6 para 64). Como $64 = 2^6$, então torna-se necessário 6 linhas de endereço.

Quando o código de endereço $A_5A_4A_3A_2A_1A_0 = 011110$ for aplicado na entrada de endereços, a saída do decodificador estará em nível alto, selecionando o registrador 30 para leitura ou escrita, pois $011110_2 = 30_{10}$.



CS = Chip select R/W = Read/Write

Escrita: Um dos registradores da memória é escolhido pelo código de endereço para ser escrito. Para escrever uma nova palavra é necessário que R/W' esteja em 0 e CS em 1.

Esta combinação proporciona a habilitação dos buffers de entrada, desabilitando os buffers de saída, que normalmente operam em alta impedância durante a operação de escrita.

Leitura: Da mesma forma que anteriormente, o código de endereço seleciona um registrador para ser lido. Para isto, é necessário que R/W' esteja em 1 e CS em 1.

Isto proporciona a habilitação dos buffers de saída, desabilitando os buffers de entrada, que normalmente operam em alta impedância durante a operação de leitura.

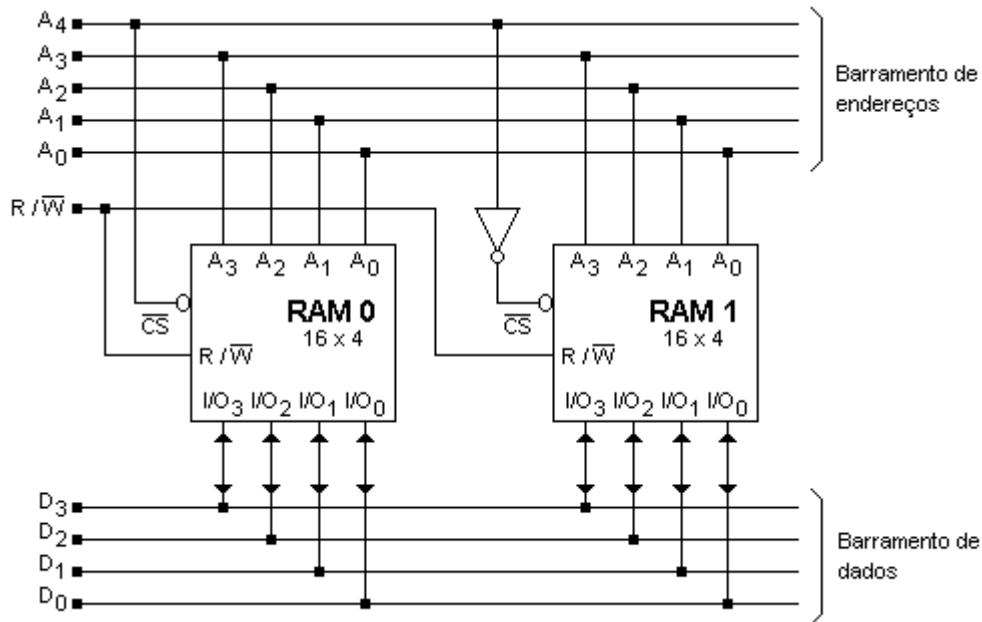
Seleção do chip (chip select): Muitas memórias possuem mais de uma entrada de seleção de chip, que quando desabilitado, não permite operações de escrita e leitura.

Quando o chip está desabilitado, consome muito pouca energia, o que é muito útil para sistemas de memória que empregam vários chips para aumentar sua capacidade, uma vez que, somente um dos chips deverá estar selecionado para a realização de uma operação.

AMPLIAÇÃO DA CAPACIDADE DE UMA RAM

Como exemplo podemos citar a necessidade de uma memória com capacidade 32 x 4 (32 palavras de 4 bits), mas dispomos apenas de chips 16 x 4.

Veja o diagrama abaixo:



O arranjo acima ilustrado, normalmente denominado *módulo de memória*, abrange os endereços:

RAM 0 - 00000 a 01111

RAM 1 - 10000 a 11111

TOTAL - 00000 a 11111 (32 palavras = $2^5 = 5$ linhas de endereços)

Observa-se que o endereço A_4 tem por função selecionar uma das duas RAMs através da entrada CS' a partir dos níveis 1 ou 0.

Quando $CS' = 1$, habilita a RAM 0, desabilitando a RAM 1.

Desta forma, a faixa de endereços para as RAMs 0 e 1 ficará por conta das entradas $A_3A_2A_1A_0$.

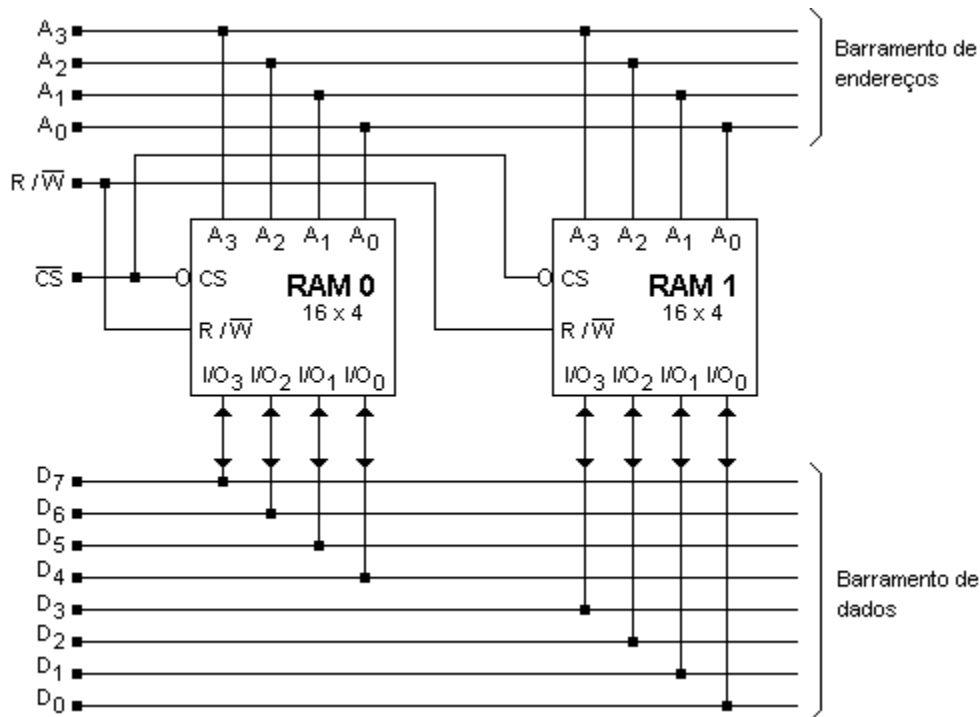
AMPLIAÇÃO DO TAMANHO DA PALAVRA DE UMA RAM

Supondo que dispomos de chips de memória RAM 16 x 4, mas precisamos apenas dobrar o tamanho da palavra, mantendo a mesma quantidade de palavras.

Isto nos daria uma organização 16 x 8.

Neste caso cada chip seria utilizado para armazenar a metade de bits de cada palavra.

A figura a seguir mostra esse arranjo:



Para o arranjo acima, a faixa de endereços compreende: 0000 a 1111, que corresponde a 16 palavras.

A RAM 0 armazena os 4 bits da mais significativos da palavra, enquanto que a RAM 1 armazena os 4 bits menos significativos da mesma palavra.

As entradas R/W' e CS' controlam a operação escrita/leitura.

Para leitura R/W' = 1 e CS' = 0; para escrita R/W' = 0 e CS' = 0.

O barramento de dados funciona como entrada e saída, podendo ser interligado a qualquer outro dispositivo, como por exemplo, um processador.

IMPLEMENTAÇÃO DE UMA RAM

O diagrama a seguir representa uma RAM implementada com FFs RS, com capacidade de 4 bits (4 x 1).

Observe que a mesma possui 2 entradas de endereço.

Para armazenar um dado no FF2 por exemplo, procede-se da seguinte maneira:

a) a entrada X deve estar em nível 1

b) $A_0A_1 = 10 = 2_{10}$

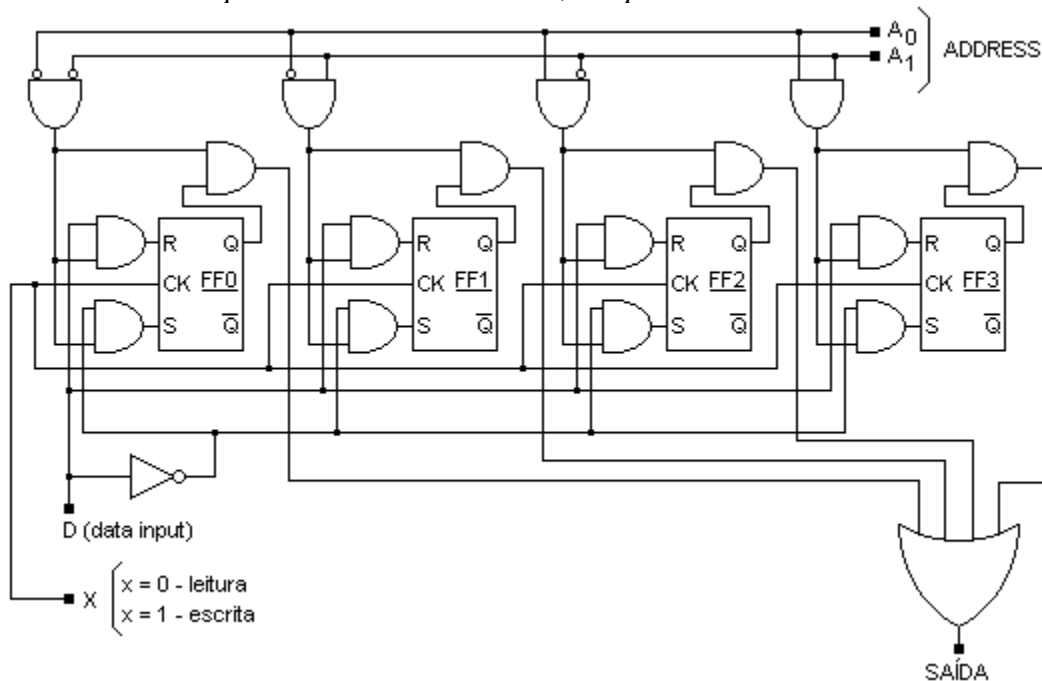
c) aplicar o dado que se deseja armazenar na entrada D (0 ou 1)

Com o endereço 10, somente o FF2 será ativado. As outras entradas dos demais FFs manterão o estado atual, de modo que a informação armazenada nos mesmos não será destruída.

Para leitura, basta levar X a nível 0.

Para ler por exemplo, a informação no FF3 basta endereçar $A_0A_1 = 11$.

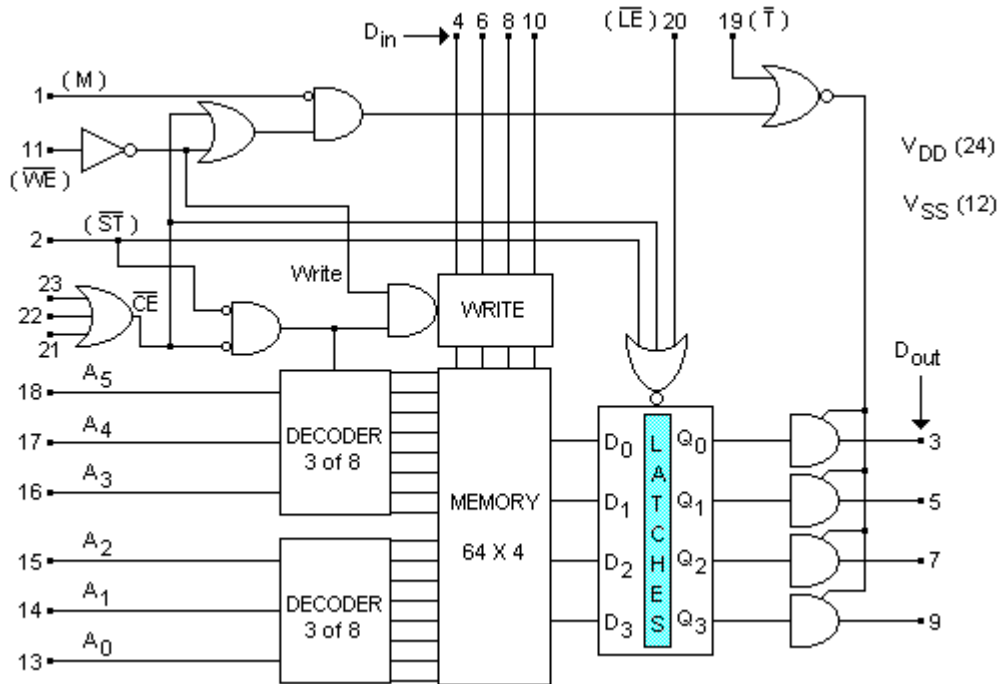
Durante o processo de leitura, as informações armazenadas não serão destruídas, uma vez que CK estando em nível 0, bloqueará todos os FFs.



Para ler por exemplo, a informação no FF3 basta endereçar $A_0A_1 = 11$.

Análise do chip MCM14552

O chip MCM14552 é uma RAM estática, produzido pela Motorola, organizado na forma 64 x 4 (256-BIT STATIC RANDOM ACCESS MEMORY), cujo diagrama lógico é mostrado a seguir:

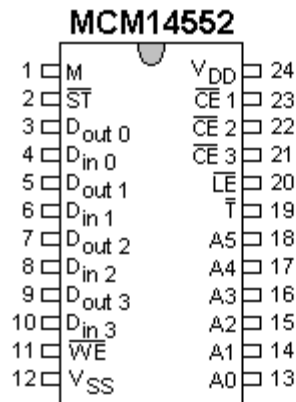


A tabela para as operações principais é mostrada abaixo:

FUNÇÕES	CE'1	CE'2	CE'3	T'	LE'	M	ST'	WE'
Escrita	0	0	0	X	X	X	0	0
Leitura	0	0	0	0	X	X	X	1
Desabilitação das saídas (HI-Z)	X	X	1	X	X	0	X	X
	X	1	X	X	X	0	X	X
	1	X	X	X	X	0	X	X
	X	X	X	1	X	X	X	X
	X	X	X	X	X	0	X	0

O chip MCM14552 é um circuito integrado fabricado segundo a tecnologia CMOS, de larga escala de integração (LSI).

O diagrama de sua pinagem é mostrado a seguir:



A entrada M é usada para mudar o controle lógico do circuito.

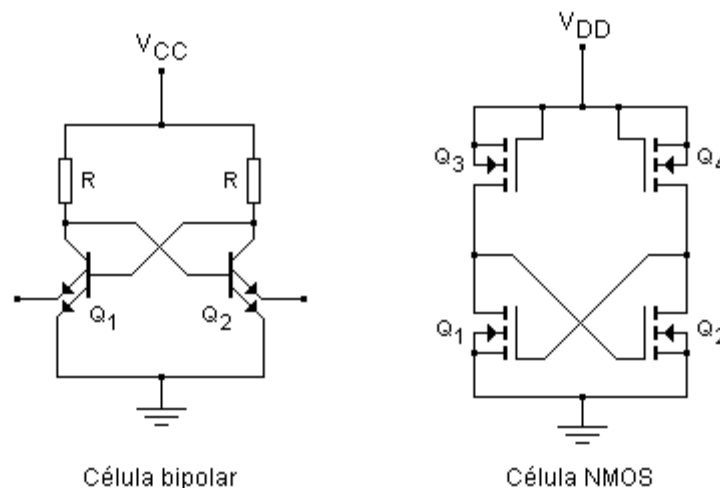
Quando M = 1, a entrada \overline{T} assume o controle de alta impedância das saídas.

Quando M = 0, a característica de alta impedância das saídas é assumida pelas entradas \overline{CE} (1, 2 e 3), \overline{WE} e \overline{T} .

As 3 entradas \overline{CE} (1, 2 e 3) podem ser usadas como extensão dos 6 endereços de entrada, criando desta forma um esquema de 9 bits de endereçamento.

As RAMs estáticas também podem ser fabricadas a partir da tecnologia bipolar, porém, a maioria emprega as tecnologias CMOS e NMOS.

A figura abaixo mostra uma comparação entre uma célula bipolar e uma célula NMOS.



A tecnologia bipolar tem a vantagem da velocidade com relação a tecnologia da família MOS.

No entanto, os dispositivos da família MOS tem maior capacidade de armazenamento e menor consumo de potência. Conforme ilustra a figura acima, a célula bipolar utiliza dois transistores bipolares e dois resistores.

A célula NMOS utiliza quatro MOSFETs de canal negativo. A célula bipolar requer mais espaço no chip devido a sua maior complexidade e necessitam de resistores, enquanto que na célula NMOS esses resistores são substituídos pelos transistores Q₃ e Q₄.

Uma célula de memória CMOS é similar à NMOS, exceto pelo fato de utilizar MOSFETs de canal positivo nos lugares de Q₃ e Q₄, resultando em um chip com baixo consumo de potência, mas com arquitetura bem mais complexa.

DRAM - (DYNAMIC RANDOM ACCESS MEMORY)

As RAMs dinâmicas, também denominadas DRAMs, possuem alta capacidade de armazenamento e são fabricadas segundo a tecnologia MOS.

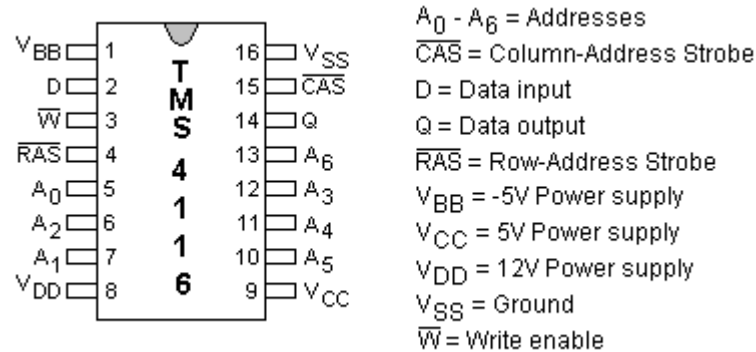
Enquanto que as RAMs estáticas armazenam informações em flip-flops, as RAMs dinâmicas armazenam os bits 0 e 1 em micro capacitores parasitas nos transistores MOS que constituem sua célula.

Devido a corrente de fuga essas informações podem ser perdidas após um determinado período, necessitando pois de um processo de restauração periódica.

O ato de restaurar um dado armazenado em uma RAM dinâmica chama-se *refresh*.

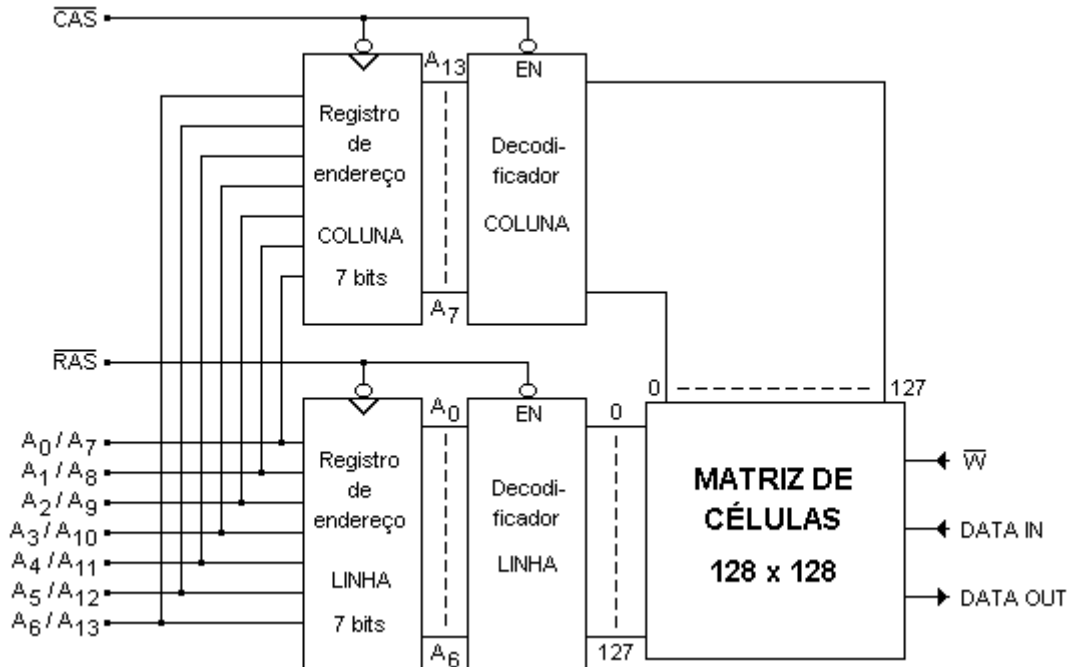
A necessidade da operação de refresh torna a RAM dinâmica desvantajosa em relação à RAM estática.

Uma DRAM muito popular é a TMS4116 (16384-BIT DYNAMIC RANDOM ACCESS MEMORY), organizada na forma 16K x 1, fabricada com a tecnologia MOS, cujo diagrama de pinagem é mostrado abaixo:



Para esse tipo de memória, a operação refresh deve ser realizada a cada 2ms (milissegundos), para reter os dados.

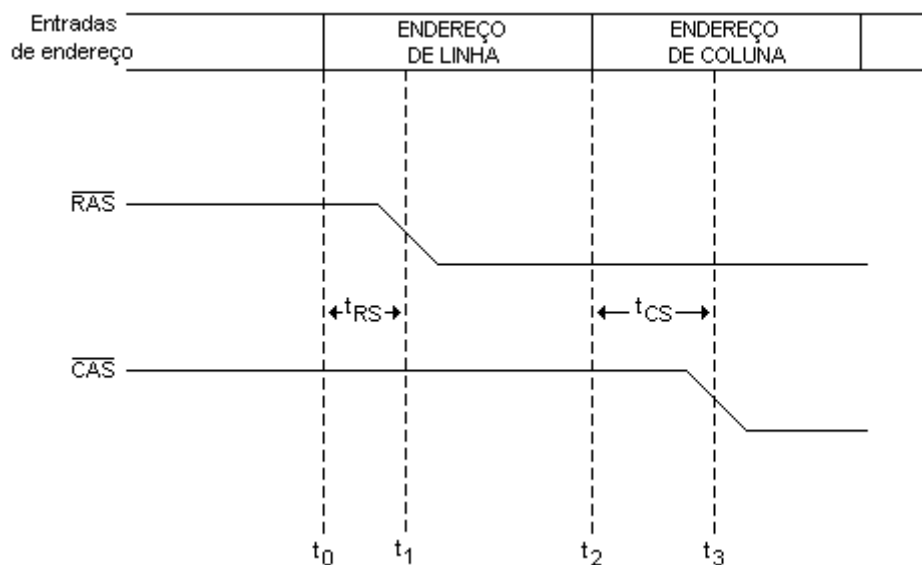
Sua arquitetura simplificada é mostrada na figura a seguir:



Como essa DRAM necessita de 14 bits de endereço, utiliza-se uma técnica denominada *multiplexação de endereços*, técnica essa, muito usada para reduzir o tamanho do invólucro do chip.

No caso da DRAM TMS4116, um endereço de 14 bits é aplicado em dois momentos através das entradas CAS' e RAS', responsáveis pelo armazenamento dos endereços de linha e coluna (RAS' para linhas e CAS' para colunas).

Para melhor entender o conceito de multiplexação de endereços, analisaremos os sinais de temporização de RAS' e CAS' conforme ilustra a figura abaixo:



Inicialmente RAS' e CAS' estão em nível alto.

No instante t_0 sete bits menos significativos do endereço, correspondentes ao endereço de linha ($A_0 - A_6$), são aplicados à entrada de endereços.

Depois de decorrido o tempo de setup (t_{RS}) a entrada RAS' passa para nível baixo, em t_1 .

Na transição negativa deste sinal, essa parte do endereço é carregada em um registrador interno de maneira que os sinais $A_0 - A_6$ aparecem na entrada do decodificador de linha.

O nível lógico baixo proveniente de RAS' habilita então o decodificador de linha, de forma que o mesmo decodificará o endereço de linha, selecionando uma das linhas da matriz.

No instante t_2 os bits restantes ($A_7 - A_{13}$), correspondentes ao endereço da coluna, são aplicados à entrada de endereços. Em t_3 CAS' é levada a nível baixo, comandando o armazenamento do endereço da coluna no registrador correspondente.

Da mesma forma, o decodificador de coluna é habilitado, decodificando então o endereço de coluna, selecionando uma das colunas da matriz.

Nestas condições as duas partes do endereço já foram devidamente processadas, para selecionar uma das células da memória para uma operação de escrita ou leitura, exatamente como no caso de uma RAM estática.

Conclui-se então que os sinais RAS' e CAS' realizam a função de seleção do chip, dispensando assim a entrada CS.

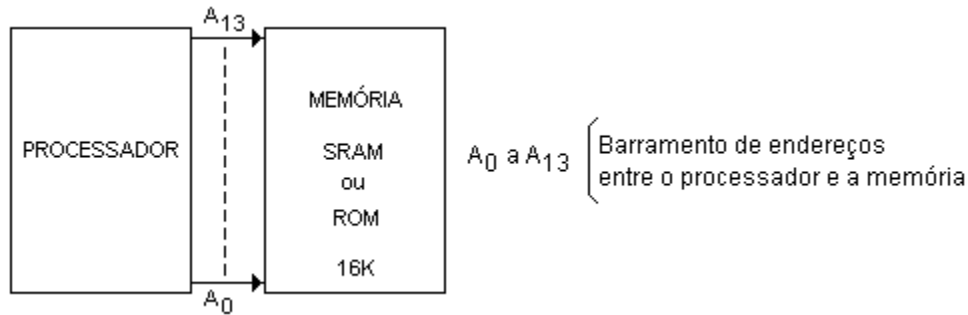
Desta forma, para uma DRAM 16 x 1, ao invés de se utilizar 14 entradas de endereço, utiliza-se apenas 7, segundo a técnica da multiplexação de endereços.

O processo de temporização das operações de leitura e escrita de uma DRAM é muito mais complexo em relação a uma SRAM.

Para se ter uma idéia, vamos considerar um barramento de endereços de um processador alimentando uma SRAM ou ROM e um barramento de endereços alimentando uma DRAM.

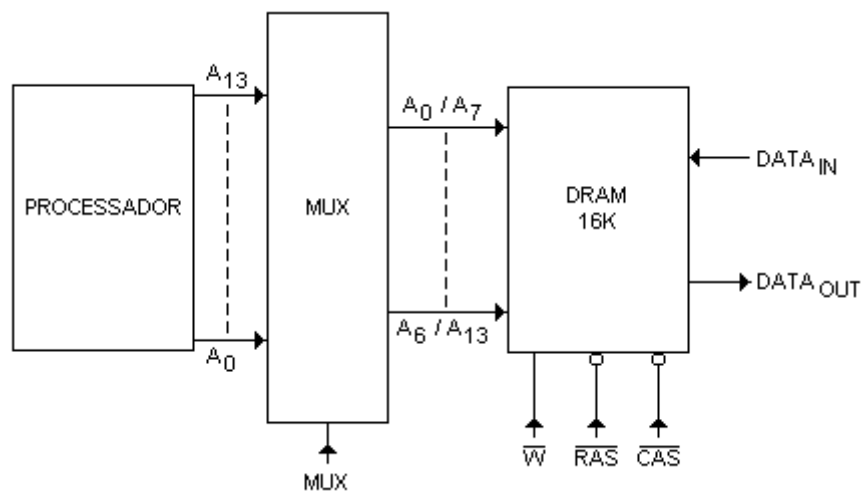
As figuras a seguir ilustram de maneira bem simplificada essas condições:

A) Barramento de endereços de um processador para uma SRAM ou ROM de 16K. O barramento de endereços compreende as linhas A_0 a A_{13} que interligam o processador com o sistema de memória.



B) Barramento de endereços de um processador para uma DRAM com a mesma capacidade. Observa-se que entre o processador e a DRAM existe um multiplexador, que recebe as 14 linhas de endereço do processador e entrega 7 linhas de endereço para a DRAM, devidamente multiplexadas.

A multiplexação ocorre nas seguintes condições: quando MUX = 0, ocorre a transmissão das linhas de endereço A_0/A_6 para a DRAM; quando MUX = 1, ocorre a transmissão das linhas de endereço A_7/A_{13} para a DRAM.



Como já foi dito anteriormente, uma das desvantagens das memórias dinâmicas é a necessidade de recarregar suas células periodicamente (em média 2ms), caso contrário, todos os dados nela armazenados serão perdidos.

Teoricamente, a operação refresh deve ser executada toda vez que ocorrer uma operação de leitura em uma determinada célula.

Levando-se em consideração uma DRAM de 16K (16384 bits), seria necessária então uma operação de leitura da ordem de 122ns (nano segundos) para cada célula, taxa de leitura esta, muito elevada para a maioria das DRAMs disponíveis no mercado, além de ser pouco provável que todas as 16384 células fossem lidas em 2ms.

Em vista disto as DRAMs em sua maioria, são projetadas de tal forma que, quando uma operação de leitura for feita em uma determinada célula, todas as células dessa mesma linha sofrerão refresh.

Dessa forma a operação refresh fica bastante simplificada, pois limita-se a apenas 128 linhas, isto é, a operação refresh ocorrerá em toda a linha se nesta linha pelo menos uma célula for lida.

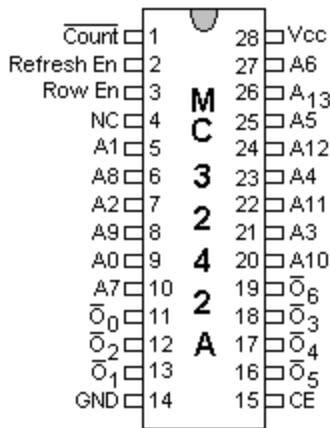
Uma forma bastante simples de implementar a operação refresh, é a utilização de um contador refresh de 7 bits usado para gerar ciclicamente os 128 endereços das linhas da DRAM (neste caso, a DRAM de 16K, TMS4116).

O contador gera então 0000000 que corresponde a 0, até 1111111 que corresponde a 127, processo esse que demora aproximadamente 50µs (microsegundos).

O endereço gerado pelo contador não pode interferir com os endereços vindos do processador durante as operações normais de escrita e leitura. Em virtude disso os endereços do contador devem ser multiplexados com os endereços do processador, para que não ocorra qualquer tipo de conflito entre endereços.

Para facilitar esse tipo de operação, muitos fabricantes de circuitos integrados desenvolveram chips especiais para essa finalidade, como por exemplo o CI MC3242A (*MEMORY ADDRESS MULTIPLEXER AND REFRESH COUNTER*).

Esse chip recebe o nome de *controlador de RAM dinâmica*, cujo diagrama de pinagem é mostrado abaixo:



O pino 15 (CE - chip enable) é uma entrada opcional.

Quando esse pino e deixado em aberto, assume valor alto, mantendo assim performance idêntica ao chip da Intel 3242.

O chip em questão possui na saída 7 bits de endereço multiplexados, os quais são ligados à entrada de endereços da DRAM, que pode ser alimentada de três fontes possíveis:

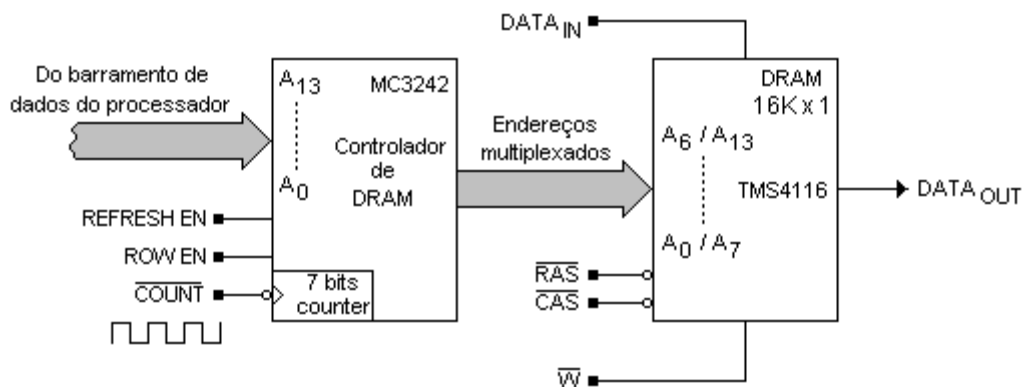
A primeira fonte é a saída do contador refresh que é incrementado por um pulso externo de clock, aplicado na entrada Count' (contador de refresh interno), que fornece os endereços de linha para a DRAM durante a operação refresh.

As duas outras são as saídas multiplexadas do controlador, obtidas da multiplexação dos 14 bits de endereço oriundos do processador, que são transformados em endereços de linha e de coluna, utilizados quando o processador precisa ler ou escrever na DRAM.

Os níveis lógicos aplicadas nas entradas Refresh Enable e Row Enable, determinam qual dos três conjuntos de 7 bits vai aparecer na saída do controlador, conforme mostra a tabela abaixo:

Refresh EN	Row EN	Saída do MC3442A
H	X	Endereço de refresh (obtido do contador interno)
L	H	Endereço de linha (A ₀ - A ₆)
L	L	Endereço de coluna (A ₇ - A ₁₃)

O diagrama a seguir ilustra o processo de operação descrito acima:



OUTROS TIPOS DE MEMÓRIAS

RAM NÃO VOLÁTIL:

Como sabemos as RAMs estáticas são utilizadas para leitura e escrita, devido a sua alta velocidade na realização dessas operações, no entanto, por serem voláteis, os dados nela armazenados se perdem quando cessa a alimentação elétrica.

Existem no entanto, duas soluções para esse problema:

A primeira é usar memórias que possam ser alimentadas por baterias sempre que ocorrer falta de energia. Isto requer memórias de baixo consumo, de modo a não consumir rapidamente toda a carga da bateria. Não resta dúvidas de que a tecnologia CMOS seria a mais indicada para o caso e para tal as memórias deveriam ser mantidas em *standby* para consumir o mínimo possível de energia. Algumas SRAMs com tecnologia CMOS incluem no chip pequenas baterias de lítio (lithium).

Outra solução emprega a NVRAM (RAM não voláteis).

Uma NVRAM possui uma matriz de RAM estática e uma matriz de EAPROM no mesmo chip, combinando dessa forma a velocidade de operação de uma SRAM com a capacidade de armazenamento de uma EAPROM.

Cada célula de SRAM corresponde a uma cela de EAPROM e as informações podem ser transferidas entre as células em ambas as direções.

Na ocorrência de falta de energia ou mesmo quando o computador for desligado, ocorre o seguinte:

a) um circuito sensor de tensão detecta a queda de tensão AC e envia um sinal à entrada Store da NVRAM;

b) com isto o conteúdo armazenado na NVRAM é transferido das células de RAM estática para as correspondentes da EAPROM; essa transferência é realizada em paralelo (da ordem de alguns milissegundos);

c) até que a energia da fonte se esgote totalmente (devido a descarga dos capacitores da fonte de 5V DC) , isto manterá a NVRAM energizada, até que a transferência se complete;

d) a partir daí a EAPROM guarda uma cópia do conteúdo da SRAM;

e) restabelecendo-se a energia, a NVRAM comanda a transferência automática das informações da EAPROM para a SRAM.

Uma NVRAM tem a vantagem de não necessitar de bateria, sendo no entanto mais complexa do que um chip de memória convencional e por isso não estão disponíveis para altas capacidades de armazenamento.

Quando se requer o uso de memórias não voláteis de alta capacidade, a alternativa é a utilização de RAMs CMOS com bateria.

MEMÓRIAS FIFO:

As FIFOs (First Input - First Output), que traduzindo significa *Primeiro a entrar - Primeiro a Sair*, é um tipo de memória de acesso sequencial, na qual são usados registradores de deslocamento na sua implementação.

Isto significa que à medida que as palavras entram na entrada de dados, são descarregadas na mesma ordem na saída de dados, ou seja, a primeira palavra escrita será a primeira palavra a ser lida.

Desta forma, a FIFO é muito utilizada na transferência de dados entre sistemas que operam com velocidades diferentes. Como exemplo mais comum, podemos citar a transferência de dados de um computador para uma impressora.



O computador envia dados para uma impressora a uma velocidade muito superior a que a impressora possa aceitá-los.

A memória FIFO neste caso age com um buffer equalizador de taxa de dados, aceitando os dados que vem do computador em velocidade alta, digamos, 40.000pps (pulsos por segundo) e armazenando-os.

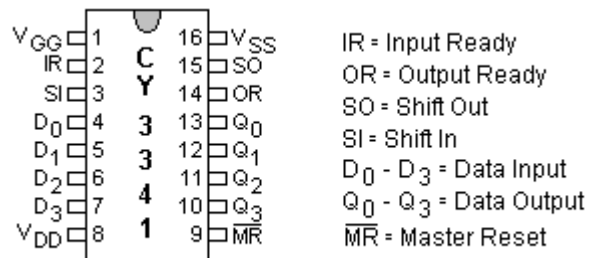
Após isso serão deslocados para uma impressora a uma velocidade bem mais baixa, digamos, 400pps.

A FIFO também pode ser utilizada para operar de forma contrária, ou seja, transmitir dados de um dispositivo extremamente lento (como por exemplo o teclado), para um dispositivo rápido como o computador.

A figura a seguir ilustra o diagrama de pinagem do CI comercial CY3341. Dentre as suas características pode-se mencionar:

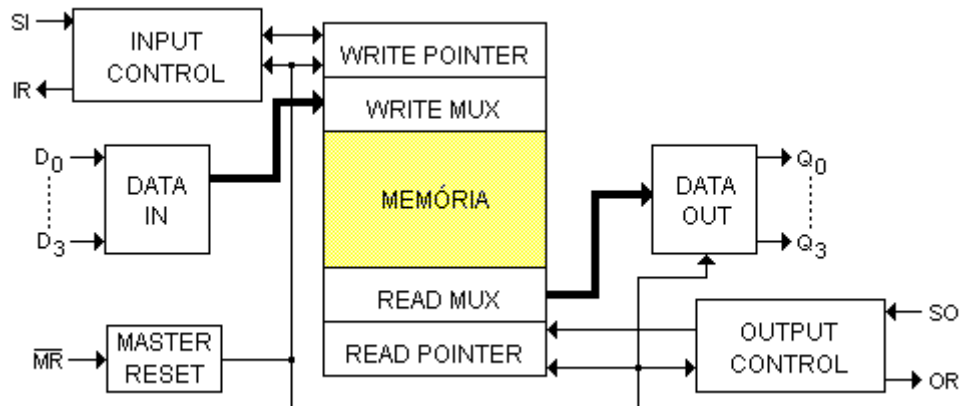
- a) taxa de dados 1,2/2 MHz;
- b) sincronização independente das entradas e saídas;
- c) tecnologia CMOS;
- d) expansível verticalmente e horizontalmente.

(64 x 4 FIFO SERIAL MEMORY)



A entrada dos dados (D₀ - D₃) é controlada pelas entradas SI e IR, enquanto que a saída dos dados é controlada pelas entradas SO e OR. A entrada Master Reset atua diretamente nos controles lógicos da entrada e saída, inibindo-os, quando à mesma for aplicado o nível lógico 0.

O diagrama lógico mostrado a seguir torna isto mais claro:



MEMÓRIAS DE BOLHAS MAGNÉTICAS (MBM):

A memória de bolhas magnéticas é uma memória a semicondutor, que armazena informação binária em forma de bolhas magnéticas muito pequenas, formadas sobre um filme fino de material magnético.

A presença ou não de uma bolha em uma determinada posição, é interpretada como nível 1 ou 0.

A movimentação das bolhas ocorre em razão da mudança do campo magnético, de maneira similar aos registradores de deslocamento circular.

Os dados circulam a partir de um determinado ponto de captura a uma taxa da ordem de 50.000 bits por segundo.

As MBM não são memórias sequenciais e tem a vantagem de não serem voláteis.

A vantagem das MBMs em relação as memórias não voláteis ROMs, PROMs, EPROMs é que podem ser lidas e escritas com a mesma facilidade.

Comparando as MBMs com o armazenamento de dados em fitas magnéticas, as MBMs são mais confiáveis, por não possuírem partes mecânicas.

Um dispositivo típico de MBM pode armazenar mais de 1 milhão de bits. No entanto, devido a sua baixa velocidade (da ordem de 1 a 20ms), não são empregadas como memória principal, mas, muito utilizadas como memória de massa.

Infelizmente, seu custo é muito alto, devido a complexidade dos circuitos internos.

Com a evolução tecnológica e a diminuição de custos, futuramente as MBMs substituirão os disquetes, pois, as MBM são em torno de 100 vezes mais rápidas do que os disquetes ,além do que, muito mais confiáveis.

CACHE DE RAM:

Como já sabemos, a performance de um computador está diretamente relacionada com sua velocidade no processamento de dados e instruções.

Alguns fabricantes, para manter os computadores dentro de um limite razoável de preço, utilizam os chips de memória RAM mais lentos como parte principal de memória e os chips de memória RAM mais rápidos e mais caros na placa mãe (motherboard), como memória de cache de RAM externo, que geralmente varia entre 64K a 512K.

Sem o cache, o processador poderia ficar vários ciclos de clock inativo, esperando que os dados solicitados lhes fossem transmitidos.

Convém esclarecer que, um ciclo de clock é o menor espaço de tempo durante o qual uma operação pode durar em um computador.

O cache tem a finalidade de manter sempre prontos os dados mais prováveis de serem solicitados pelo processador. A eficiência do cache de RAM externo é controlado por dois fatores:

- a) velocidade
- b) algoritmo

O algoritmo é importante, uma vez que, determina quais os dados que deverão ser armazenados no cache, ou seja, quais os dados que são solicitados mais vezes pelo processador durante a execução de um programa.

Desta forma, quando o processador solicita um determinado dado, o cache consegue fornecê-lo através de seus chips mais rápidos.

Quando um novo dado é solicitado por algum programa, o cache substitui o dado mais antigo (contido nos chips mais rápidos) pelo novo dado e pelos demais que ficam ao redor dos endereços da memória, obedecendo a regra FIFO (First-In First-Out), isto é, o dado que está mais tempo sem uso tem menos chances de ser futuramente solicitado pelo programa.

Ciclo de operação:

I - Um programa qualquer solicita através da CPU dados para serem utilizados na própria CPU;

II - O cache de RAM intercepta essa solicitação que está a caminho da memória, pega esses dados e os envia à CPU;

III - O cache também armazena uma cópia dos dados que foram gravados no cache (chips de alta velocidade);

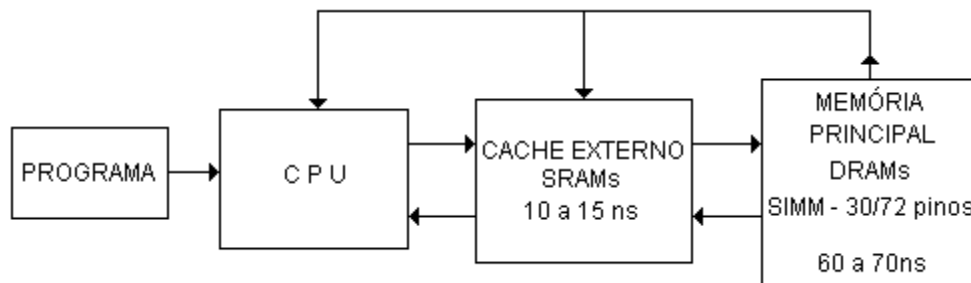
IV - Assim que o cache nota que a CPU está inativa, pega os dados do programa dos endereços de memória adjacentes e leva para os endereços dos dados que o programa solicitou inicialmente. O cache armazena então esses dados;

V - Quando o programa solicitar novamente que os dados sejam enviados à CPU, o cache verifica se tais dados já estão armazenados no cache. Se estiverem, o cache os envia diretamente à CPU, sem ter que acessar os chips de memória mais lentos. Desta forma, a CPU fica menos tempo inativa;

VI - Quando a CPU quer alterar algum dado que já está na memória, o cache verifica primeiro se o dado a ser alterado encontra-se nos chips de alta velocidade. Se estiver, ele compara o dado que possui e envia dados somente para aqueles endereços de memória que contém dados diferentes dos armazenados. Este processo é mais rápido do que alterar todo o bloco de dados.

Conclui-se então que o cache controla todos os endereços e bloco de instruções necessários para a execução de um programa, tornando dessa forma o computador mais rápido.

A figura abaixo ilustra de forma simplificada a trânsito dessas informações, entre a programa, CPU, cache externo e memória principal:



CACHE DE DISCO:

O ponto mais lento de qualquer computador é a unidade de disco, em virtude de suas partes móveis. As mesmas movem-se dentro de um tempo real, levando-se em conta sua massa e inércia.

Uma das formas de minimizar esse inconveniente é certificar-se de que os arquivos do disco estão desfragmentados, ou seja, os clusters que compõe tais arquivos estejam próximos, evitando assim que as cabeças de leitura/escrita percorram todo o espaço físico do disco para unir as partes do arquivo.

Outra forma é optar por uma unidade de RAM que faz parte do computador, configurada para tapear o computador, simulando assim, uma unidade real.

O funcionamento do cache de disco é similar ao cache de RAM.

Levando-se em conta que, a diferença de velocidade de uma unidade de disco e a velocidade de qualquer chip de RAM, por mais lento que seja, em comparação com a velocidade de um chip de RAM rápido e um mais lento é muito grande, o cache de disco produz resultados mais evidentes.

Existem vários programas de cache de disco disponíveis no mercado; o sistema operacional MS-DOS, por exemplo, inclui um programa desse tipo (Smartdrv).

Algumas controladoras de disco mais modernas, possuem circuitos para unidades de cache de RAM próprios, evitando assim que o cache utilize a memória do computador que seria solicitada pelos programas.

Ciclo operacional:

I - Quando um programa de cache de disco é carregado, o programa residente da memória reserva uma parte da memória convencional, expandida ou estendida para seu uso, podendo atingir de alguns kilobits a vários megabits de RAM;

II - Sob o comando de um aplicativo, a CPU envia um comando para a unidade de disco solicitando dados;

III - O cache lê os dados do disco e busca mais dados, geralmente nos clusters adjacentes. O cache intercepta essa solicitação de dados;

IV - Quando a CPU não está processando instruções, o cache assume o controle para ler mais dados na unidade de disco, que também são guardados na RAM (geralmente os setores próximos aos arquivos já lidos).

O cache possui uma lógica interna que permite descobrir quais blocos tem mais chances de serem solicitados mais tarde pelo aplicativo.

A inteligência dessa lógica é que permite diferenciar a eficiência de cada um desses programas;

V - Posteriormente, quando o aplicativo solicitar mais dados, o cache verifica se os dados solicitados estão disponíveis na RAM. Se estiverem, esses dados serão fornecidos diretamente à CPU, sem ter que acessar a unidade de disco, caso contrário, o cache repete os procedimentos anteriores.

Quando a RAM utilizada pelo cache estiver lotada, os dados lá armazenados há mais tempo sem utilização são liberados, sendo substituídos pelos dados mais recentes provenientes do acesso à unidade de disco;

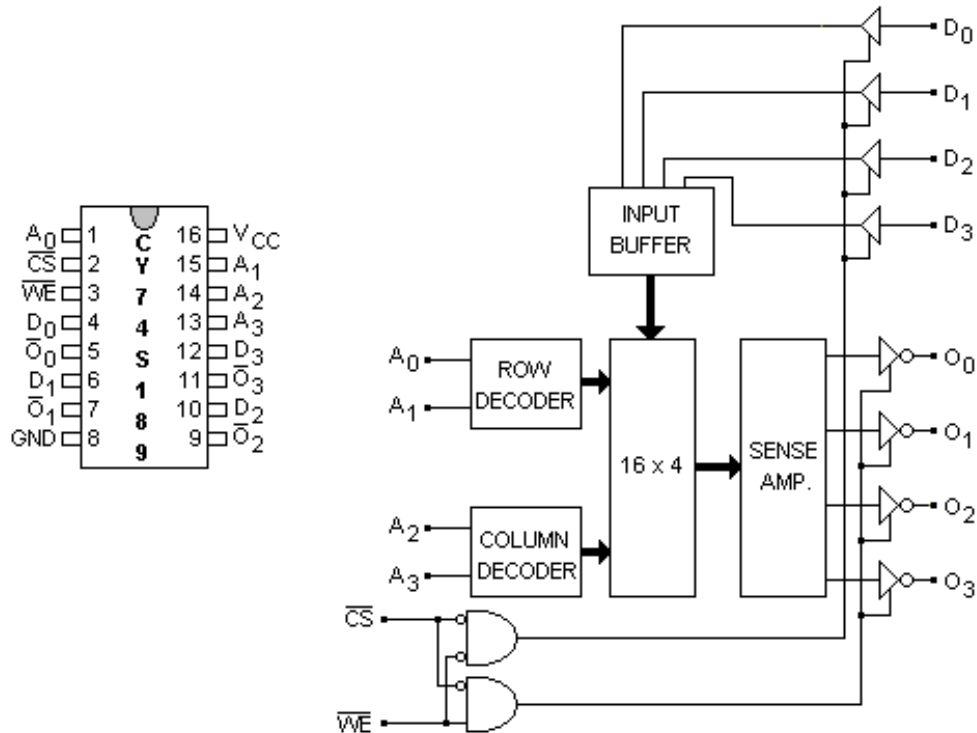
VI - Quando um programa envia um comando para gravar no disco, alguns caches interceptam os dados, não os gravando no disco até que a CPU esteja ociosa, tornando assim, essa operação mais ágil visto que, a CPU não executa operações de qualquer tipo de processamento e gravações em disco;

VII - Se o arquivo a ser gravado ainda estiver na RAM controlado pelo cache, este grava na unidade de disco somente os clusters que foram alterados, minimizando assim, o movimento das cabeças de leitura/escrita.

PARTE PRÁTICA

Utilização do chip comercial CY74S189 (74189 / SN7489) - 16 x 4 STATIC R/W RAM.

A figura abaixo ilustra o diagrama de pinagem e o diagrama lógico:



O chip CY74S189 é fabricado segundo a tecnologia CMOS, com um tempo de acesso da ordem de 35ns.

As saídas, que correspondem aos pinos 5,7,9 e 11 estão em *coletor aberto*.

As entradas correspondem aos pinos 4,6,10 e 12.

Os pinos de endereçamento são: 1,15,14 e 13 utilizados para selecionar 1 entre 16 palavras.

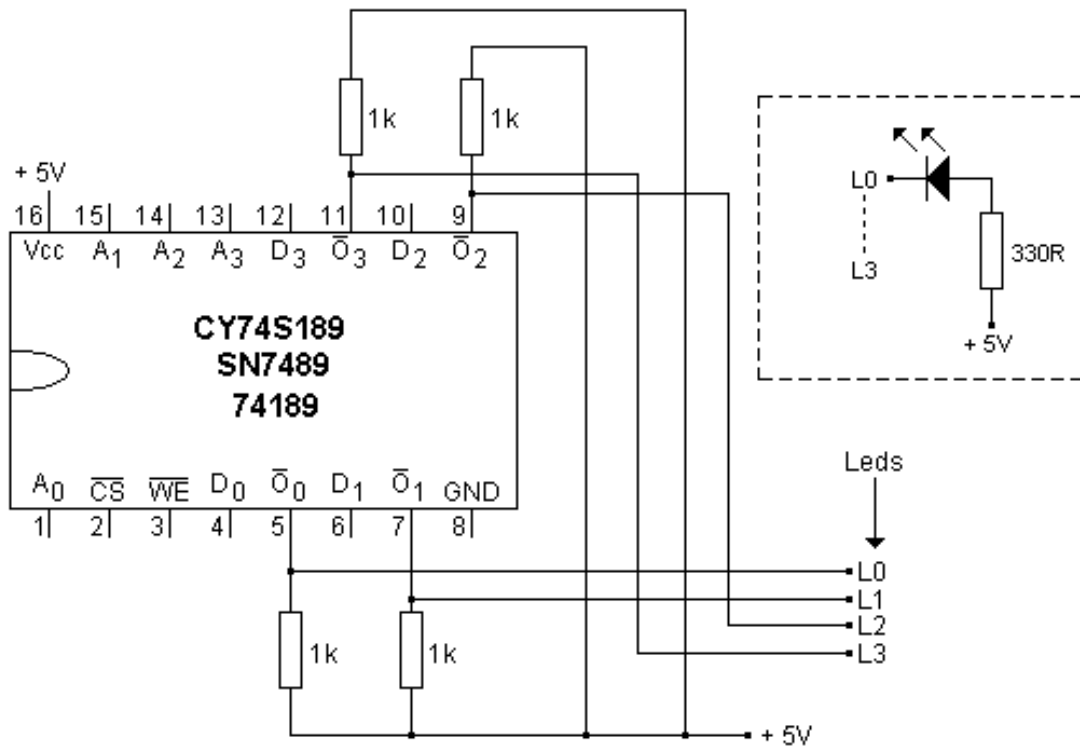
As linhas de controle são: pino 2 (chip select) e pino 3 (write enable), cujas possibilidades de operação são mostradas na tabela a seguir:

CS'	WE'	OPERAÇÃO
0	0	Escrita
0	1	Leitura
1	0	Armazenamento inibido
1	1	Não opera (HI-Z)

MATERIAIS NECESSÁRIOS

- 1 - chip CY74S189 ou SN74189 ou SN7489
- 4 - Leds vermelhos
- 4 - Resistores de 1k Ω - 1/4W
- 4 - Resistores de 330 Ω - 1/4W
- 1 - Fonte de alimentação 5V
- 1 - Multímetro analógico ou digital
- 1 - Placa tipo Proto-board

1 - Conectar o chip ao proto-board, conforme indica a figura abaixo:



2 - Complete a tabela 1 a seguir - OPERAÇÃO ESCRITA:

Controle		Endereço				Palavra de entrada				Palavra de saída			
CS'	WE'	A ₃	A ₂	A ₁	A ₀	D ₃	D ₂	D ₁	D ₀	O' ₃	O' ₂	O' ₁	O' ₀
		1	1	0	1	1	0	0	0				
		0	0	0	1	1	1	0	0				
		0	0	1	1	0	0	1	1				
		1	0	1	0	0	1	1	0				
		0	1	1	0	0	1	0	1				
		0	1	0	0	0	0	1	0				
		1	1	0	1	0	1	1	1				
		0	1	1	1	1	0	0	1				

3 - Complete a tabela 2 a seguir - OPERAÇÃO LEITURA:

Controle		Endereço				Palavra de entrada				Palavra de saída			
CS'	WE'	A ₃	A ₂	A ₁	A ₀	D ₃	D ₂	D ₁	D ₀	O' ₃	O' ₂	O' ₁	O' ₀
		1	1	0	1								
		0	0	0	1								
		0	0	1	1								
		1	0	1	0								
		0	1	1	0								
		0	1	0	0								
		1	1	0	1								
		0	1	1	1								

4 - As palavras lidas na tabela 2, correspondem às palavras escritas na tabela 1? Caso isto não ocorra, o que deverá ser feito?

5 - O que ocorre quando o chip opera na condição *não opera*?

6 - O que ocorre quando o chip opera na condição *armazenamento proibido*?

7 - Por qual motivo são ligados os resistores de 1kΩ, do Vcc às saídas O'₀ a O'₃ ?

8 - Sem esses resistores o chip funcionaria? Justifique.
