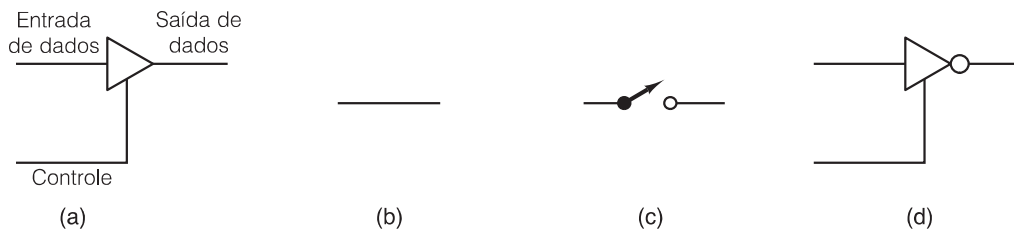


Figura 3.29 (a) Buffer não inversor. (b) Efeito de (a) quando o controle está alto. (c) Efeito de (a) quando o controle está baixo. (d) Buffer inversor.



dispositivos de três estados, porque podem produzir 0, 1, ou nenhum dos dois (circuito aberto). *Buffers* também amplificam sinais, portanto, podem comandar muitas entradas simultaneamente. Às vezes, eles são usados em circuitos por essa razão, mesmo quando suas propriedades de comutação não são necessárias.

Voltando ao circuito de memória, agora já deve estar claro para que servem os três *buffers* não inversores nas linhas de saída de dados. Quando CS, RD e OE estiverem todos altos, o sinal *output enable* também está alto, habilitando os *buffers* e colocando uma palavra nas linhas de saída. Quando qualquer um dos CS, RD ou OE estiver baixo, as saídas de dados são desconectadas do resto do circuito.

3.3.5 Chips de memória

O bom da memória da Figura 3.28 é que ela pode ser ampliada com facilidade para tamanhos maiores. Em nosso desenho, a memória é 4×3 , isto é, quatro palavras de 3 bits cada. Para ampliá-la para 4×8 , basta adicionar cinco colunas de quatro *flip-flops* cada, bem como cinco linhas de entrada e cinco linhas de saída. Para passar de 4×3 para 8×3 , devemos acrescentar quatro linhas de três *flip-flops* cada, bem como uma linha de endereço A_2 . Com esse tipo de estrutura, o número de palavras na memória deve ser uma potência de 2 para que haja o máximo de eficiência, mas o número de bits em uma palavra pode ser qualquer um.

Como a tecnologia de circuitos integrados se ajusta bem à fabricação de chips cuja **estrutura interna é um padrão bidimensional repetitivo, chips de memória são uma aplicação ideal para ela**. À medida que a tecnologia melhora, o número de bits que podem ser colocados em um chip continua crescendo, normalmente por um fator de dois a cada 18 meses (lei de Moore). Os chips maiores nem sempre tornam os menores obsoletos devido aos diferentes compromissos entre capacidade, velocidade, energia, preço e conveniência da interface. Em geral, os chips maiores disponíveis no momento são vendidos por preços mais elevados, portanto, são mais caros por bit do que os antigos, menores.

Há vários modos de organizar o chip para qualquer tamanho de memória dado. A Figura 3.30 mostra duas organizações possíveis para um chip de memória mais antigo de 4 Mbits de tamanho: $512\text{ K} \times 8$ e $4.096\text{ K} \times 1$. (A propósito, os tamanhos de chips de memória costumam ser citados em bits em vez de bytes, e por isso adotaremos essa convenção.) Na Figura 3.30(a), são necessárias 19 linhas de endereço para endereçar um dos 2^{19} bytes e oito linhas de dados para carregar e armazenar o byte selecionado.

Cabe aqui uma observação sobre tecnologia. Em alguns pinos, a alta tensão provoca uma ação. Em outros, é a baixa tensão que causa uma ação. Para evitar confusão, preferimos manter a coerência e dizer sempre que **o sinal é afirmado (em vez de dizer que fica alto ou baixo)**, o que significa que **foi disparado para provocar alguma ação**. Assim, para alguns pinos, afirmá-lo significa estabelecê-lo alto. Para outros, significa estabelecer o pino baixo. Os nomes de sinais de pinos afirmados baixos são distinguidos por uma barra superior. Assim, um sinal com rótulo $\overline{\text{cs}}$ é ativado alto, mas um sinal com rótulo $\overline{\text{cs}}$ é ativado baixo. **O oposto de afirmado é negado. Quando nada de especial estiver acontecendo, os pinos são negados.**

Agora, vamos voltar ao nosso chip de memória. Uma vez que um computador costuma ter muitos chips de memória, é preciso um sinal para selecionar o chip necessário no momento em questão, de modo que ele responda e todos os outros não. O sinal \overline{CS} (*chip select* – seleção de chip) existe para essa finalidade e é ativado para habilitar o chip. Além disso, é preciso uma maneira de distinguir entre leituras e escritas. O sinal \overline{WE} (*write enable* – habilitar escrita) é usado para indicar que os dados estão sendo escritos, e não lidos. Por fim, o sinal \overline{OE} (*output enable* – habilitar saída) é afirmado para comandar os sinais de saída. Quando ele não é afirmado, a saída do chip é desconectada do circuito.

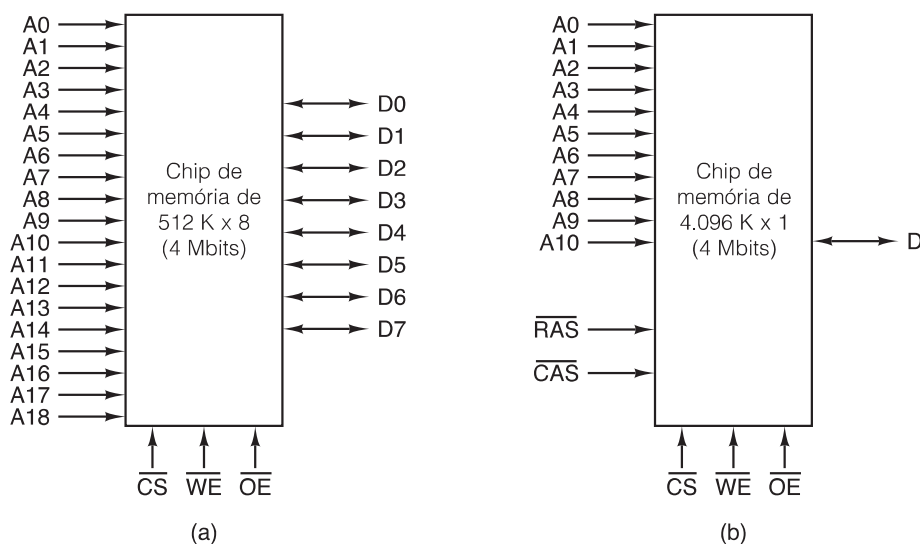
Na Figura 3.30(b), é usado um esquema de endereçamento diferente. Esse chip é organizado internamente como uma matriz 2.048×2.048 de células de 1 bit, o que dá 4 Mbits. Para endereçar o chip, em primeiro lugar uma linha é selecionada ao se colocar seu número de 11 bits nos pinos de endereço. Então o \overline{RAS} (*row address strobe* – *strobe* de endereço de linha) é afirmado. Em seguida, um número de coluna é colocado nos pinos de endereço e o \overline{CAS} (*column address strobe* – *strobe* de endereço de coluna) é afirmado. O chip responde aceitando ou entregando um bit de dados.

Chips de memória de grande porte costumam ser construídos como matrizes $n \times n$ endereçadas por linha e coluna. Essa organização reduz o número de pinos requerido, mas também torna mais lento o endereçamento do chip, já que são necessários dois ciclos, um para a linha e outro para a coluna. Para recuperar um pouco da velocidade perdida por esse projeto, alguns chips de memória podem receber um endereço de linha acompanhado por uma sequência de endereços de coluna para acessar bits consecutivos em uma linha.

Anos atrás, os maiores chips de memória costumavam ser organizados como os da Figura 3.30(b). À medida que as palavras de memória cresciam de 8 bits até 32 bits e mais, os chips de 1 bit começaram a ser inconvenientes. Construir uma memória com uma palavra de 32 bits usando chips de $4.096 \text{ K} \times 1$ requer 32 chips em paralelo. Esses 32 chips têm capacidade total de no mínimo 16 MB, ao passo que usar chips de $512 \text{ K} \times 8$ requer somente quatro chips em paralelo e permite memórias pequenas, de até 2 MB. Para evitar ter 32 chips para memória, grande parte dos fabricantes lançou famílias com 4, 8 e 16 bits de largura. A situação com as palavras de 64 bits é pior ainda, é claro.

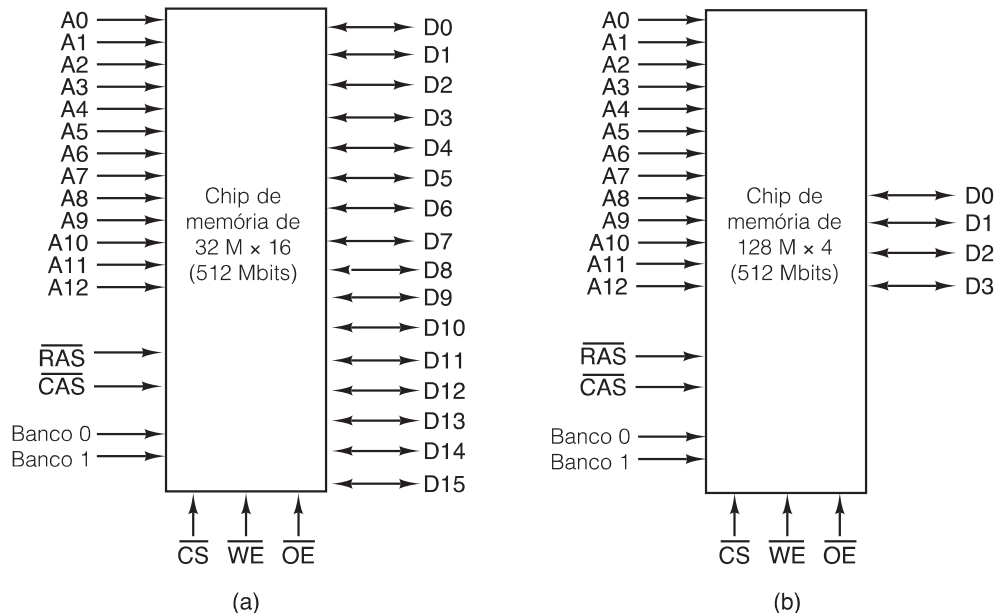
Dois exemplos de chips modernos de 512 Mbits são dados na Figura 3.31. Esses chips têm quatro bancos de memória internos de 128 Mbits cada, o que requer duas linhas de seleção de banco para escolher um banco. O projeto da Figura 3.31(a) é de um chip de $32 \text{ M} \times 16$ com 13 linhas para o sinal \overline{RAS} , 10 linhas para o sinal \overline{CAS} e 2 linhas para a seleção de banco. Juntos, esses 25 sinais permitem o endereçamento de cada uma das 2^{25} células

Figura 3.30 Dois modos de organizar um chip de memória de 4 Mbits.



internas de 16 bits. Em comparação, a Figura 3.31(b) apresenta um projeto de $128\text{ M} \times 4$ com 13 linhas para o sinal $\overline{\text{RAS}}$, 12 linhas para o sinal $\overline{\text{CAS}}$ e 2 linhas para a seleção de banco. Nesse caso, 27 sinais podem selecionar quaisquer das 2^{27} células internas de 4 bits a serem endereçadas. A decisão sobre o número de linhas e de colunas que um chip tem é tomada por razões de engenharia. A matriz não precisa ser quadrada.

Figura 3.31 Dois modos de organizar um chip de memória de 512 Mbits.



Esses exemplos demonstram duas questões separadas e independentes para o projeto do chip de memória. A primeira é a largura da saída (em bits): o chip entrega 1, 4, 8, 16 ou algum outro número de bits de uma vez só? A segunda é se todos os bits de endereço são apresentados em pinos separados de uma vez só ou se as linhas e colunas são apresentadas em sequência, como nos exemplos da Figura 3.31. Um projetista de chips de memória tem de responder a ambas as perguntas antes de iniciar o projeto do chip.

3.3.6 RAMs e ROMs

Todas as memórias que estudamos até aqui **podem ser escritas e lidas**. Elas são denominadas memórias **RAM** (**Random Access Memory** — **memória de acesso aleatório**), um nome suspeito porque todos os chips de memória têm acesso aleatório. No entanto, o termo já é muito utilizado para que o mudemos agora. **RAMs podem ser de duas variedades, estáticas e dinâmicas**. Nas **estáticas (Static RAMs — SRAMs)**, a construção interna usa **circuitos similares ao nosso flip-flop D básico**. Uma das propriedades dessas memórias é que **seus conteúdos são conservados enquanto houver fornecimento de energia**: segundos, minutos, horas e até mesmo dias. **As RAMs estáticas são muito rápidas**. Um tempo de acesso típico é da ordem de um nanossegundo ou menos. Por essa razão, elas são **muito usadas como memória cache**.

RAMs dinâmicas (Dynamic RAMs — DRAMs), ao contrário, **não usam flip-flops**. Em vez disso, uma RAM dinâmica **é um arranjo de células, cada uma contendo um transistor e um pequenino capacitor**. Os **capacitores podem ser carregados ou descarregados, permitindo que 0s e 1s sejam armazenados**. Como a carga elétrica tende a vaziar, cada bit em uma RAM dinâmica deve ser renovado (recarregado) com alguns milissegundos de intervalo para evitar que os dados desapareçam. Como a lógica externa é que tem de cuidar da renovação, as RAMs dinâmicas precisam de uma interface mais complexa do que as estáticas, embora em muitas aplicações essa desvantagem seja compensada por suas maiores capacidades.

Visto que as RAMs dinâmicas precisam de apenas um transistor e um capacitor por bit, em comparação com os seis transistores por bit para a melhor RAM estática, elas têm densidade muito alta (muitos bits por chip). Por essa razão, as memórias principais quase sempre são construídas com RAMs dinâmicas. Contudo, essa grande capacidade tem um preço: são lentas (dezenas de nanossegundos). Dessa maneira, a combinação de uma cache de RAM estática e uma memória principal de RAM dinâmica tenta combinar as boas propriedades de cada uma.

Existem diversos tipos de RAMs dinâmicas. A mais antiga ainda existente (em computadores antigos) é a DRAM FPM (Fast Page Mode – modo de página rápida). Ela é organizada internamente como uma matriz de bits e funciona da seguinte maneira: o hardware escolhe um endereço de linha e então seleciona endereços de coluna um a um, como descrevemos para o \overline{RAS} e o \overline{CAS} no contexto da Figura 3.30. Sinais explícitos informam à memória quando é hora de responder, de modo que ela funciona de forma assíncrona com o *clock* do sistema principal.

A DRAM FPM foi substituída pela EDO (Extended Data Output – saída de dados ampliada), que permite iniciar uma segunda referência à memória antes de ser concluída a referência à memória precedente. Esse paralelismo simples não acelerava uma referência individual à memória, mas melhorava a largura de banda da memória, resultando em mais palavras por segundo.

FPM e EDO funcionavam bastante bem quando os tempos de ciclo de chips de memória eram de 12 nanossegundos ou mais lentos. Quando os processadores ficaram tão rápidos que era mesmo preciso ter memórias mais rápidas, a FPM e a EDO foram substituídas pela SDRAM (Synchronous DRAM – DRAM síncrona), que é uma híbrida de RAM estática e dinâmica, comandada pelo *clock* do sistema principal. A grande vantagem da SDRAM é que o *clock* elimina a necessidade de sinais de controle para informar ao chip de memória quando responder. Em vez disso, a CPU informa à memória por quantos ciclos ela deve funcionar e então a inicia. Em cada ciclo subsequente, a memória entrega 4, 8 ou 16 bits, dependendo de quantas linhas de saída ela tem. Eliminar a necessidade de sinais de controle aumenta a taxa de dados entre CPU e memória.

A melhoria seguinte em relação à SDRAM foi a SDRAM DDR (Double Data Rate – dupla taxa de dados). Com esse tipo de memória, o chip de memória produz saída na borda ascendente do *clock* e também na borda descendente, dobrando a taxa de dados. Portanto, um chip DDR de 8 bits de largura funcionando a 200 MHz entrega dois valores de 8 bits 200 milhões de vezes por segundo (por um curto intervalo, é claro), o que dá uma taxa de saída (*burst*) teórica de 3,2 Gbps. As interfaces de memória DDR2 e DDR3 oferecem desempenho adicional em relação à DDR, aumentando as velocidades do barramento de memória para 533 MHz e 1.067 MHz, respectivamente. No momento em que este livro era impresso, os chips DDR3 mais velozes poderiam enviar dados a 17,067 GB/s.

● Chips de memória não volátil

RAMs não são o único tipo de chip de memória. Em muitas aplicações, como brinquedos, eletrodomésticos e carros, o programa e alguns dos dados devem permanecer armazenados mesmo quando o fornecimento de energia for interrompido. Além do mais, uma vez instalados, nem o programa nem os dados são alterados. Esses requisitos levaram ao desenvolvimento de **ROMs (Read-Only Memories – memórias somente de leitura)**, que **não podem ser alteradas nem apagadas**, seja intencionalmente ou não. Os dados de uma ROM **são inseridos durante sua fabricação por um processo que expõe um material fotossensível por meio de uma máscara que contém o padrão de bits desejado e então grava o padrão sobre a superfície exposta (ou não exposta)**. A única maneira de mudar o programa em uma ROM é **substituir o chip inteiro**.

ROMs são muito mais baratas que RAMs quando fabricadas em volumes grandes o bastante para cobrir o custo da fabricação da máscara. Todavia, são inflexíveis porque não podem ser alteradas após a manufatura, e o tempo decorrido entre fazer o pedido e receber as ROMs pode chegar a semanas. Para facilitar o desenvolvimento pelas empresas de novos produtos com ROM, foi inventada a **PROM (Programmable ROM – ROM programável)**. Uma PROM é como uma ROM, exceto que ela **pode ser programada (uma vez) em campo, eliminando o tempo de espera entre produção e entrega**. Muitas PROMs contêm um arranjo de minúsculos fusíveis em seu interior. Um fusível específico pode ser queimado selecionando sua linha e coluna e então aplicando alta tensão a um pino especial no chip.

O desenvolvimento seguinte nessa linha foi a **EPROM (Erasable PROM – PROM apagável)**, que não só **pode ser programada, mas também apagada em campo**. Quando a janela de quartzo de uma EPROM é exposta a uma forte luz ultravioleta durante 15 minutos, todos os bits são definidos em 1. Se a expectativa é ter muitas alterações durante o ciclo de projeto, as EPROMs são muito mais econômicas do que as PROMs, porque **podem ser reutilizadas**. As EPROMs costumam ter a mesma organização que as RAMs estáticas. A EPROM 27C040 de 4 Mbits, por exemplo, usa a organização da Figura 3.31(a), que é típica de uma RAM estática. O interessante é que chips antigos como este não desaparecem. Eles apenas se tornam mais baratos e são usados em produtos inferiores, que são altamente sensíveis ao custo. Um 27C040 agora pode ser comprado no varejo por menos de US\$ 3, e por muito menos em grandes volumes.

Ainda melhor do que a EPROM é a **EEPROM**, que **pode ser apagada aplicando-se pulsos em vez de ser exposta à luz ultravioleta dentro de uma câmara especial**. Além disso, uma EEPROM **pode ser reprogramada no local**, enquanto uma EPROM tem de ser inserida em um dispositivo especial de programação de EPROM para ser programada. Uma desvantagem é que a capacidade das maiores EEPROMs é em geral somente 1/64 da capacidade das EPROMs comuns, e sua velocidade é a metade. EEPROMs não podem competir com DRAMs ou SRAMs porque são 10 vezes mais lentas, sua capacidade é 100 vezes menor e são muito mais caras. Elas são usadas somente em situações em que sua não volatilidade for crucial.

Um tipo mais recente de **EEPROM é a memória flash**. Diferente da EPROM, que é apagada pela exposição à luz ultravioleta, e da EEPROM, cujos bytes podem ser apagados, **os blocos da memória flash podem ser apagados e reescritos**. Como a EEPROM, a memória *flash* pode ser apagada sem ser removida do circuito. Vários fabricantes produzem pequenas placas de circuito impresso com até 64 GB de memória *flash* que são utilizadas como um “filme” para armazenar fotos em câmeras digitais e muitas outras finalidades. Como já vimos no Capítulo 2, a memória *flash* agora está começando a substituir os discos mecânicos. Assim como um disco, a memória *flash* oferece tempos de acesso menores com menor consumo de energia, mas com um custo por bit muito mais alto. **Um resumo dos diversos tipos de memória pode ser visto na Figura 3.32.**

Figura 3.32 Comparação entre vários tipos de memórias (Arranjo de portas programável em campo).

Tipo	Categoria	Modo de apagar	Byte alterável	Volátil	Utilização típica
SRAM	Leitura/escrita	Elétrico	Sim	Sim	Cache de nível 2
DRAM	Leitura/escrita	Elétrico	Sim	Sim	Memória principal (antiga)
SDRAM	Leitura/escrita	Elétrico	Sim	Sim	Memória principal (nova)
ROM	Somente de leitura	Não é possível	Não	Não	Equipamentos de grande volume
PROM	Somente de leitura	Não é possível	Não	Não	Equipamentos de pequeno volume
EPROM	Principalmente leitura	Luz UV	Não	Não	Prototipagem de dispositivos
EEPROM	Principalmente leitura	Elétrico	Sim	Não	Prototipagem de dispositivos
Flash	Leitura/escrita	Elétrico	Não	Não	Filme para câmera digital

Field-programmable gate arrays

Como vimos no Capítulo 1, **Field-Programmable Gate Arrays (FPGAs)** são chips que **contêm lógica programável, de modo que podem formar um circuito lógico qualquer simplesmente carregando o FPGA com dados de configuração apropriados**. A principal vantagem dos FPGAs é que novos circuitos de hardware podem ser construídos em horas, em vez dos meses necessários para fabricar ICs. Porém, os circuitos integrados não serão extintos, pois ainda possuem uma vantagem de custo significativa em relação aos FPGAs para aplicações de alto volume, e também são mais rápidos e usam muito menos energia. Contudo, com suas vantagens de tempo de projeto, os FPGAs são usados constantemente para protótipo de projeto e aplicações com baixo volume.