



Centro de Formação Profissional “Aloysio Ribeiro de Almeida”

ELETRÔNICA

II

FIEMG

CIEMG

SESI

SENAI

IEL

Sistema FIEMG



Presidente da FIEMG

Robson Braga de Andrade

Gestor do SENAI

Petrônio Machado Zica

Diretor Regional do SENAI e

Superintendente de Conhecimento e Tecnologia

Alexandre Magno Leão dos Santos

Gerente de Educação e Tecnologia

Edmar Fernando de Alcântara

Elaboração

Vanderlei Batista Flausino

Unidade Operacional

Centro de Formação Profissional “Aloysio Ribeiro de Almeida”
Varginha – MG
2004.

FIEMG

CIEMG

SESI

SENAI

IEL



Sumário

APRESENTAÇÃO	4
TEMPORIZADOR 555	5
CODIFICADORES E DECODIFICADORES	8
CI'S DECODIFICADORES TTL	11
MULTIVIBRADORES BIESTÁVEIS (FLIP-FLOP).....	12
CI'S DIGITAIS COM FLIP-FLOPS	18
REGISTRADORES DE DESLOCAMENTO (SHIFT REGISTER)	20
CONTADORES.....	23
TRABALHO SEMESTRAL.....	29
TIPOS DE MEMÓRIAS SEMICONDUTORAS.....	30
REFERÊNCIAS BIBLIOGRÁFICAS	40

FIEMG

CIEMG

SESI

SENAI

IEL

Apresentação

“Muda a forma de trabalhar, agir, sentir, pensar na chamada sociedade do conhecimento.”
Peter Drucker

O ingresso na sociedade da informação exige mudanças profundas em todos os perfis profissionais, especialmente naqueles diretamente envolvidos na produção, coleta, disseminação e uso da informação.

O **SENAI**, maior rede privada de educação profissional do país, sabe disso, e, consciente do seu papel formativo, educa o trabalhador sob a égide do conceito da competência: *“formar o profissional com responsabilidade no processo produtivo, com iniciativa na resolução de problemas, com conhecimentos técnicos aprofundados, flexibilidade e criatividade, empreendedorismo e consciência da necessidade de educação continuada.”*

Vivemos numa sociedade da informação. O conhecimento, na sua área tecnológica, amplia-se e se multiplica a cada dia. Uma constante atualização se faz necessária. Para o **SENAI**, cuidar do seu acervo bibliográfico, da sua infraestrutura, da conexão de suas escolas à rede mundial de informações – internet – é tão importante quanto zelar pela produção de material didático.

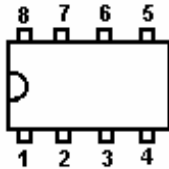
Isto porque, nos embates diários, instrutores e alunos, nas diversas oficinas e laboratórios do **SENAI**, fazem com que as informações, contidas nos materiais didáticos, tomem sentido e se concretizem em múltiplos conhecimentos.

O **SENAI** deseja, por meio dos diversos materiais didáticos, aguçar a sua curiosidade, responder às suas demandas de informações e construir *links* entre os diversos conhecimentos, tão importantes para sua formação continuada!

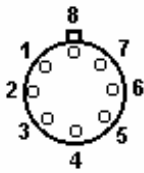
Gerência de Educação e Tecnologia

Temporizador 555

O encapsulamento mais usual em que se apresenta este integrado obedece à disposição duplo em linha , com apenas oito pinos . A figura mostra a pinagem de dois dos tipos de encapsulamento deste dispositivo.



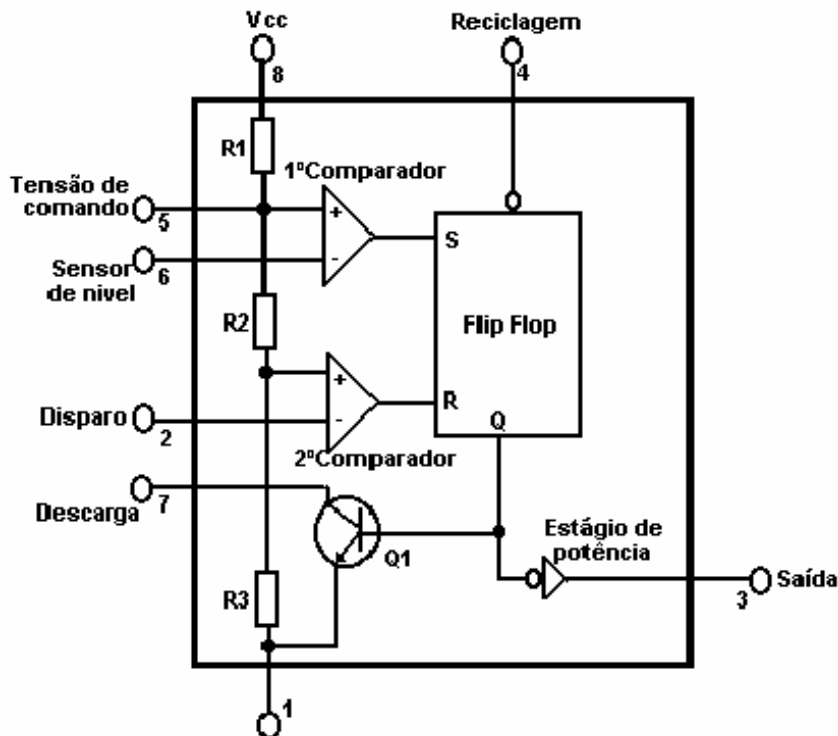
Ligações dos pinos C.I 555, vista de cima ,
encapsulamento tipo duplo em linha



Ligações dos pinos C.I 555, vista de cima ,
encapsulamento tipo Metálico

A principal característica deste integrado é a estabilidade , é pouco sensível a variação de temperatura e tensão de alimentação .essa pode se estender desde 4.5 volts até 18 Volts . Este C.I pode manipular diretamente , sob certas condições , cargas maiores que 100mA.

Devido às características acima , este C.i é compatível com a lógica TTL e MOS. Observe o diagrama em blocos do timer 555



Descrição Do Circuito Integrada 555

Os circuitos 555, são circuitos monolíticos projetados para funcionarem como temporizadores de precisão.

Estes circuitos integrados são capazes de produzir atrasos de tempo e oscilações. Os C.I 555, operam nos modos astável e/ou monoestável.

No modo monoestável, o tempo é controlado por uma simples malha composta de um resistor e um capacitor, conectados externamente ao C.I.

No modo astável, a frequência e o período de operação pode ser controlados independentemente por meio de dois resistores e um capacitor, conectados externamente ao C.I.

Descrição dos Pinos

Pinos 1 e 8- Alimentação

Destina-se a alimentação do C.I, que pode variar de 4.5 a 15V

Pino 2-Disparo

Este pino possibilita situar o C.I, no ciclo ativo, toda vez que nele for aplicado uma tensão inferior que a terça parte de VCC.

Pino 3 –Saída

Quando este está ativo, assume nível próximo de +VCC, quando em repouso o nível é praticamente nulo 0V.

Pino 4- Reciclagem

Quando nesta for aplicado nível baixo, a saída irá para o repulso.

Pino 5- Tensão de controle

Esta entrada é destinada a realizar desacoplamentos dos resistores internos, permite a alteração dos níveis de referência em tensão para cada um dos comparadores, possibilitando a modulação de sinais.

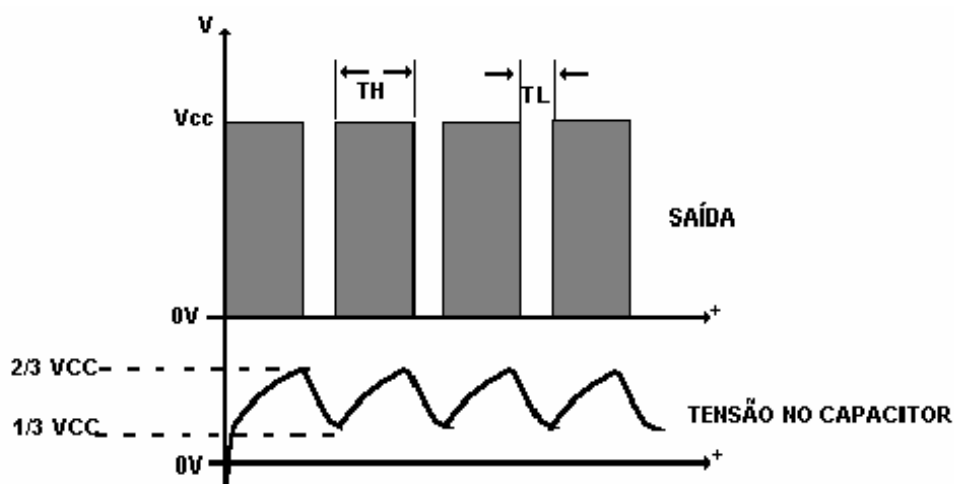
Pino 6- Sensor de Nível

Esta entrada, recicla o C.I, quando a tensão aplicada a ele for superior a dois terços de VCC.

Pino 7 –Descarga

Este pino é uma saída tipo coletor aberto, o qual acompanha as variações do pino 3.

Formas de Ondas Características



Formulas:

Duty cycle.

$$D = \frac{TL}{TH + TL} = \frac{RB}{RA + 2RB}$$

Tempo de saída em alta (TH)

$$TH = \ln 2(RA + RB)C$$

Tempo de saída em repouso (TL)

$$TL = \ln 2(RB)C$$

Tempo total (Período)

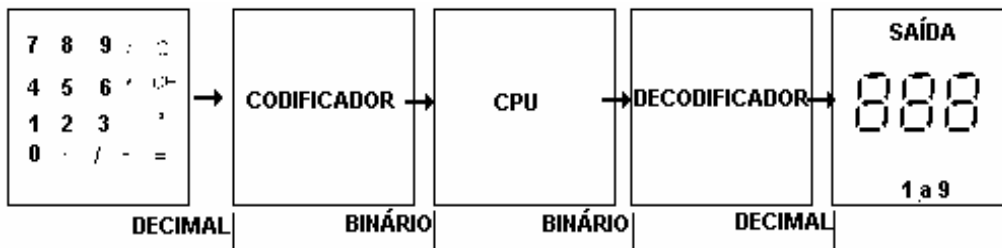
$$T = \ln 2(RA + 2RB)C$$

Frequência (f)

$$f = \frac{1}{T} = \frac{1}{\ln 2(RA + 2RB)C} = \frac{1.44}{(RA + 2RB)C}$$

Codificadores e Decodificadores

Os códigos mais usuais em circuitos lógicos são o binário, BCD (8421), o hexadecimal e o decimal. Além destes, há códigos cuja aplicação é vantajosa em relação a outros, em algumas aplicações. Considere o diagrama de blocos de uma calculadora manual apresentado.



Na figura acima, o codificador é o dispositivo que traduz o número digitado no teclado em um código binário, tal como o BCD. Pode-se dizer que ele codifica uma linguagem entendida por pessoas para uma linguagem entendida pela calculadora. Após ser realizada a operação matemática, os resultados são disponibilizados em um código binário. O decodificador é o elemento que traduz o número binário de saída para um display de sete segmentos. Pode-se dizer que ele decodifica a linguagem da calculadora para uma linguagem facilmente identificada por nós.

Codificadores

Chama-se de codificador o circuito combinacional que torna possível a passagem de um código conhecido para um desconhecido.

Um codificador executa função inversa à do decodificador. As entradas do codificador, muitas vezes, são as saídas de um decodificador. Para cada linha de entrada escolhida, uma palavra de código correspondente, com bits A, B, C ... N, aparece nas linhas de saída. Em geral, não é necessário que exista uma relação especial entre o número de linhas de entrada e o número de linhas de saída.

Codificador Decimal / Binário

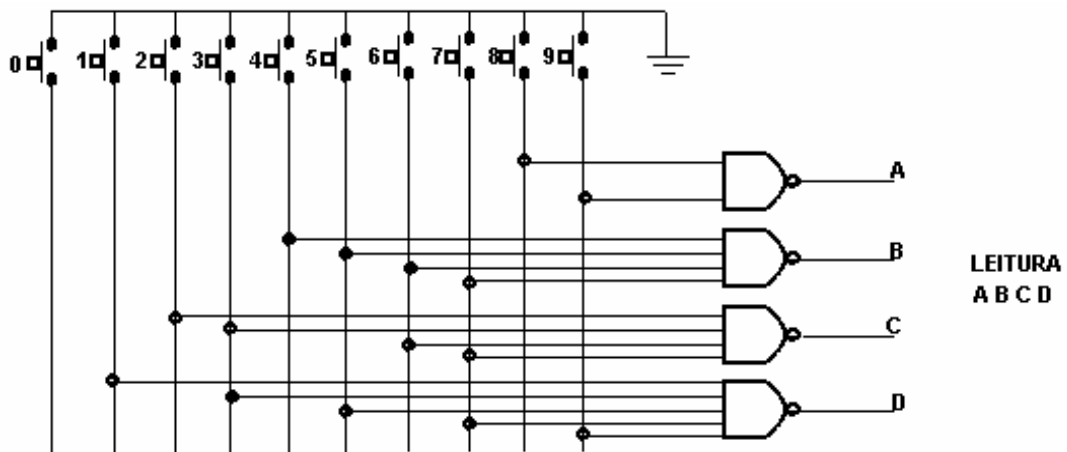
Para transformar um código decimal em um código binário, a entrada decimal será feita através de um conjunto de chaves numeradas de 1 a 9 e a saída por quatro fios, para fornecer o código binário de quatro bits.



A seguir vamos construir a tabela da verdade do codificador que relaciona cada chave de entrada com a respectiva saída em binário

Chave	A	B	C	D
Ch0	0	0	0	0
Ch1	0	0	0	1
Ch2	0	0	1	0
Ch3	0	0	1	1
Ch4	0	1	0	0
Ch5	0	1	0	1
Ch6	0	1	1	0
Ch7	0	1	1	1
Ch8	1	0	0	0
Ch9	1	0	0	1

Pela tabela da verdade o circuito codificador , é visto abaixo.



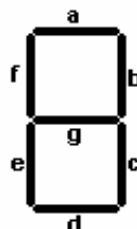
Decodificadores

Chama se de decodificador o circuito com operação inversa ao codificador , ou seja ,passa um código desconhecido para um conhecido.

Para que possamos entender melhor e mais rápido um código binário , basta converte-lo em dígito decimal. Para isto usa se um tipo qualquer de display.

Decodificador para display de sete segmentos.

O display de sete segmentos possibilita escrever números de 0 á 9 e alguns símbolos e letras.



Entre as tecnologias de fabricação das unidades de display , estudaremos a mais comum que é o display a led , este possui cada seguimento um led . Estes display podem ser do tipo cátodo comum ou anodo comum .

O display tipo cátodo comum é aquele que possui todos cátodos dos segmentos (led's) interligados , sendo necessário aplicar nível 1 no anodo respectivo para acender cada seguimento .

O display de anodo comum possui todos os anodos dos seguimentos interligados, sendo necessário aplicar nível 0 ao cátodo respectivo. Para efetuar projetos com este dispositivos , deve se observar a seqüência dos caracteres

CARACTERES	CÓDIGO PARA 7 SEGMENTOS						
	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1

Para alguns tipos de decodificadores, existem versões específicas de CI's. Uma classe importante de decodificadores são os de "n'linhas para 2n linhas (2 para 4, 3 para 8 etc.). Estes decodificadores dispõem de "n'linhas de entrada e 2n linhas de saída, das quais uma ficará ativa para determinada combinação da entrada. De forma similar são feitos os decodificadores de 3 para 8, 4 para 16 etc. Estes decodificadores são utilizados em circuitos de Multiplex e Demultiplex.

Exercício

Com base no projeto do codificado , desenvolva um decodificado de BCD para sete segmentos.

CI's Decodificadores TTL

7422 Decodificador BCD para Decimal

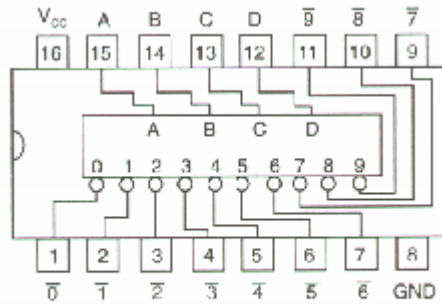


Fig. 9.6

D,C,B,A são entradas BCD (D é o bit mais significativo) 0,1, 2, 3, 4, 5, 6, 7, 8, 9 são saídas ativas em nível lógico zero. Se o número BCD aplicado à entrada for inválido, todas as saídas ficarão em nível lógico "1"

9368 Decodificador binário/hexa para 7 segmentos com driver

SÍMBOLO LÓGICO

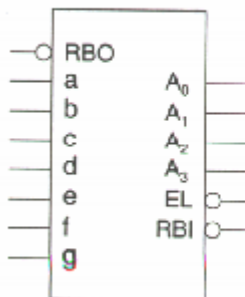
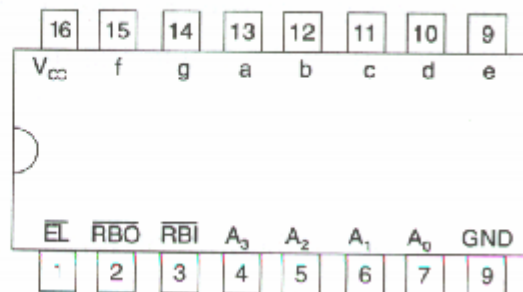


DIAGRAMA DE ENCAPSULAMENTO



O 9368 é um decodificador binário para display de sete segmentos que pode ser conectado diretamente ao display, não necessitando desta forma de resistores de limitação, porque tem driver nas saídas. Este CI decodifica o formato hexadecimal reproduzindo alfanuméricos de 0 a F.

Multivibradores Biestáveis (Flip-Flop)

Introdução

FLIP-FLOP é também chamado de multivibrador BIESTÁVEL, e como possui 2 estados de estabilidade, pode memorizar informações de 1 BIT.

O FLIP-FLOP (abreviado temos FF), são interligados para formar circuitos lógicos para armazenamento, temporização, contagem e sequenciação.

O FLIP-FLOP pode ser classificado, do ponto de vista da função lógica, conforme segue:

- FLIP-FLOP RS (FF - RS)
- FLIP-FLOP RST (FF - RST)
- FLIP-FLOP JK (FF - JK)
- FLIP-FLOP D (D - FF) (FF - D)
- FLIP-FLOP T (T - FF) (FF - T)

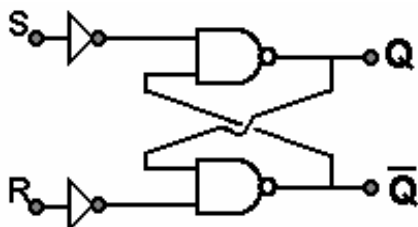
Neste capítulo, iremos estudar o funcionamento de cada um dos tipos mostrados acima.

Flip-Flop R. S

Na figura abaixo apresentamos a estrutura do FLIP-FLOP RS utilizando 2 portas NAND. Como sabemos, a saída de uma porta NAND só será L quando todas as entradas forem H. E ainda, as entradas são S e R, o que significa que este circuito funciona com pulso negativo (ou nível L).

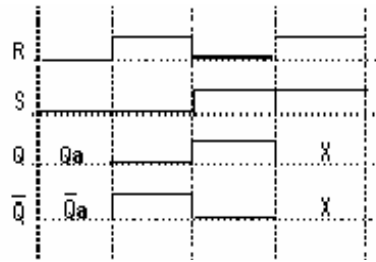
Obs.:

- L = nível lógico 0(zero);
- H = nível lógico 1;
- Q = saída normal;
- Q' = saída complementar.



Vamos, agora, estudar o funcionamento lógico deste circuito observando a tabela da verdade, que mostra o diagrama de tempo, que indica o seu estado de funcionamento.

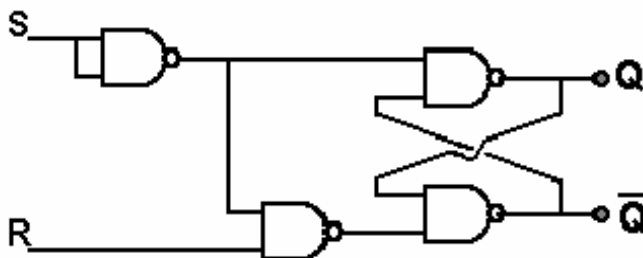
ENTRADA		SAÍDA	
R	S	Q	\bar{Q}
0	0	Q_a	\bar{Q}_a
0	1	1	0
1	0	0	1
1	1	X	X



Quando as entradas S e R estiverem ambas com nível 0, teremos a condição de estabilidade, isto é, a saída manterá o estado anterior. Quando 1 é aplicado à entrada S (e 0 na entrada R), a saída Q passará para 1 (e Q' passará para 0), e o estado de SET será estável. Em seguida, se 1 é aplicado à entrada R (e 0 em S), a saída Q será 0 (e Q' será 1), e o estado RESET será estável. Se for aplicado nível 1 às 2 entradas, teremos a condição de entrada proibida, pois a saída será indeterminada.

Flip-Flop com Preferência Set

O FLIP-FLOP explicado até agora, houve uma condição de entrada proibida que não identificava uma saída estável. A figura abaixo apresenta a estrutura do circuito e a tabela verdade do FLIP-FLOP com preferência SET. Este FLIP-FLOP tem a característica de assumir o estado de SET de saída, quando for aplicado à entrada uma condição de entrada proibida. Isso acontece porque quando é aplicado nível 1 à entrada S, será inserido 0 na porta NAND do lado da entrada R, impedindo que a entrada RESET atue no FLIP-FLOP. Desta forma, haverá uma preferência pelo estado SET.



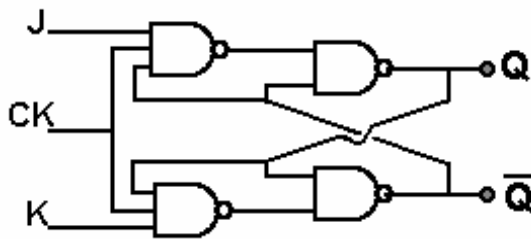
ENTRADA		SAÍDA	
R	S	Q	\bar{Q}
0	0	Q_a	\bar{Q}_a
0	1	1	0
1	0	0	1
1	1	1	0

Flip-Flop Jk (Jk - FF)

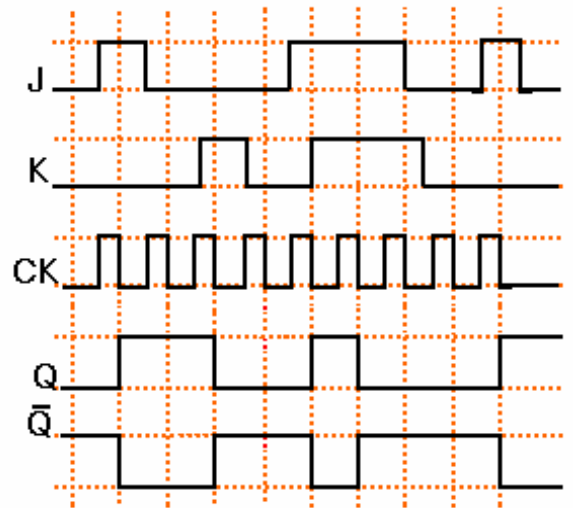
Conforme demonstrado na figura abaixo, o FLIP-FLOP JK é formado pelas entradas J, K e de Clock T (também representada por C, CK ou CP), e as saídas Q e Q'. As entradas J e K correspondem respectivamente às entradas S e R do RS-FF. O FLIP-FLOP JK tem, além da função do RS-FF, a função de inverter o

estado das saídas através do pulso de Clock, quando as 2 entradas estiverem com nível 1. Isto é, não há condição de entrada proibida para o FLIP-FLOP JK. Devido ao fato do FLIP-FLOP JK possuir todas as funções dos outros FLIP-FLOPS é chamado, também de Rei dos FLIP-FLOPS.

O FLIP-FLOP, normalmente comercializado é o tipo TRIGGER EDGE NEGATIVO que funciona na descida do pulso de Clock, conforme indicado na figura . Os que funcionam na subida do pulso de Clock é chamado de TRIGGER EDGE POSITIVO. Vejamos então o seu funcionamento lógico.



ENTRADA			SAÍDA	
R	S	CK	Q	\bar{Q}
0	0		Qa	$\bar{Q}a$
0	1		1	0
1	0		0	1
1	1		\bar{Q}	Q



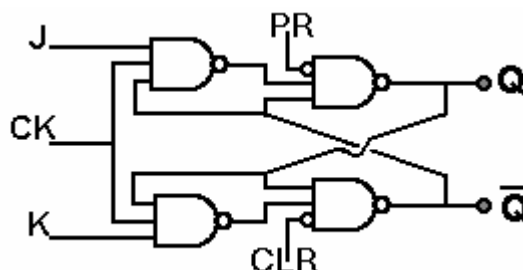
Quando as entradas J e K estiverem com nível 0, e for inserido um pulso de Clock, a saída não se altera, mantendo o estado anterior. Se quando for aplicado um pulso de Clock, a entrada J estiver com 0 e a entrada K estiver com 1, então a saída Q vai para nível 0 e Q' para nível 1.

Se agora, tivermos a entrada J com 1 e a entrada K com 0, quanto for aplicado o pulso de Clock a saída Q vai para nível 1 e a Q' vai para nível 0. Na situação que tivermos as 2 entradas, J e K, com nível 1, cada pulso de Clock aplicado fará com que as saídas invertam de nível, isto é, passem para o estado inverso do estado anterior ao pulso de Clock.

Resumindo o funcionamento lógico do FLIP-FLOP JK, teremos:

- 1) J = L, K = L: mesmo inserindo CK, Q e Q' = invariável
- 2) J = L, K = H: se CK é inserido, Q = L e Q' = H RESET
- 3) J = H, K = L: se CK é inserido, Q = H e Q' = L SET
- 4) J = H, K = H: se CK é inserido, Q e Q' = invertem o estado

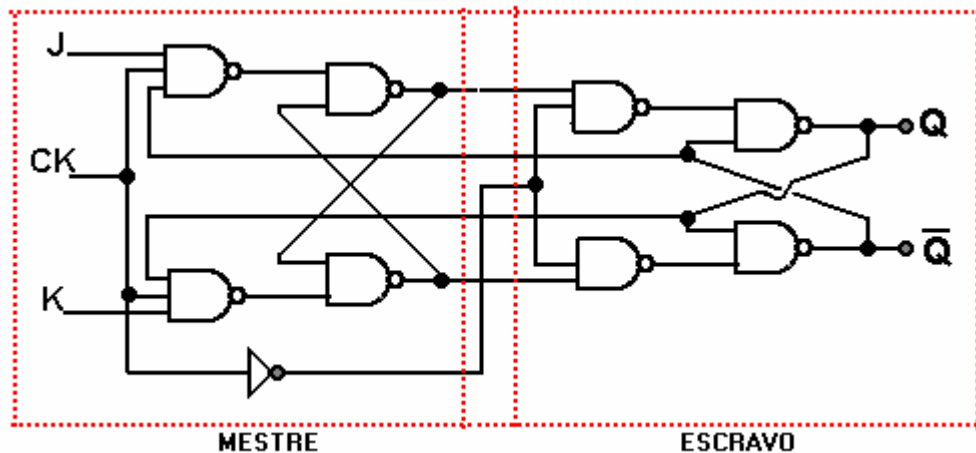
Flip Flop Jk com Entrada de Clear e Preset



O FLIP-FLOP JK possui ainda o terminal de entrada de PRESET (PR), e o terminal de entrada de CLEAR (CLR), conforme mostrado na figura acima. Utiliza-se a representação PRESET (PR) e CLEAR (CLR), ao invés de SET (S) e RESET (R), mas o significado é o mesmo. Isso quer dizer que para que funcione as entradas J, K e CK as entradas PR e CLR devem estar com nível 1.

Flip- Flop Master Slave (Master Slave FF)

O FLIP-FLOP MASTER SLAVE, conforme apresentado na figura , é formado por um FLIP-FLOP MASTER (mestre) e um FLIP-FLOP SLAVE (escravo).



O funcionamento do FLIP-FLOP obedece ao seguinte raciocínio: na subida do pulso de Clock (de 0 para 1) o FLIP-FLOP MASTER lê o sinal de entrada e na descida do Clock (de 1 para 0), o conteúdo da saída do FLIP-FLOP MASTER passa para a saída do FLIP-FLOP SLAVE.

Baseado no diagrama de tempo, vamos agora estudar o funcionamento do FLIP-FLOP MASTER e do FLIP-FLOP SLAVE.

Quando o pulso de Clock T1 é inserido ($J = 1$ e $K = 0$), a entrada CK passa para 1 fazendo com que o circuito NAND mestre fique ON (permite que o sinal das entradas passe). Isso faz que a saída Q do FLIP-FLOP mestre passe de 0 para 1, e Q' de 1 para 0.

Neste instante CK', que é a saída do inversor, está em 0 e as saídas Q e Q' do FLIP-FLOP escravo não se alteram. Quando a entrada CK volta para o nível 0, o circuito NAND mestre passa para OFF, e o circuito NAND escravo passa para ON pois CK' está com nível 1.

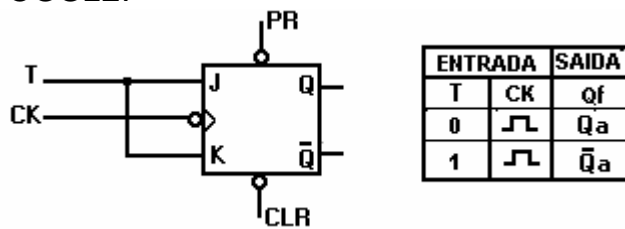
Assim sendo, o conteúdo das saídas do FLIP-FLOP mestre (Q e Q') passam para o FLIP-FLOP escravo fazendo com que Q passe de 0 para 1 e Q' de 1 para 0. Aplicando agora novo pulso de Clock, para as condições de entrada $J = 0$ e $K = 1$, teremos CK = 1 permitindo que o sinal de entrada passe, fazendo com que a saída Q fique com o nível 0 e Q' fique com o nível 1. Nesse instante CK' está com nível 0 o que indica que o FLIP-FLOP escravo mantém a sua saída sem alteração. Assim que a entrada CK volta para o nível 0, o circuito NAND mestre passa para OFF e CK' passa para 1, fazendo com que Qm e Q'm

passem para Qs e Q's. Desta forma, podemos observar que a cada pulso de Clock toda a operação se repete.

Baseado, então, no funcionamento e no diagrama de tempo, podemos afirmar que há um defasamento entre a leitura e memorização da entrada, pelo FLIP-FLOP mestre, e a recepção e saída pelo FLIP-FLOP escravo.

Flip- Flop T (T-FF)

Este Flip Flop é obtido a partir de um JK mestre escravo , com as entradas em curto circuito . O FLIP-FLOP T é chamado de FLIP-FLOP TRIGGER ou FLIP-FLOP TOGGLE.



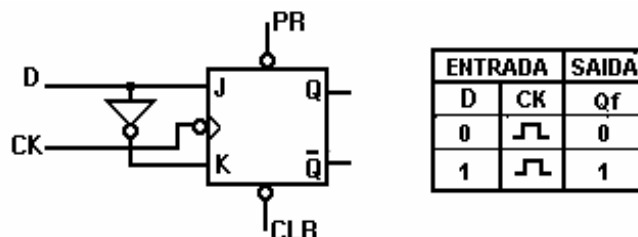
Devido ao fato de o flip flop tipo T , com a entrada T igual a 1 , complementar a saída a cada descida de clock , este será usado como célula principal dos contadores assíncronos.

Vejamos seu funcionamento :

O FLIP-FLOP T inverte o estado de saída Q toda vez que é inserido um pulso de Clock. São necessários 2 pulsos de entrada para se obter 1 pulso na saída. Isso significa que a saída é a metade da freqüência dos pulsos de entrada. Devido a isso, é utilizado no circuito de cálculo e no circuito divisor.

Flip- Flop D (D-FF)

O FLIP-FLOP é do tipo de atraso (Delay) e conforme mostrado na figura , possui a entrada de dados D e a entrada de Clock CK. A saída apresenta um atraso de 1 pulso de Clock em relação à entrada, isto é, o sinal de entrada passa para a saída com certo tempo de atraso (no máximo de 1 ciclo de Clock), por isso é também chamado de FLIP-FLOP do tipo atraso.



Resumindo o funcionamento lógico do FLIP-FLOP D, teremos:

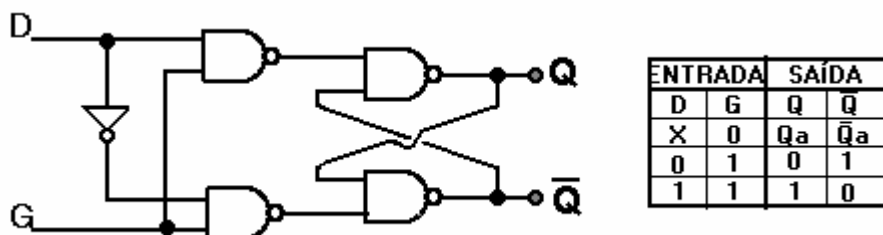
- 1) D = L : se for inserido CK, Q = L e Q' = H
- 2) D = H: se for inserido CK, Q = H e Q' = L

Devido as características do FLIP-FLOP tipo D, de transferir a entrada para a saída com um pulso de Clock, e manter essa saída até o próximo pulso de Clock (memoriza o sinal de entrada), faz com que ele seja utilizado no circuito "LATCH" e em registradores de deslocamento (SHIFT REGISTER), que memorizam por algum tempo a informação da entrada.

Circuito Latch

O circuito "LATCH", é um circuito que memoriza durante determinado tempo, o sinal, ou partes do sinal, de entrada, o que em microcomputação é denominado de fazer "LATCH" de um sinal.

A figura abaixo apresenta o circuito LATCH estruturado com portas NAND e a tabela verdade. O terminal D é o terminal de entrada de dados, e o terminal G é a entrada do sinal STROBE (significa sinal de "pinçamento"), que definirá o tempo de memória do sinal na saída.



Vejamos como funciona este circuito. Quando a entrada G for 0, saída Q mantém seu estado anterior. Se G passar para 1, o sinal de entrada (D) passa para a saída. Parece que o funcionamento é o mesmo do FLIP-FLOP D mas na realidade o D-FF faz a leitura da entrada D na subida do pulso de Clock e esse valor é mantido até que ocorra outra subida do pulso de Clock. Já o "LATCH", lê o sinal de entrada quando G e 1 mantém esse sinal enquanto G for 0.

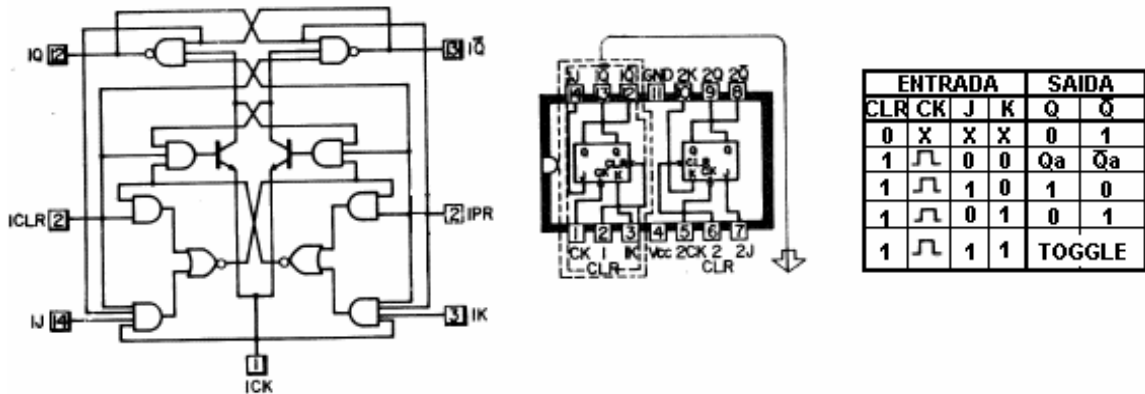
Ci's Digitais com Flip-Flops

Vamos, agora, escolher 2 Ci's da Texas para explicar o funcionamento. O primeiro será o SN 7473 que é um FLIP-FLOP JK MASTER SLAVE e o segundo, o CI _ SN 7474 que é um FLIP-FLOP D com EDGE TRIGGER POSITIVO.

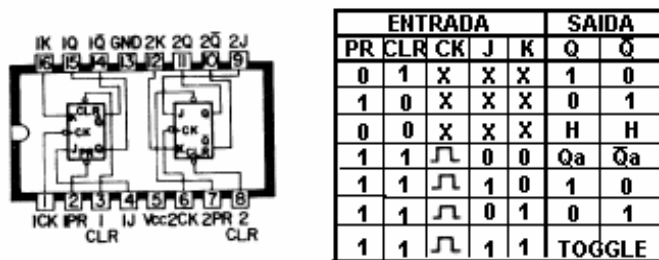
CI 7473 (Dual JK FLIP-FLOP Whith Clear)

Na figura , apresentamos a distribuição dos pinos, a tabela verdade e o circuito equivalente do 7473. Este CI é um FLIP-FLOP JK MASTER SLAVE com entrada de CLEAR. Se a entrada CLR estiver com nível L, a saída Q passa para 0 e Q' para 1, independente das entradas J, K e Ck. Com a entrada CLR em nível 1, o FLIP-FLOP JK funciona como já foi explicado anteriormente.

Dentre os TTL que podem substituir o 7473, temos o M53273 da MITSUBISHI, TD3473 da TOSHIBA e HD2515 da HITACHI.



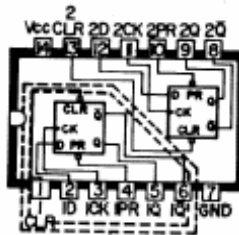
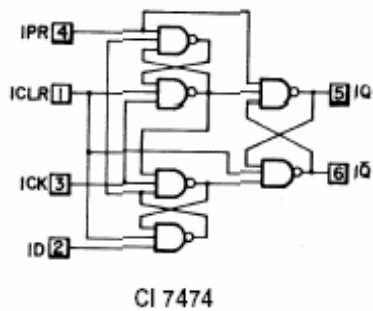
Existe também o 7476 que possui 2 FLIP-FLOP JK com PRESET e CLEAR.



CI 7474 - (Dual D - Tipe Positive Edge Trigger o Flip-Flops White Preset and Clear)

Na figura , apresentamos a distribuição dos pinos, o circuito equivalente e a tabela verdade do 7474. Este CI possui 2 FLIP-FLOPS tipo D com TRIGGER EDGE POSITIVO com entradas de PRESET e CLEAR. O seu funcionamento é o mesmo já explicado anteriormente.

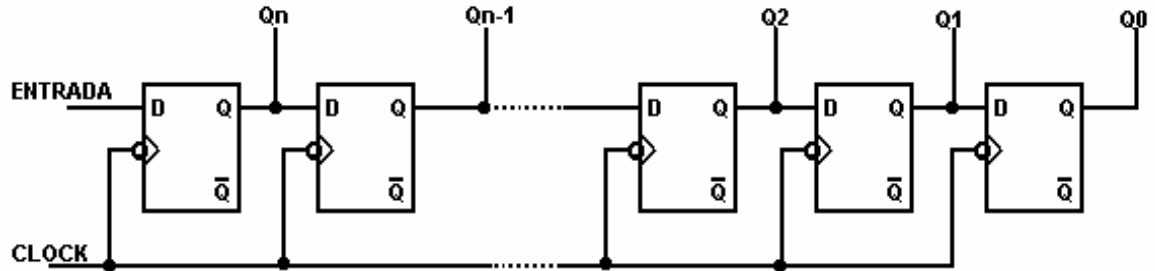
Dentre os CI's TTL que podem substituí-lo temos o M53274 da MITSUBISHI, o TD 3474 da TOSHIBA e o HD 2515 da HITASHI.



ENTRADA				SAÍDA	
PR	CLR	CK	D	Q	\bar{Q}
0	1	X	X	1	0
1	0	X	X	0	1
0	0	X	X	H	H
1	1	1	1	1	0
1	1	1	0	0	1
1	1	0	X	Qa	$\bar{Q}a$

Registadores de Deslocamento (Shift Register)

Como vimos os flip flops podem armazenar durante um período o bit de sua saída. Porém, se necessitarmos guardar uma informação maior que um bit, o flip flop irá se tornar-se insuficiente. Para isso usa-se os registradores de deslocamento, trata-se de um certo número de FF JK mestre escravo ligados de forma que a saída de cada bloco sejam aplicadas nas entradas do flip flop seguinte. Observe um registrador de deslocamento para N+1 bits



Os registradores de deslocamentos podem ser aplicados como:

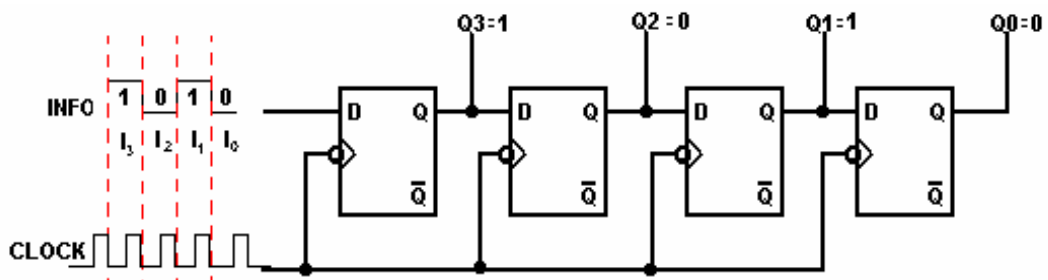
- 1- Conversor série paralelo
- 2- Conversor paralelo série
- 3- Registrador de entrada série e saída série
- 4- Registrador de entrada paralela e saída paralela.

Informação paralela _ é a informação onde todos os bits se apresentam simultaneamente, assim uma informação paralela necessita de tantos fios quanto forem os bits contidos nela.

Informação série_ é a informação onde todos os bits utiliza apenas um fio, sendo que os bits de dados vêm seqüencialmente, um após os outros.

Conversor Série -Paralelo

O registrador de deslocamento pode ser usado para converter uma informação série em paralela. Se aplicarmos uma informação série $I = 1010$ (I_3, I_2, I_1, I_0) à entrada série do registrador as saídas Q_3, Q_2, Q_1 e Q_0 conterá a informação em paralelo.



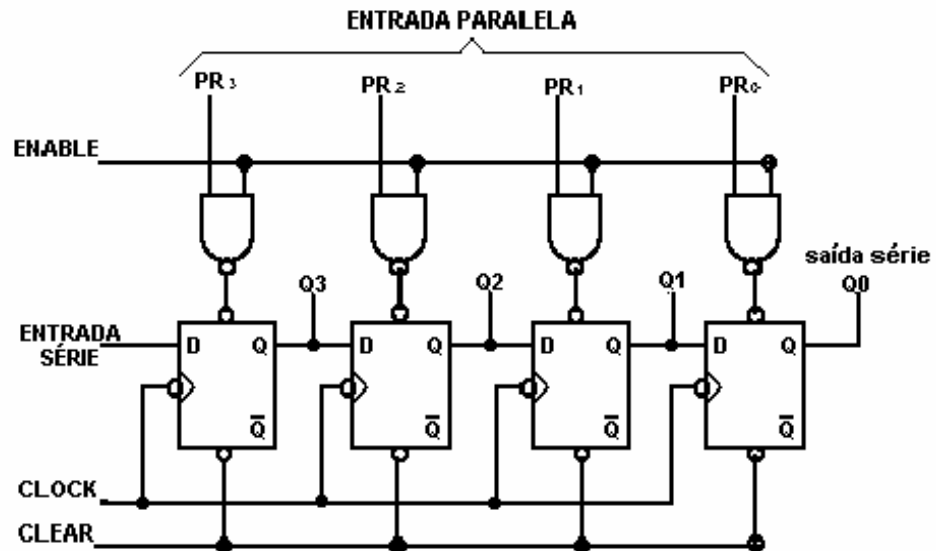
Deve-se ressaltar que os flip flop são mestre escravos e têm comutação na descida do clock, após o quarto pulso de clock, teremos a seguinte situação: $Q_0 = 0$, $Q_1 = 1$, $Q_2 = 0$ e $Q_3 = 1$

Observe a seqüência das transferências na tabela abaixo:

Informação	descida de clock	Q ₃	Q ₂	Q ₁	Q ₀
I ₀ = 0	1ª	0	0	0	0
I ₁ = 1	2ª	1	0	0	0
I ₂ = 0	3ª	0	1	0	0
I ₃ = 1	4ª	1	0	1	0

Conversor Paralelo – Série

Para entrarmos com uma informação paralela , necessitamos de um registrador com entradas de preset e clear , pois é através destas que fazemos que o registrador armazene uma informação paralela .



Quando a entrada de enable estiver em 0 ,as entradas preset assumirão níveis 1, fazendo com que o registrador atue normalmente . Quando a entrada enable for 1,as entradas de preset assumirão valores complementares das entradas PR₀ , PR₁ , PR₂ ,PR₃ . Logo as saídas dos flip flops assumirão os valores presentes nas entradas de PR₀ , PR₁ , PR₂ ,PR₃ . A informação de modo série aparece na saída Q₀ , ou seja, a cada descida de pulso de clock , irá assumir seqüencialmente os valores I₀ , I₁ , I₂ , I₃ .

Registrador Com Entrada Série e Saída Série

Nesta aplicação , após a entrada de informação , se inibirmos a entrada de clock a informação permanecerá no registrador até que haja nova entrada. Observa se que o registrador funciona como uma memória . A saída série é recolhida na saída Q_0 .

Registrador Com Entrada Paralela e Saída Paralela

Fazendo a entrada paralela pelas entradas de PR , se inibirmos a entrada de clock, a informação paralela pode se colhida nas saídas Q_0 , Q_1 , Q_2 , Q_3 .

Contadores

Introdução

Os contadores são importantes circuitos eletrônicos digitais, são circuitos lógicos seqüenciais e necessitam de uma característica de memória. Os contadores digitais tem as seguintes características importantes:

1. Número máximo de contagens (módulo do contador).
1. Contagem crescente e decrescente .
2. Operação assíncrona ou síncrona.

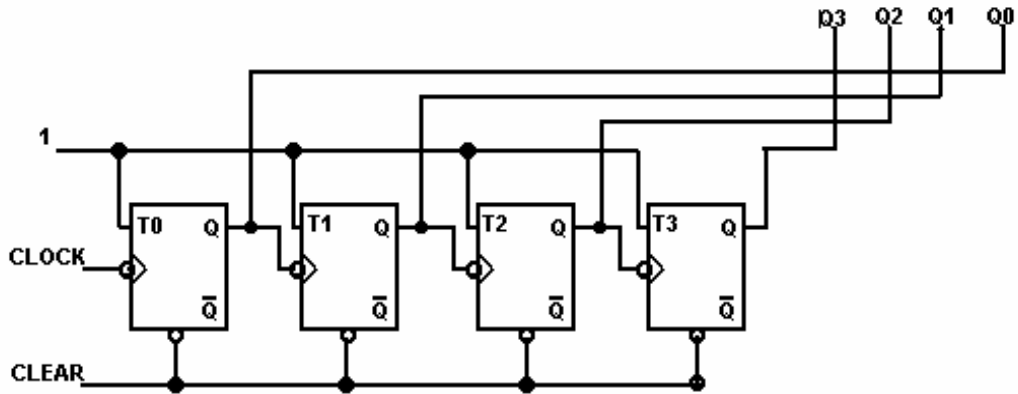
Contadores Assíncronos

Os contadores digitais contarão apenas em binário ou em códigos binários. A Fig. mostra a seqüência da contagem em binário desde 0000 até 1111 (0 a 15 em decimal). Um contador digital que contaria desde o binário 0000 até 1111 como indicado na tabela pode ser chamado de contador de módulo 16. O módulo de um contador é o número de contagens que ele completa. O termo "módulo" às vezes é abreviado para "mod". Este contador pode ser chamado de contador mod-16.

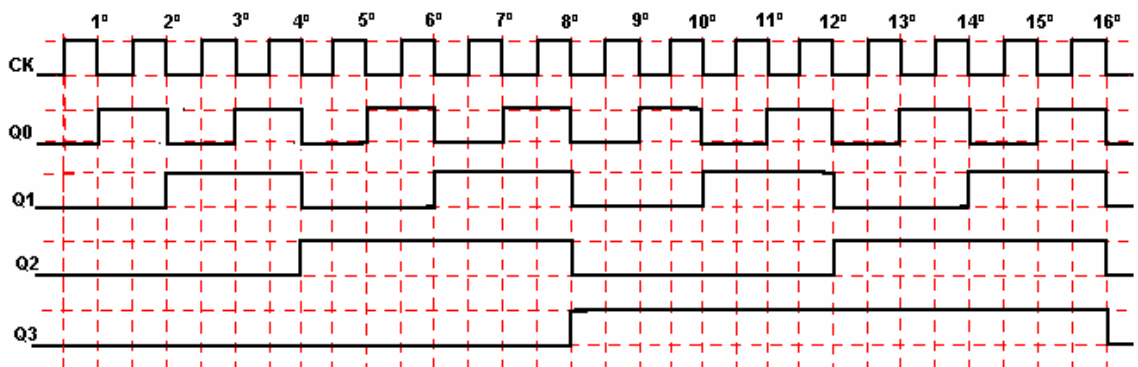
CONTAGEM DECIMAL	CONTAGEM BINÁRIA				CONTAGEM DECIMAL	CONTAGEM BINÁRIA			
	D	C	B	A		D	C	B	A
	0	0	0	0		0	8	1	0
1	0	0	0	1	9	1	0	0	1
2	0	0	1	0	10	1	0	1	0
3	0	0	1	1	11	1	0	1	1
4	0	1	0	0	12	1	1	0	0
5	0	1	0	1	13	1	1	0	1
6	0	1	1	0	14	1	1	1	0
7	0	1	1	1	15	1	1	1	1

Um diagrama lógico do contador mod-16 usando biestáveis JK é mostrado abaixo. Nota-se primeiro que as entradas de dados J e K dos biestáveis são unidas, formando um FF tipo T. Isso significa que cada biestável está em seu modo chave (toggle). Cada pulso de clock fará então com que o biestável implemente para seu estado oposto. Nota-se também que a saída Q do FF1, está conectada diretamente à entrada de clock do seguinte (FF2) e assim por diante.

Os indicadores de saída (lâmpadas ou LEDs) são mostrados à direita na parte superior para monitorar a saída binária do contador. O indicador A é o LSB (bit menos significativo), enquanto D é o MSB (bit mais significativo).



Costuma-se analisar uma operação do contador usando formas-de-onda (diagramas de temporização). A Figura abaixo é uma forma-de-onda do contador mod-16. A linha superior representa a entrada (CK) de clock no FF1. A linha inferior mostra a contagem binária. Note que o contador binário está limpo, (reset) para 0000, à esquerda. Cada pulso de clock aumentará a contagem binária de 1 quando nos deslocarmos para a direita sobre o diagrama.



O círculo na entrada de clock do biestável JK, significa que a unidade chaveará na transição 1-para-0 (transição de descida) do pulso de clock. Observemos o pulso 1 de clock. A transição 1-para-0 chaveia FF1. A saída Q de FF1 vai de BAIXA para ALTA. A contagem binária é agora 0001.

Observemos o pulso 2 de clock. O edge (borda) posterior do pulso dispara FF1. Este FF1 chaveia, e a saída Q vai de ALTA para BAIXA. Quando a saída Q de FF1 vai de ALTA para BAIXA, ela por sua vez chaveia FF2 (a saída Q de FF1 está ligada à entrada CK de FF2). FF2 chaveia de BAIXA para ALTA. Depois do pulso 2 de clock, a contagem binária aumentou para 0010.

Observemos o pulso 3 de clock. O edge posterior (transição de descida) dispara FF1, que chaveia. A saída Q de FF1 chaveia de BAIXA para ALTA. A contagem binária (ver linha inferior) aumentou para 0011.

Observemos o pulso 4 de clock. O edge posterior dispara FF1, que chaveia (muda de estado), com Q indo de ALTA para BAIXA. Esta transição 1-para-0 em Q de FF1 por sua vez chaveia FF2. A saída Q de FF2 chaveia de ALTA para BAIXA. Esta transição 1-para-0 em Q de FF2 por sua vez faz com que FF3

chaveie. A saída Q de FF3 chaveia de BAIXA para ALTA. A contagem binária agora permanece em 0100.

Observemos a linha pontilhada após o pulso 4 na forma-de-onda ALTA em Q de FF3. Note que um grande espaço de tempo passa antes que FF3 finalmente chaveie para seu estado ALTO. Isto é porque FF1 chaveia, que por sua vez chaveia FF2, que por sua vez chaveia FF3.

Tudo isto consome tempo. Este tipo de contador é chamado contador com ondulação. O disparo de biestável a biestável efetivamente ondula através do contador. Este contador é também denominado contador assíncrono, porque nem todos os biestáveis chaveiam exatamente ao mesmo tempo com o pulso de clock.

Observemos o restante da forma-de-onda, para nos certificarmos de que compreendemos sua operação. Note particularmente que no pulso 16 a transição 1-para-0 chaveia FF1. A saída de FF1 vai de ALTA para BAIXA. FF2 é chaveado por FF1. A saída de FF2 vai de ALTA para BAIXA. FF3 é chaveado (comutado) por FF2 e assim por diante.

Note que todos os biestáveis comutam (chaveiam ou mudam de estado) alternadamente e vão desde seu estado ALTO até seu estado BAIXO. A contagem binária é depois voltada para 0000. O contador não pára em sua contagem máxima, mas continua contando enquanto os pulsos de clock são introduzidos na entrada CK de FFI.

Ao contar cuidadosamente o número de pulsos ALTOS sob os 16 primeiros pulsos de clock (na linha de saída de FF1). Acharemos oito pulsos. Dezesesseis pulsos entram em FF1 e somente oito pulsos saem. Este biestável é portanto um divisor de frequências. 16 dividido por 8 é igual a 2. FF1 pode, portanto, ser também considerado um contador de dividir-por-2.

Contemos os pulsos de saída ALTA em FF2. Para 16 pulsos de clock, somente quatro pulsos aparecem na saída de FF2. Então, dividindo 16 por 4 é igual a 4. A saída Q de FF2 pode ser considerada um contador de dividir por-4. Acha-se que a saída de FF3 é um contador de dividir-por-8. A saída de FF4 é um contador de dividir-por-16. Em alguns dispositivos, tais como clocks digitais, a divisão de frequência é uma tarefa muito importante dos contadores.

A forma-de-onda confirma que um contador é um dispositivo lógico seqüencial. A característica de memória é também importante, pois o biestável deve "lembrar" quantos pulsos de clock chegaram à entrada CK. O contador assíncrono é o tipo mais simples de contador. Seu inconveniente é o retardo de tempo, pois um biestável dispara o seguinte e assim por diante.

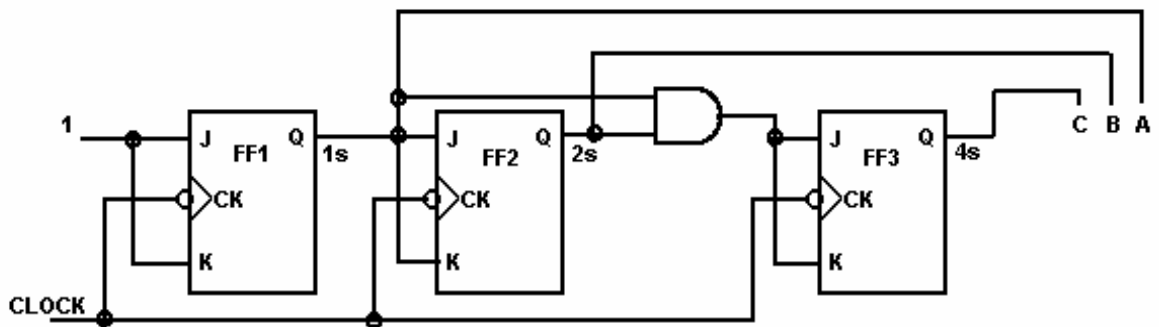
Com base nesta análise é fácil entender os contadores assíncronos decrescente e de contagem sequencial de 0 a n.

Contadores Síncronos (Paralelos)

O contador com ondulação assíncrono tem a limitação do atraso de tempo no disparo de todos, biestáveis. Para eliminar este problema, podem ser usados contadores paralelos. O diagrama de símbolos lógicos de um contador paralelo de 3 bits está ilustrado abaixo.

Note que todas as entradas CK estão ligadas diretamente ao pulso de clock. Elas estão montadas em paralelo. Note também que são usados biestáveis JK. O FF1 é o contador de casas 1s e está sempre no modo chave (toggle). FF2 tem suas entradas J e K ligadas à saída de FF1 e estará no modo de retenção (hold) ou no modo chave (toggle):

As saídas de FF1 e FF2 são introduzidas numa porta AND. A porta AND controla o modo de operação de FF3. Quando a porta AND é ativada pelos 1s em A e B, FF3 estará em seu modo chave (toggle). Com a porta AND desativada, FF3 estará em seu modo de retenção (hold). FF2 é o contador de casas 2s e FF3 é o contador de casas 4s.



A seqüência de contagem deste contador paralelo de 3 bits é mostrada abaixo. Note que este é um contador de módulo 8 (mod-8). O contador iniciará a contagem em binário 000 e contará até 111. Depois ele volta a 000 para iniciar a contagem novamente. Na A forma-de-onda (diagrama de temporização) do contador paralelo mod-8, a linha superior representa as entradas de clock para todos os três biestáveis.

As saídas (Q) de cada um dos biestáveis são mostradas nas três linhas do meio. A linha inferior dá a contagem binária indicada.

O pulso 1 chega em cada um dos três biestáveis. FF1 comuta de BAIXO para ALTO. FF2 e FF3 não chaveiam porque eles estão no modo de retenção (J e K = 0). A contagem binária é agora 001.

O pulso 2 chega em todos os biestáveis. FF1 e FF2 comutam porque eles estão no modo chave (J e K = 1). FF1 vai de ALTO para BAIXO, enquanto FF2 vai de BAIXO para ALTO. FF3 está ainda no modo de retenção, e portanto não comuta. A contagem agora é 010.

O pulso 3 chega em todos os biestáveis ao mesmo tempo. Somente FF1 comuta. FF2 e FF3 estão no modo de retenção devido a ter J e $K = 0$. A contagem binária é agora 011.

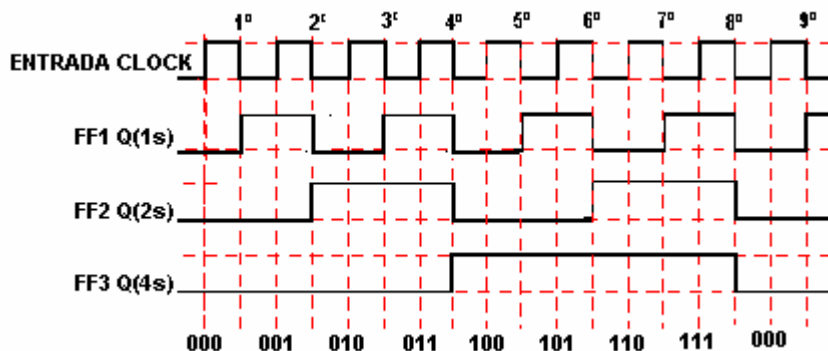
No pulso 4, a porta AND é ativada pouco antes do pulso de clock ir de ALTO para BAIXO. A porta AND colocará FF3 no modo chave (J e $K = 1$).

Na transição 1-para-0, do pulso 4 do clock, todos os biestáveis comutam. FF1 e FF2 vão de ALTO para BAIXO. FF3 comuta de BAIXO para ALTO. A contagem binária é agora 100.

Note a linha pontilhada abaixo da borda posterior do pulso 4 de clock. Dificilmente qualquer atraso de tempo é evidente de FF1 a FF3, porque todos os biestáveis são sincronizados (clocados) exatamente ao mesmo tempo. Esta é a vantagem do contador do tipo paralelo.

Os contadores paralelos são também chamados contadores síncronos, porque todos os biestáveis chaveiam exatamente em tempo com o clock. Os contadores paralelos são mais complicados, mas são usados onde o problema do atraso de tempo com um contador com ondulação (assíncrono) causaria inconvenientes.

Observe o restante da forma-de-onda, onde cada biestável é sincronizado em cada pulso de clock. FF1 sempre comuta. FF2 e FF3 podem estar no modo chave ou no modo de retenção.



Exercício

- Supomos que estamos precisando de um contador assíncrono de módulo-6. A primeira etapa na construção de um contador assíncrono mod-6 consiste em listar a seqüência da contagem. A seqüência da contagem do contador de mod-6 é de 000 a 101. Note que um contador de 3 bits é necessário três FF tipo T: Conforme mostrado, o contador de 3 bits normalmente conta de 000 a 111. As duas últimas contagens (110 e 111) devem ser omitidas. Desenvolva o circuito lógico para tal contagem e desenhe o diagrama de tempo
- Desenvolva o desenho lógico e o diagrama de tempo para um contador assíncrono, com contagem decrescente
- Projete um contador assíncrono com contagem crescente e decrescente. Faça seu diagrama de tempo.

Contador De Década

Um contador de décadas é provavelmente o contador mais amplamente usado. Um contador de décadas poderia também ser descrito como um contador de módulo 10. A Figura abaixo mostra em diagrama um contador assíncrono de mod-10. São usados quatro biestáveis JK mais uma porta NAND para montar o contador de décadas. A unidade conta exatamente como o contador de mod-16 até 1001. O binário 1001 é a contagem máxima desta unidade. Quando a contagem tenta avançar para 1010, as saídas altas (D = 1 e B = 1) são introduzidos na porta NAND. A porta NAND é ativada, reajustando o indicador visual em 0000.

Um símbolo lógico geral às vezes é usado num contador quando comprado em forma de CI. O símbolo lógico Fig.B , pode ser substituído no diagrama de contador de décadas na Fig. A.

Uma entrada limpar (clear) , foi acrescentada ao contador de décadas na Fig.B. Esta entrada de limpar não aparece no contador de décadas na Fig. A. Um 0 lógico ativa o clear e limpa a saída em 0000.

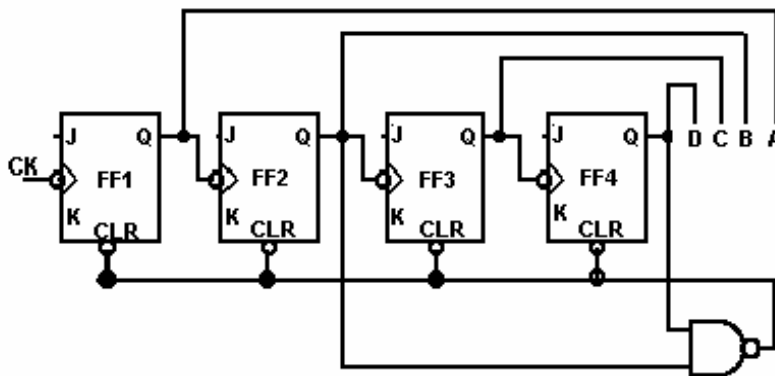


FIG. A

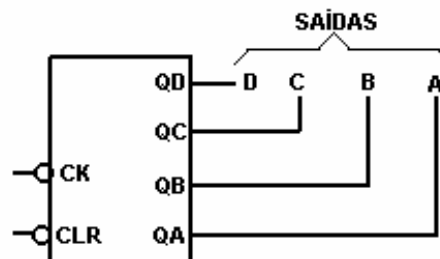
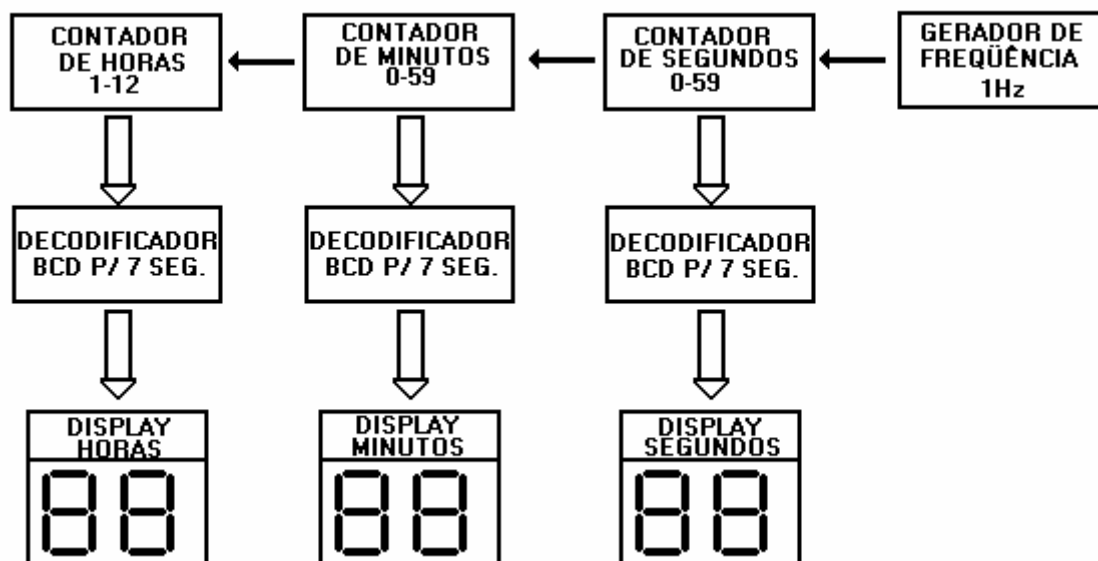


FIG. B

Trabalho Semestral

Com os elementos estudados até aqui , desenvolva um circuito lógico e monte-o , com base no circuito em blocos de um relógio digital básico, desenhado abaixo.



Observação : este trabalho deve ser realizado em grupo de no máximo quatro alunos , e com apresentação em laboratório.

Tipos de Memórias Semicondutoras

As memórias são locais onde armazenam-se dados e programas em um sistema de computação. As memórias são as partes mais ativas de um computador, armazenando programas e dados antes, durante e após a execução. Pode-se afirmar que a memória é equivalente a milhares de registradores; cada um armazenando uma palavra binária.

Há duas únicas ações que podem ser realizadas em uma memória. A primeira é a ação de guardar um elemento (ou um grupo de elementos) - em computação, esta ação genericamente denominada de armazenar e a operação em si, que é realizada para a consecução dessa ação de armazenamento, é chamada de escrita ou gravação ("write"). A segunda é a ação de recuperação do elemento guardado (ou grupo de elementos) para um uso qualquer - em computação esta ação se denomina recuperar ("retrieve") e a operação para realizá-la chama-se leitura ("read").

Até o final da década de 60, as memórias dos computadores eram magnéticas. As mais antigas eram de "tambor": um cilindro magnético, girando a alta velocidade, com cabeças de gravação e leitura escrevendo e lendo dados e instruções em sua superfície. Outras eram construídas com núcleo de ferrite: minúsculos toróides de ferrite, costurados por fios de acesso de dados e de endereçamento. Em 1969, a IBM introduziu em seu processador modelo 360/85, uma pequena (pelos padrões atuais) memória de 16 Kbytes, construída com transistores. Surgiam ali as memórias monolíticas, ou memórias a semicondutor. Desde então diferentes tipos de memórias tornaram-se disponíveis no mercado. Como consequência, o projetista tem muito onde escolher, mas a escolha é mais difícil, e deve se basear na adequação das características da memória às necessidade da aplicação.

Conceitos Preliminares

Quando se fala em Sistemas Digitais de um ponto de vista genérico, as memórias são vistas em dois grandes grupos: as de leitura e escrita e as de apenas leitura. As memórias do tipo de leitura e escrita são as tradicionais RAM's e como seu nome diz, tem seu conteúdo alterado durante o funcionamento do circuito. Em contrapartida, as memórias de apenas-leitura têm conteúdo fixo e os circuitos conseguem apenas ler o que ali está escrito. Como metáfora pode-se fazer a analogia de uma memória do tipo de leitura e escrita com um caderno e das memórias do tipo apenas-leitura com um livro. No caso de uma memória de computador, o elemento a ser manipulado fisicamente é o bit, o qual, em grupo de n bits (n pode variar consideravelmente, dependendo daquilo a que se está referindo), corresponde a unidade de informação a ser armazenada, transferida, recuperada, etc. Ou seja, a memória serve para guardar (armazenar) informações (na forma de bits) e recuperá-las quando desejado.

Para isso, realizam-se de escrita (transferência de bits de outro componente do sistema de computação; por exemplo: da UCP, de disco) e de leitura (transferência de bits da memória para a UCP, disco, etc.). Por informação, entendem-se as instruções e os dados de um programa.

Para que a informação possa ser armazenada em uma memória (operação de escrita) é necessário que seja definido um local disponível identificado de alguma forma precisa e única (um número, por exemplo). O número ou código que é associado ao local é o endereço ("address") que irá permitir que a informação possa ser localizada, assim como nossa residência é localizada pelo seu endereço.

Antes de analisar os aspectos práticos do uso de memórias como elementos de projeto, é preciso definir e tornar mais precisos alguns termos como endereço, posição, modo de acesso, entre outros, e apresentar algumas classes distintas de memória.

Posição e Endereço

Costuma-se chamar de memória os circuitos utilizados para armazenar uma alta quantidade de bits, organizados em palavras e posições de memória.

A palavra de memória é o conjunto de bits que pode ser, simultaneamente, lido ou escrito (gravado) na memória. Tipicamente, as memórias têm palavras de 8, 16 ou 32 bits (1, 2 ou 4 bytes); existem pastilhas de memórias de 1 bit e de 4 bits (meio byte ou um nibble), que podem ser associadas para construir memórias maiores.

A posição de memória é o local onde se armazena uma palavra. A cada posição está associado um endereço, que é um código binário utilizado para acessar a posição. A quantidade de endereços define, portanto, a quantidade de posições. A capacidade da memória é a medida do total de bits que ela armazena. Calcula-se a Capacidade da memória multiplicando-se a quantidade de posições pela largura da palavra. A capacidade da memória é normalmente expressa em tantas posições de tantos bits. Exemplo: Uma memória de 1024 posições, com palavras de 8 bits, tem uma capacidade de $8 \times 1024 = 8192$ bits (8Kb).

Ao expressar a capacidade de memória em bits mascara-se a sua organização interna (ou seja, o tamanho da palavra), e uma vez que essa organização é fundamental para a escolha e a associação das memórias num projeto, é comum referir-se à capacidade das memórias em termos do número de posições x tamanho da palavra.

Exemplo: Memória de 1024 x 8 bits, ou 1024 bytes, ou ainda 1 Kbyte (1K = 1024 posições)

Os endereços, codificados em binário, são apresentados ao módulo de memória através da via de endereço (address bus). Portanto, a cada posição de memória corresponde um endereço, em binário. Uma memória de 1024 posições terá 10 bits de endereço ($1024 = 2^{10}$); por outro lado, uma via de endereços de 16 bits poderá endereçar uma memória com $2^{16} = 65536$ posições (64Kb).

As Memórias do tipo Lê-Escreve, inúmeras são as características de acordo com as quais pode-se classificar as memórias em categorias, tais como:

- ◆ Modos de Acesso;
- ◆ Volatilidade;
- ◆ Tipo de Armazenamento.

Obs. Ressalte-se, entretanto, que o “universo” das memórias é vasto e muito rico em técnicas e idéias.

Modos de Acesso

Existem memórias de acesso seqüencial. Nelas, as posições de memória estão fisicamente organizadas em lugares sucessivos, e a leitura ou escrita de uma certa posição requer que se “caminhe” sobre todas as posições anteriores. Como ocorre com as fitas magnéticas: só se consegue atingir trechos no final da memória passando sobre os trechos iniciais. Dessa forma, o tempo necessário para se ter acesso a uma dada posição de memória (chamado de tempo de acesso), será tanto maior quanto mais distante a posição desejada estiver da posição inicial. Os discos magnéticos e as fitas magnéticas são dois típicos exemplos de memórias com acesso seqüencial.

Outro tipo de memória é o de acesso direto ou aleatório (RAM - Random Access Memory). Nessas é possível realizar-se a seleção direta de qualquer posição sem ter que passar sobre posições anteriores. O que caracteriza a memória de acesso direto é que o tempo de acesso é o mesmo para todas as posições, independentemente da posição inicial.

O termo Random Access Memory - RAM que deveria ser aplicado para diferenciar memórias de acesso direto ou “aleatório”, das seqüenciais, tem sido usado inadequadamente para descrever as memórias do tipo lê-escreve. As memórias apenas de leitura, ROM (Read-Only Memory), também são de acesso direto, e não são comercialmente referidas por RAM.

Volatilidade

Com relação à capacidade de reter os dados armazenados, os dispositivos de memórias podem ser divididos em duas categorias: voláteis e não-voláteis. As memórias voláteis mantêm o seu conteúdo armazenado apenas enquanto estiverem “alimentadas” com energia elétrica. As memórias não-voláteis mantêm seu conteúdo mesmo que falte energia.

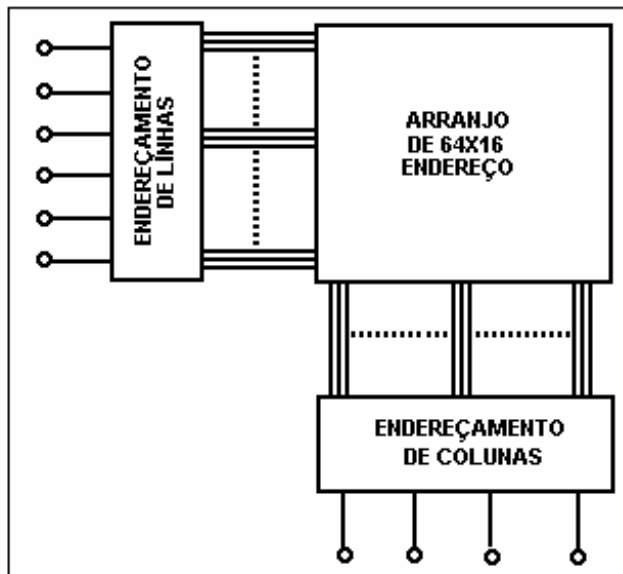
Tipicamente as memórias magnéticas são não-voláteis. As memórias do tipo lê-escreve de acesso direto são em geral voláteis, enquanto que as memórias de apenas-leitura, as ROM (Read-Only Memory) e seus derivados PROM, EPROM, EEPROM e EAROM são memórias não-voláteis.

Tipo de Armazenamento

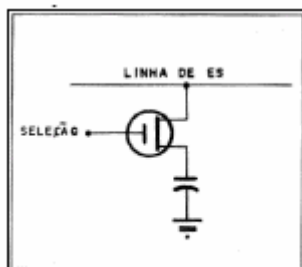
De acordo com a forma como a informação é armazenada, existem dois tipos de memórias ram: as estáticas e as dinâmicas.

As memórias estáticas são velozes e simples de serem utilizadas: a célula básica (a unidade que armazena um bit de informação) é constituída por um flip-flop tradicional que armazena "0" ou "1" (conteúdo daquela posição). Seu inconveniente é que a célula tem dimensões grandes, o que limita a quantidade de posições que se consegue integrar em uma pastilha.

Na figura abaixo mostra um sistema de memória RAM estática de 1024 por 4. Isto significa que ela está formada por 64 linhas e 16 colunas ($64 \times 16 = 1024$) e o tamanho da palavra (byte) é de 4 bits. Como esta memória é formada por uma palavra de 4 bits, teremos 4 planos com arranjo de 1024 endereços para cada um, formando a memória 1024 x 4.



As memórias dinâmicas apresentam alta capacidade, velocidade moderada e baixo consumo. Sua célula é um capacitor para o armazenamento de carga, associado a um transistor. A presença de carga no capacitor é interpretada pelos circuito sensores da memória como nível H ; a ausência de carga, como nível lógico L. Exemplo dessa célula.



A RAM dinâmica utilizada como célula o elemento MOS. O armazenamento da informação é feito nas capacitâncias parasitas que se formam entre a porta e a fonte do transistor FET . Uma célula de memória dinâmica consiste em um

transistor MOSFET e um único capacitor MOS de alguns picofarads. O transistor atua como chave e o capacitor armazena a tensão correspondente ao nível zero ou um. A operação de leitura é feita ativando-se o gatilho ou porta do transistor.

Como a tendência da carga é de se dissipar, as memórias dinâmicas necessitam de um reforço periódico na carga, ou refrescamento. Esse requisito implica na necessidade de circuitos adicionais para realizar tal reforço de cargas periodicamente, o que torna mais complexa sua utilização. Além disso, na ocasião do refrescamento a memória não permite a realização de escritas ou leituras, reduzindo sua disponibilidade. Existem circuitos especialmente fabricados para controlar as memórias dinâmicas, os DRAM Controllers (Dynamic RAM Controllers), integrados em pastilhas LSI.

Devido a sua simples estrutura, a RAM dinâmica possibilita concentrar em sua área 4 vezes mais informações do que a RAM estática. Para diminuir o número de pinos de endereçamento, os fabricantes de chips de memória utilizam o sistema de multiplexação de endereços. Nesse caso, o endereço da fileira é aplicado em primeiro lugar, para depois aparecer o endereço da coluna. Uma alternativa intermediária, que pode ser utilizada em pequenos sistemas são as memórias pseudo-estáticas, as IRAM (Integrated RAM). Elas são dispositivos que contêm, num só encapsulamento, a unidade de memória dinâmica e a unidade de controle.

Do ponto de vista do usuário, a memória iRAM se comporta como se fosse estática. Suas características de consumo, capacidade e velocidade, todavia, são as de uma memória dinâmica.

As Memórias de Apenas-Leitura (ROM)

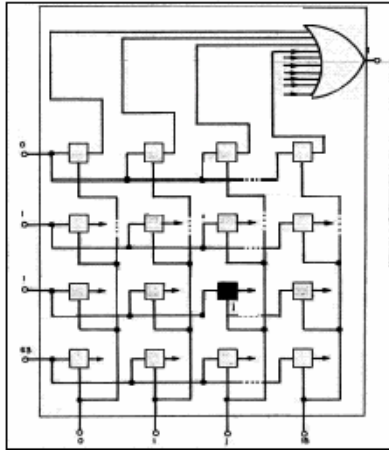
As memórias de apenas leitura (ROM - Read-Only Memory) tratadas aqui, são do tipo RAM (Random-Access Memory): acessam diretamente cada posição de armazenamento. Como o nome diz, as memórias de apenas-leitura não permitem a gravação dos dados. Em uso normal, pode-se apenas ler seu conteúdo. O conteúdo de uma ROM é gravado no momento da fabricação da memória, ou mais tarde, em aparelho especial. Dos vários tipos de ROM, a maioria deles tem a gravação de seu conteúdo feita fora do circuito.

Por isso, as ROM são usadas basicamente para o armazenamento de informações que não estão sujeitas a mudanças ao longo do processamento. Memórias do tipo ROM surgiram a partir da necessidade de armazenar informação (programação, tabelas, constantes, etc.) em equipamentos microprogramados.

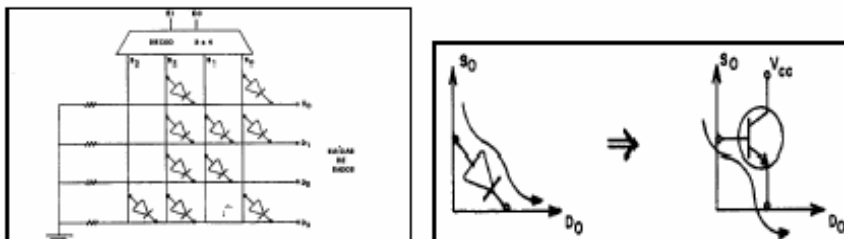
Nos microprocessadores, o programa básico (BIOS - Basic Input/Output System) é fixo e armazenado numa ROM, enquanto que os programas aplicativos e os dados manipulados são armazenados em memórias RAM do tipo lê-escreve.

Os aspectos tecnológicos ligados à construção das memórias de apenas-leitura são diversos, e dependem do particular tipo de ROM. Esses detalhes estão apresentados a seguir.

As Mask-ROM são o tipo mais antigo de memória ROM. Ela é composta de uma matriz de células de diodos ou transistores. A programação da memória com 1 ou 0 é feita pela conexão ou não de dispositivos à grade da memória, o que é feito durante a etapa de fabricação da memória, na fase de metalização das conexões. Veja a figura . Ela mostra uma matriz de uma memória com 4 posições de 4 bits. Os endereços E1 E0 selecionam uma das quatro posições, ativando uma das linhas S0 a S3 com um nível 1. A linha ativada, por exemplo S0 provoca a condução dos diodos que a une às saídas.



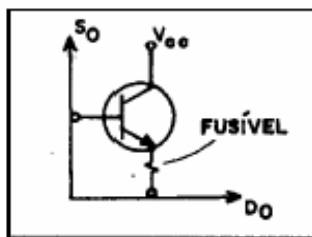
O conteúdo armazenado na posição selecionada, ou seja 1101, surge então na saída. Para evitar problemas de fan-out nas linhas de seleção, e diminuir as correntes envolvidas, as ROM utilizam transistores ao invés de diodos em cada um dos nós. Dessa forma, quem deve fornecer as maiores correntes, não é mais o decodificador, mas a própria fonte de alimentação, cabendo às linhas de seleção o fornecimento da reduzida corrente de base dos transistores que estiverem a ela ligados . No caso de dispositivos MOS essas correntes são ainda menores.



Um detalhe importante aqui, é que o conteúdo da memória deve ser fornecido pelo consumidor ao fabricante de memórias para que, na fabricação de memória, a máscara de metalização seja tal que faça as conexões corretas. É claro que essa programação (metalização) uma vez feita, não pode ser desfeita. Por isso, as ROM só devem ser encomendadas ao fabricante após ter seu conteúdo muito testado, para ser pequena a probabilidade de conter erros, e apenas em casos de grande volume, porque o custo dessa produção sob medida só se justifica quando se prevê a utilização em altas quantidades.

Memória Prom (Programável Pelo Usuário)

Para superar o problema de custo de programação, foi desenvolvida a PROM - Programmable ROM, que é semelhante à ROM programável por máscara, exceto pelo fato de ser o usuário quem faz a programação do seu conteúdo, utilizando programadores de PROM. Internamente as PROM são também constituídas por uma matriz de diodos ou transistores como as ROM; com a diferença que esses dispositivos encontram-se previamente ligados à grade da memória, o que significa ter valor lógico 1 armazenado em todos os bits. O procedimento de programação consiste em gravar 0 nas posições desejadas o que é conseguido "queimando-se" minúsculos fusíveis existentes na grade, para cada bit, de forma a separar os transistores da grade.

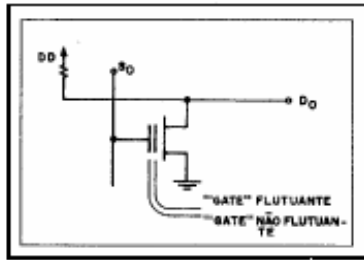


O fusível é queimado selecionando-se na PROM o endereço e a linha de dados desejados, e aplicando-se um pulso de alta tensão, tipicamente de 10 Volts a 30 Volts, através de um pino especial da pastilha. Esse procedimento também é feito uma só vez, o que significa que se a programação foi errada, a pastilha estará perdida. Entretanto o custo da pastilha é bem menor que o de uma ROM já que não é fabricada sob medida.

Memória Eprom – (ROM Programável e Apagável)

O passo seguinte da indústria foi o das EPROM - Erasable Programmable ROM, que são programáveis pelo usuário, mas que ainda podem ter seu conteúdo apagado. O processo de apagamento as faz retornar ao estado inicial, com conteúdo 1 em todos os bits.

Ao invés da tecnologia de "fusíveis" utilizados nas PROM, as EPROM baseiam-se em programação por armazenamento de carga. Cada bit da memória possui um transistor MOS com dois gates, um deles flutuante, não conectado à grade da memória, e isolado por material de altíssima impedância. Em estado "apagado", como ao sair da fábrica, esses transistores não conduzem quando selecionados e o conteúdo das posições de memória é levado a 1 por resistores chamados de pull-up. Para gravar um valor 0 numa determinada posição, aplica-se uma alta tensão no gate flutuante, o que causa uma ruptura (break-down) no material isolante e permite o acúmulo de cargas no gate flutuante, as quais ali permanecem mesmo após o término do pulso de tensão, devido à alta impedância do material isolante. A presença dessas cargas no gate do transistor provoca a condução quando a posição daquele bit for selecionada, Com isso aquela linha de bit é levada para 0 .



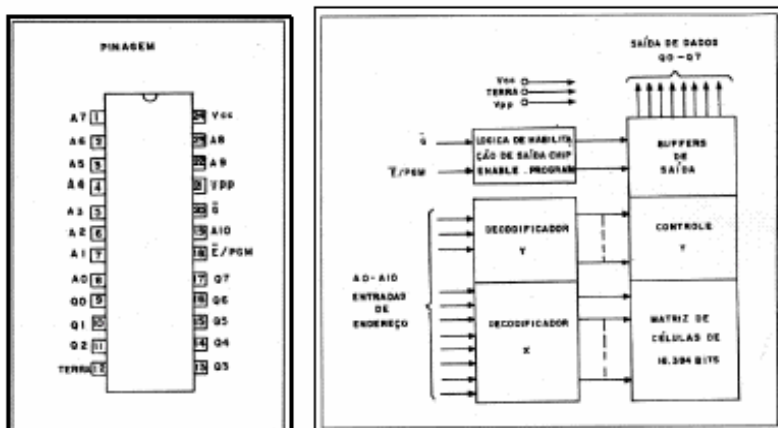
A alta impedância do material isolante, permite que uma EPROM mantenha sua programação por no mínimo 10 anos, se bem programada pelo aparelho programador de EPROM.

Para reprogramá-la, é necessário, antes, apagar a programação anterior, o que é conseguido expondo-a à luz ultravioleta (dai elas serem chamadas de UVEPROM). Os fótons de alta energia da luz UV colidem com os elétrons armazenados no gate flutuante e provocam o seu espalhamento, desfazendo o acúmulo de cargas e, portanto, a programação. Existe uma quantidade limite de vezes que a EPROM pode ser apagada e regravada.

Dados dos fabricantes mostram que a exposição constante à luz fluorescente de ambientes de trabalho pode apagar uma EPROM em cerca de 3 anos, enquanto que a exposição ao sol desfaria o conteúdo de uma pastilha EPROM em cerca de uma semana. Para evitar isso, costuma-se cobrir a janela de quartzo da pastilha com uma etiqueta opaca.

Exemplo de uma Memória EPROM (2716)

Para ilustrar na prática o estudo das memórias, escolhemos o CI 2716. É uma EPROM de 16Kb organizada em 2048 palavras de 8 bits cada. A EPROM 2716 é alimentada com 5 volts e suas entradas operam com os níveis TTL, exceto a entrada de programação que opera com tensão de 25 volts. Este CI não necessita de clock e nem de refresh. Na figura abaixo vemos a pinagem e o diagrama em blocos interno.



A operação de apagamento começa a ocorrer quando o dispositivo é exposto à luz cujo comprimento de onda é de 4000 Angstroms. Deve-se observar que a luz do sol bem como de certas lâmpadas fluorescentes tem esta faixa de comprimento de onda. Segundo estimativas, a exposição contínua do dispositivo em uma sala com iluminação fluorescentes pode apagá-lo em aproximadamente 3 anos, ao passo que a luz do sol poderá efetuar o apagamento em uma semana.

De qualquer forma, é bom cobrir a janela do dispositivo com um selo escuro a fim de evitar o apagamento indesejável.

Para o apagamento do CI 2716 recomenda-se sua exposição à luz ultravioleta com comprimento de onda de 2537 Angstroms, devendo se distanciar uns 3 cm do tubo de luz e permanecer ali por cerca de 20 minutos.

EEPROM ou E2PROM (PROM Eletricamente Apagável)

As E2PROM são apagáveis e reprogramáveis como as EPROM, com a diferença de que isso pode ser feito no próprio circuito, eletricamente, sem necessidade de uso de equipamentos adicionais (apagadores e programadores). Além disso pode-se programar apenas um bit, ou um byte, sem ter que apagar a memória inteira.

A E2PROM também tem sua célula constituída de transistores com gate flutuante, embora o material isolante seja de uma espessura muito menor. Desta forma cada bit pode ser apagado pela aplicação de uma tensão no gate não flutuante, oposta à que gerou as cargas na gravação.

A E2PROM, apesar do nome ROM, permite leitura e escrita; todavia, ela não substitui uma autêntica" memória do tipo RAM (lê-escreve) pois ela tem tempos de escrita muito superiores, tem custo muito maior e aceita um número limitado (10 mil) de ciclos de apagamento/gravação.

Com essas características, e lembrando que a E2PROM não é volátil, ela é muito útil para o armazenamento de dados que devem ser preservados quando o equipamento for desligado, ou se ficar sem alimentação. Tipicamente esses dados podem ser alterados, desde que não freqüentemente, ao longo da operação. Esses seriam os casos dos dados de configuração de um equipamento, dos dados de tabelas, etc.

Memória Flash (Flash Memory)

As memórias flash são uma outra alternativa para aplicações de memórias não-voláteis que requerem reprogramação no circuito e maiores capacidades/densidades. Tais memórias são um misto de EPROM e E2PROM: são eletricamente apagáveis como a última, mas o apagamento necessariamente é feito em toda a memória, como na primeira. Não é possível apagar byte a byte. O tempo de apagamento e regravação é bastante pequeno, cerca de 5 segundos

para 1 Megabit, mas pode exigir uma tensão adicional de 12 ou 13 Volts, conforme o fabricante.
A vantagem das flash memories é o seu custo que tende a ser menor que o da E2PROM em função de sua maior densidade; entretanto como as tecnologias estão por se consolidar no mercado, é conveniente conferir as várias opções oferecidas pelos fabricantes.

Referências Bibliográficas

IDOETA ,Ivan – CAPUANO , Francisco ; Elementos de Eletrônica Digital.
São Paulo : Érica, 1998

SZAJNBERG, Mordka ; Eletrônica Digital; LTC- 1988

AZEVEDO, João B; TTL-CMOS. São Paulo ; Érica - 1984