

OUTROS LIVROS DA PETIT

- MANUTENÇÃO DE MICROCOMPUTADORES
- ELETRÔNICA DE VIDEOGAMES
Teoria e Manutenção
- CONSTRUA SEU COMPUTADOR POR MEIO SALÁRIO-MÍNIMO
- ELETRÔNICA BÁSICA
Teoria e Prática
- RÁDIO
Teoria e Técnicas de Consertos
- SILK SCREEN
Para Eletrônica
- TELEVISÃO
TEORIA E CONSERTOS
- AUTOMÓVEIS
Guia de Pequenos Consertos
- FOTOGRAFIA

PETIT

EDITORA E MARKETING DIRETO LTDA.

Av. Brigadeiro Luiz Antonio, 383 - SP CEP 01317

Petit

ELETRÔNICA DIGITAL

Maurício Caruzo Reis

MAURÍCIO CARUZO
REIS

ELETRÔNICA DIGITAL

Petit
editora

ELETRÔNICA DIGITAL

TEORIA E APLICAÇÕES

MAURICIO CARUZO REIS

**Dados de Catalogação na Publicação (CIP) Internacional
(Câmara Brasileira do Livro, SP, Brasil)**

R311e Reis, Maurício Caruzo, 1950-
Eletrônica digital: teoria e aplicações / Maurício Caruzo Reis. —
São Paulo: Petit, 1986.

1. Eletrônica digital I. Título.

86-1348

CDD-621.3815

Índices para catálogo sistemático:

1. Eletrônica digital: Engenharia 621.3815

PETIT

Editora e Marketing Direto Ltda.
Av. Brig. Luiz Antonio, 383 – 2º
São Paulo – Capital
Cep 01317

TODOS OS DIREITOS RESERVADOS

É expressamente proibida a reprodução parcial ou integral sem a autorização
do editor

1985

	<u>Página</u>
INDICE	
Introdução_____	9
Capítulo 1 - <u>LÓGICA DIGITAL</u> _____	12
Sinal Digital	
Funções Lógicas	
Combinações Lógicas	
Capítulo 2 - <u>ARITMÉTICA DIGITAL</u> _____	37
Números Binários	
Conversão de decimal em binário	
Adição de binários	
Subtração de binários	
Scaling	
Códigos binários	
7-LED display	
Capítulo 3 - <u>DISPOSITIVOS ELETRÔNICOS</u> _____	62
Diodos semicondutores	
Transistor bipolar	
JFET - transistor FET de junção	
MOSFET	
CMOS	
Amplificadores operacionais	
Interfaces (buffer e latch)	
Capítulo 4 - <u>FAMÍLIAS LÓGICAS</u> _____	87
RTL	
IIL (ou I ² L)	
DTL	
ECL	
TTL	
Interface TTL-CMOS	

	<u>Página</u>
Capítulo 5 - <u>FLIP-FLOP E MULTIVIBRADORES</u>	100
Configuração básica e estado do FF	
Clock	
FF-SR sincronizado	
Controle direto	
FF-Master - Slave	
FF-ac acoplado	
FF-armazenamento capacitivo	
FF-atraso de propagação	
FF-JK	
FF-tipo D (delay-atraso)	
Flip-flop na prática	
Multivibrador monoestável	
Multivibrador a estável	
Capítulo 6 - <u>REGISTROS E CONTADORES</u>	123
Registro de deslocamento	
Contador de ripple	
Scaler	
Contador circular	
Gerador de sequência	
Capítulo 7 - <u>CONVERSÃO ANALÓGICO-DÍGITAL</u>	140
Chaveamentos digitais	
Conversão digital para analógico - D/A	
Conversão analógica para digital - A/D	
Capítulo 8 - <u>MEMÓRIAS SEMICONDUTORAS</u>	156
Memória de acesso sequencial	
RAM	
ROM	

	<u>Página</u>
Capítulo 9 - <u>MICROPROCESSADORES</u>	167
Evolução dos microprocessadores	
Arquitetura do microprocessador	
Temporização do microprocessador	
Barras de endereços, dados e controles	
Microcontroladores	
Microprocessador Z-80	
Capítulo 10 - <u>INSTRUMENTOS PARA TESTE E MANUTENÇÃO DE APARELHOS DIGITAIS</u>	183
Ponta de prova lógica	
Tester estático (COMPUTESTER)	
LSA - analisador de níveis lógicos	
Analisador de assinaturas	
Frequencímetro	
Proto-board	
Wire-Wrap	
Capítulo 11 - <u>CIRCUITOS ILUSTRATIVOS</u>	192
Joystick do ODISSEY	
Endereçamento de ROM do TK	
Detector - Contador	
Seletor de Canais de TV	
Capítulo 12 - <u>PROJETOS PARA VOCÊ MONTAR</u>	203
Ponta de prova lógica	
Cronômetro digital	
Gerador de Tons	
Apêndice - Código ASCII	

INTRODUÇÃO

A Eletrônica Digital se situa entre um dos mais antigos ramos do pensamento humano - a Lógica - e outro de vanguarda - a Microeletrônica. Juntar esses dois campos foi obra conjunta de várias disciplinas, principalmente da Física (Microeletrônica), Matemática (Sistemas) e Engenharia (Aplicação e Produtos). O resultado é uma avassaladora revolução que muda hábitos (automatização), cria novos parâmetros culturais (fácil acesso individual a grandes massas de conhecimentos - microcomputadores) e chega mesmo a questionar os mais sacrossantos dogmas políticos (robôs, capital ou trabalho?!)

Da Lógica, já trabalhada e amada pelos gregos 500 anos antes de Cristo, retirou-se o pensamento isento de tendências individuais, com processos reconhecidos universalmente como corretos. Estipuladas as condições das premissas - se verdadeiras ou falsas (o que na Eletrônica Digital será implementado por dois níveis de voltagem) - o argumento terá uma resposta de dutível e demonstrável por regras universais.

Entretanto, é necessário observar que existem várias "Lógicas". Aquela empregada na Eletrônica Digital é a lógica Formal, ou Bivalente (porque reconhece só dois estados num elemento - verdadeiro ou falso). A ela se agrega a Álgebra de Boole, ramo da Matemática onde aos dados também só se reconhecem dois valores (1 - verdadeiro, 0 - falso).

A Microeletrônica tem suas raízes mais pro

fundas na Física Quântica, área da Ciência com cerca de 50 anos, com princípios e postulados altamente discutíveis e até hoje com resistências à sua aceitação. Desenvolve uma teoria de partículas elementares, entre elas o Elétron, que levada à prática resultou nos materiais semicondutores. Estes num primeiro passo foram empregados para a fabricação de componentes discretos (cada um isolado, podendo ser manipulado sozinho). Numa segunda etapa inventou-se o "circuito integrado", produzindo-se numa pastilha semicondutora de alguns centímetros até milhares de componentes microscópicos, mas com as mesmas características dos componentes discretos (transistores, diodos).

A tecnologia microeletrônica para produção de circuitos integrados é extremamente complexa e geralmente as inovações ficam por conta de americanos e japoneses, que gastam quantias enormes em pesquisas e desenvolvimento na área. No Brasil vem sendo desenvolvido um esforço de atualização, principalmente na Universidade de São Paulo e mais ainda na UNICAMP. A SEI - Secretaria Especial de Informática, órgão do governo federal, criou em Campinas o CTI que está sendo equipado para ser o mais importante empreendimento nacional no setor.

Como o leitor pode perceber pelas siglas Microeletrônica, Eletrônica Digital e Informática são partes indissociáveis, ocorrendo sempre juntas num processo. A Eletrônica Digital associa-se à parte de "hardware" do sistema, executando eletronicamente a programação - "software".

Devido a suas características de alta precisão, baixo custo e versatilidade a Eletrônica Digital vem conquistando cada vez mais espaço. Aplicada inicialmente somente em computadores e instrumentação, está agora dominando o setor de jogos eletrônicos (vídeo-games), áudio (sistema de áudio digital a laser, rádio sistema PCM) e televisão (sintonizadores e outros estágios). Nesta última, aliás, deve se operar u

ma grande revolução muito em breve, já que está sendo lançada com grande sucesso a televisão microprocessada - totalmente digital (baseada no Kit DIGIT 20000).

CAPÍTULO 1 – LÓGICA DIGITAL

A Eletrônica Digital tem como característica a extensa utilização da Lógica Bivalente, sistema em que se pode atribuir apenas dois estados a um elemento, mutuamente exclusivos (alto x baixo, aceso x apagado, verdadeiro x falso).

(1) SINAL DIGITAL

Dizemos "dispositivo analógico" quando todos os níveis elétricos (voltagem ou corrente) são processados proporcionalmente. Assim, num alto-falante poderíamos ter os seguintes valores de entradas (inputs) e saídas (outputs):

<u>INPUT (volts)</u>	<u>OUTPUT (volts)</u>
10	50
11	55
18	90
20	100
22	110
22,5	111
25	125

Observe que entre 10 V e 25 V na entrada pode haver qualquer outro valor (por exemplo, 22V) e todos eles serão "multiplicados" (amplificados) por 5,

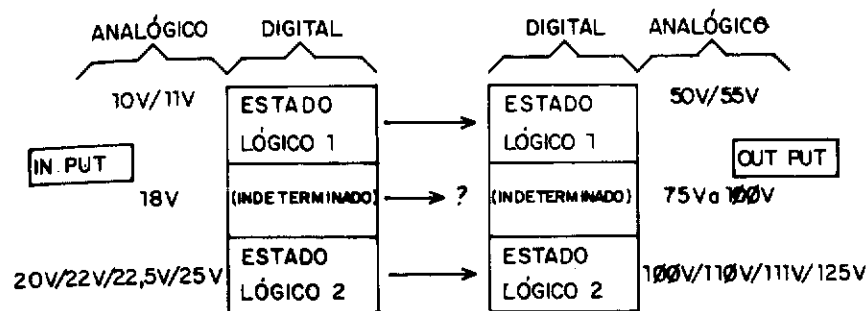


figura 1-1 - Sinais analógicos e digitais

no exemplo. É claro que cada dispositivo tem seu próprio fator de amplificação.

"Dispositivos digitais", por sua vez, só reconhecem dois níveis elétricos, tanto na entrada como na saída. Até um certo nível elétrico caracteriza-se um único estado lógico, excedendo-o atingimos outro estado; entre os dois interpõe-se uma faixa de segurança onde não se deve operar, pois não há confiança de qual estado se trata.

A figura 1-1 apresenta uma visualização, tomando a tabela anterior e estipulando que de 10 a 15 V na entrada é ESTADO LÓGICO (INPUT) 1, de 20 a 25 V ESTADO LÓGICO (INPUT) 2; na saída teremos de 50 V a 75 V ESTADO LÓGICO (OUTPUT) 1 de de 100 a 125 V ESTADO LÓGICO (OUTPUT) 2.

As observações abaixo, referidas à figu

ra 1-1 são válidas para qualquer circuito digital:

- (1) dentro de um mesmo estado lógico não importa o valor do sinal elétrico, contanto que ele esteja dentro de sua faixa (20V, 22V, 22,5V, 25V são todos ESTADO LÓGICO (INPUT) 2 e não se faz qualquer diferença entre eles).
- (2) se o dispositivo receber na entrada um sinal de estado indeterminado (de 15V a 20V) é desconhecido qual será a saída, que poderá ser ESTADO LÓGICO (OUTPUT) 1, ESTADO LÓGICO (OUTPUT) 2 ou INDETERMINADO.
- (3) o valor do sinal elétrico é especificado não só pela voltagem como também pela corrente. Se tivermos a voltagem correta mas a amperagem não for suficiente haverá uma indeterminação de qual estado lógico se trata. Para técnicos acostumados com Eletrônica Analógica, onde raramente se considera (e menos ainda se mede) a corrente elétrica, este é um ponto problemático. Testes que indicam somente voltagem não são confiáveis, sendo necessários instrumentos sensíveis simultaneamente a voltagem /corrente.

Dos dois estados lógicos o que tiver maior faixa de voltagem é indicado por 'H' (de "high" - alto em inglês) ou por '1'. O outro, de menor faixa de voltagem, é representado por 'L' (de "low" - baixo, em inglês) ou por '0' (zero). Podemos refazer o exemplo anterior como na figura 1-2. (Note que '0' não indica zero Volt mas sim a menor faixa de voltagem).

Na verdade os valores do nosso exemplo são incomuns. A figura 1-3 apresenta números da "lógica TTL", o padrão mais usado em Eletrônica Digital.

Denomina-se "lógica negativa" quando o estado alto - H representa a menor faixa de voltagem e baixo - L a maior. O exemplo da figura 1-a seria escrito em lógica negativa como na figura 1-4.

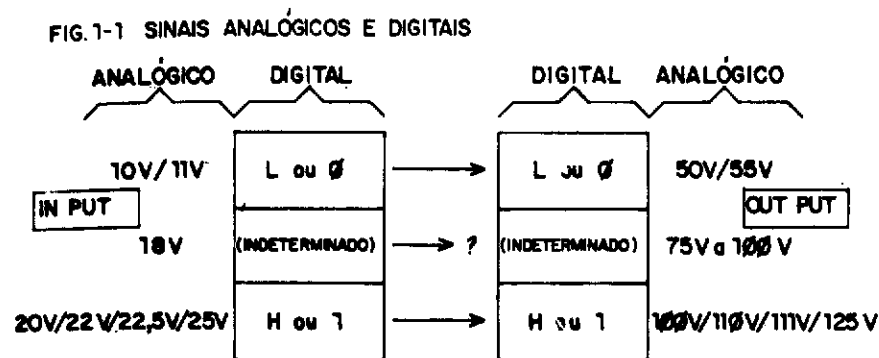


figura 1-2 - Notação de sinais digitais

Algumas vezes é necessário que o estágio se comporte como se não existisse dentro do circuito, em nada influenciando. Este é o chamado TRISTATE (terceiro estado). Apesar do nome não é realmente um terceiro estado. No circuito digital qualquer estágio só poderá assumir um dos dois estados - H ou L; se estiver em tristate pode se dizer que ele não pertencerá ao sistema.

O sinal digital em nível H (1) ou L (0) é denominado "BIT", contração de "binary digit" (dígito binário). Geralmente temos uma cadeia de bits sendo emitida ou recebida, o que se registra como 1001110, por exemplo.

O bit mais à direita (figura 1-5) é denominado "LSB" ("least significant bit" - bit menos significante) e aquele mais à esquerda "MSB" ("most significant bit" - bit mais significante).

VOLTAGEM IN PUT ALTO V_{ih}	MAIOR QUE 2,8 V	VOLTAGEM IN PUT BAIXO V_{il}	MENOR QUE 0,4 V
CORRENTE IN PUT ALTO I_{ih}	MAIOR QUE 1000 μ A	CORRENTE IN PUT BAIXO I_{il}	MENOR QUE -1,3 mA
VOLTAGEM OUT PUT ALTO V_{oh}	2,4V a 3V	VOLTAGEM OUT PUT BAIXO V_{ol}	MENOR QUE 0,4 V
CORRENTE OUT PUT ALTO I_{oh}	-0,7mA a 0	CORRENTE OUT PUT BAIXO I_{oh}	MENOR QUE 10mA

figura 1-3 - Níveis de sinal TTL

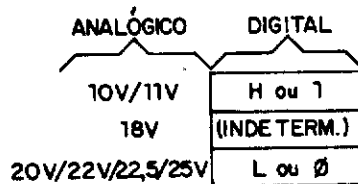


figura 1-4 - Lógica negativa

É comum representar uma sequência de bits como uma onda quadrada onde os picos equivalem a bit = 1 e os mínimos a bit = 0. O exemplo anterior seria representado graficamente como na figura 1-6.

Na verdade a sequência de bits é emitida

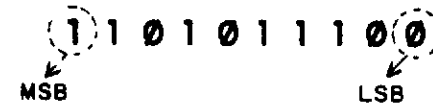


figura 1-5 - Posição do MSB e do LSB na cadeia de bits

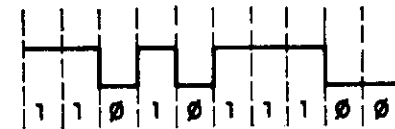


figura 1-6 - Representação de bits em forma de onda

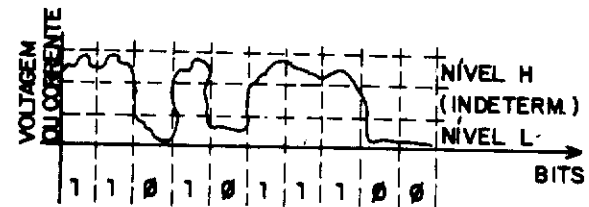


figura 1-7 - Forma de onda real da cadeia de bits

da ou recebida como um sinal elétrico bastante irregular, já que os estados lógicos abrangem uma faixa de níveis de voltagem e corrente. O exemplo da figura 1-6 poderia ser, na realidade, como registrado na figu-

ra 1-7.

A grande vantagem da Eletrônica Digital consiste exatamente nisso, conseguir dar grande precisão (1 ou 0) a sinais que por força dos meios físicos são necessariamente imprecisos mas limitados a uma certa faixa.

(2) FUNÇÕES LÓGICAS

O bit pode ser usado isoladamente, por exemplo para controlar uma máquina como na figura 1-8, que abre a chave S1 quando o bit = 1 ou a chave S2 quando bit = 0

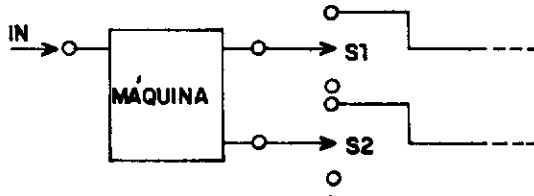


figura 1-8 - Bit de controle

Esse tipo de emprego da Eletrônica Digital não é importante e inclusive pode ser realizado pela Eletrônica Analógica comum. A sua utilização mais vantajosa ocorre quando o resultado de uma operação depende da combinação de vários fatores. Digamos, a chave S1 estará aberta quando as entradas IN1, IN3, IN4 estiverem em nível 1 e S2 estará aberta quando IN2 e IN5 estiverem em nível 1 (figura 1-9). Agora é preciso colocar após as entradas um circuito que combine-as adequadamente e apresente na sua saída um único bit que comandará a máquina.

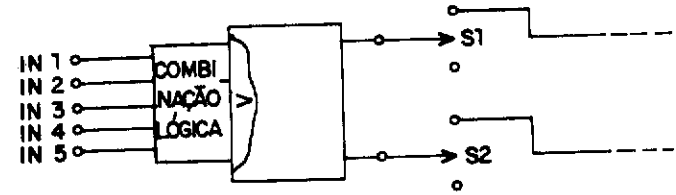


figura 1-9 - Conjunto de bits controlando uma máquina

Da lógica Matemática sabemos que com duas entradas e uma única saída, cada uma delas podendo assumir dois estados (1 ou 0), é possível construir 16 tipos diferentes de combinações. A Eletrônica Digital utiliza apenas algumas dessas combinações, que passam a ser chamadas "FUNÇÕES LÓGICAS" e são implementadas através de "PORTAS LÓGICAS". Neste capítulo estudaremos funções e portas do ponto de vista lógico-matemático e no capítulo 3 veremos como componentes eletrônicos transformam tais abstrações em realidade.

ENTRADAS		SAÍDA
A	B	A + B
1	1	1
0	1	1
1	0	1
0	0	0

+
OR

figura 1-10 - Tabela verdade da função OR

Chamamos TABELA VERDADE o diagrama que representa todas as possibilidades de níveis lógicos das entradas (inputs) no estágio e da saída (output), após processamento pela porta. Cada função tem sua tabela verdade, que a caracteriza.

ENTRADAS			SAÍDA
A	B	C	A or B or C A + B + C
1	1	1	1
0	1	1	1
1	0	1	1
0	0	1	1
1	1	0	1
0	1	0	1
1	0	0	1
0	0	0	0

figura 1-11 - Tabela verdade OR com 3 entradas

Na função OR (do inglês, "or" - ou) o sinal de saída será nível 1 se qualquer uma das entradas estiver em nível 1. A tabela verdade representando esta combinação é vista na figura 1-10, utilizando-se duas entradas - A e B. A saída A OR B também pode ser escrita A + B e só será 0 quando A = B = 0.

Para todas funções binárias o número de combinações possíveis com as entradas (linhas horizontais na tabela verdade) é dado pela fórmula

$$n^{\circ} \text{ combinações} = 2^N,$$

onde N é a quantidade de entradas atuando. No exemplo acima temos 2 entradas e daí

$$n^{\circ} \text{ de combinações} = 2^2 = 4.$$

Trabalhando com 3 entradas teremos uma tabela verdade com n^o de combinações = $2^3 = 8$, que é mostrada na figura 1-11.

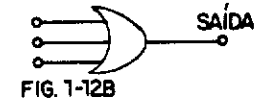
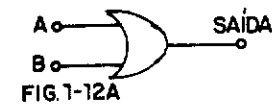


figura 1-12 - Símbolo da porta OR
(A) duas entradas (B) três entradas

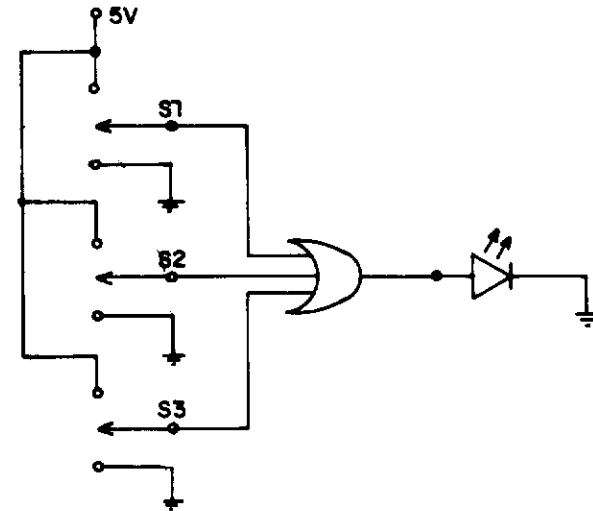


figura 1-13 Circuito com porta OR

Pela técnica mais comum de construir ta bela verdade segue-se a sequência:

- (1) calcula-se o número total de combinações
- (2) preenche-se a coluna da primeira entrada com 1 e

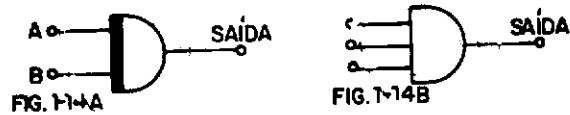


figura 1-14 - Símbolo da porta AND
(A) duas entradas (B) três entradas

- \emptyset , alternadamente, até atingir o número total de combinações.
- (3) preenche-se a coluna da segunda entrada com 1 e \emptyset , alternadamente de 2 em 2
 - (4) preenche-se a coluna da terceira entrada com 1 e \emptyset , alternadamente de 4 em 4

A função ou porta OR é indicada pelo símbolo da figura 1-12, podendo haver quantas entradas (inputs) forem necessárias, mas sempre uma só saída.

Um emprego típico da porta OR é mostrado na figura 1-13. Quando qualquer chave (S1, S2, S3) for comutada para 5V (nível 1) ocorrerá uma saída nível 1 na porta que acenderá o LED. Somente se todas as chaves estiverem aterradas (nível \emptyset) haverá uma saída \emptyset e o LED permanecerá apagado.

A função ou porta AND (do inglês, "and" - e) é simbolizada pela figura 1-14 e tem as tabelas verdade das figuras 1-15 e 1-16. A saída terá nível \emptyset quando qualquer uma das entradas estiver em nível \emptyset .

Na figura 1-17 temos uma aplicação simples de porta AND. Se pelo menos uma das chaves for aterrada (nível \emptyset) resultará saída em nível \emptyset , o que manterá o LED apagado. Se todas as chaves estiverem comutadas em 5V (nível 1) teremos saída nível 1 e o

ENTRADAS		SAÍDA
A	B	A • B A AND B
1	1	1
\emptyset	1	\emptyset
1	\emptyset	\emptyset
\emptyset	\emptyset	\emptyset



figura 1-15 - Tabela verdade da função AND

ENTRADAS			SAÍDA
A	B	C	A and B and C A . B . C
1	1	1	1
\emptyset	1	1	\emptyset
1	\emptyset	1	\emptyset
\emptyset	\emptyset	1	\emptyset
1	1	\emptyset	\emptyset
\emptyset	1	\emptyset	\emptyset
1	\emptyset	\emptyset	\emptyset
\emptyset	\emptyset	\emptyset	\emptyset

figura 1-16 - Tabela verdade AND com três entradas

LED acenderá.

A função ou porta NOT (do inglês, "not"-não), símbolo conforme figura 1-18 e tabela verdade 1-19, simplesmente muda o nível do bit na entrada, transformando 1 em \emptyset e \emptyset em 1. Essa transformação de

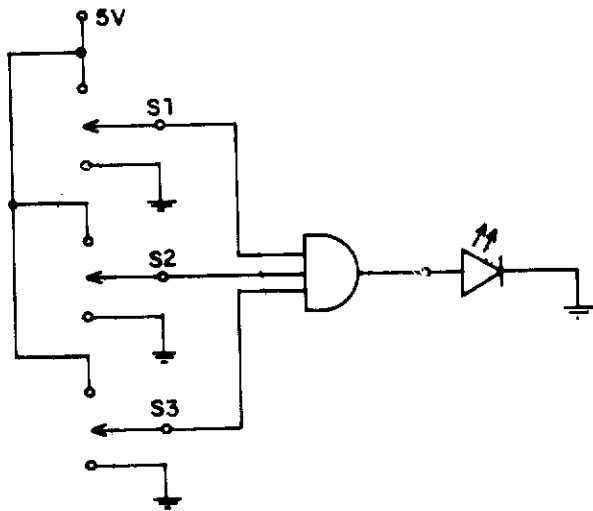


figura 1-17 - Circuito com porta AND

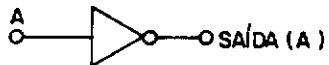


figura 1-18 - Símbolo da porta NOT

nível lógico pela porta NOT é indicada por uma barra a cima do símbolo da entrada, que é dito "complementado".

Na figura 1-20 vemos uma aplicação de porta NOT. Se o bit de entrada A estiver em nível 1 (5V) o LED 2 acenderá; ao mesmo tempo ele aparecerá com

ENTRADA	SAÍDA
A	\bar{A}
1	0
0	1

figura 1-19 - Tabela verdade da função NOT

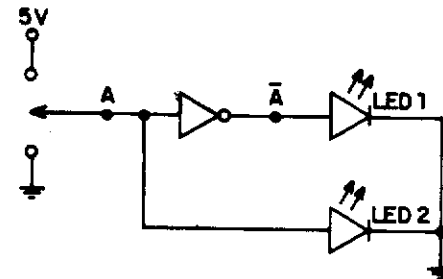


figura 1-20 - Circuito com porta NOT

plementado - nível 0 - no anodo do LED 1, mantendo-o apagado. Com $A = 0$ (aterrado) o LED 2 estará apagado e no anodo do LED 1 resultará $A = 1$, fazendo-o brilhar.

Para simplificar o desenho de circuitos

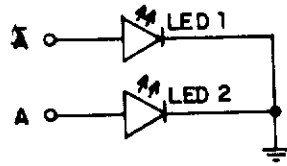


figura 1-21 - Desenho omitindo porta NOT

É comum omitir a porta NOT e mostrar o bit já complementado. O exemplo anterior seria desenhado como na figura 1-21. Deve-se então lembrar que quando $A = 1$, temos $\bar{A} = 0$ e para $A = 0$ resulta $\bar{A} = 1$.

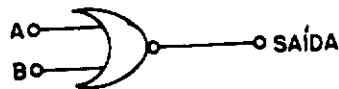


figura 1-22 - Símbolo da porta NOR

ENTRADAS		SAÍDAS	
A	B	$\bar{A} + \bar{B}$	$\overline{A + B}$
1	1	0	0
0	1	0	0
1	0	0	0
0	0	1	1

figura 1-23 - Tabela verdade da função NOR

A função NOT pode ser combinada com as anteriores, formando outras. Assim, NOR (NOT+OR) é a porta OR complementada, tendo seu símbolo e tabela verdade nas figuras 1-22 e 1-23. Observe que no símbolo basta acrescentar uma "bolinha" e que na tabela verdade a saída é exatamente o complemento da OR, inclusive podendo ser indicada por uma barra acima de $A + B$.

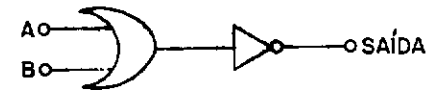


figura 1-24 - Desmembramento da porta NOR

O resultado de entradas numa NOR é o mesmo que combiná-las em OR e intercalar uma NOT antes da saída, conforme figura 1-24.

NAND (NOT+AND) é a complementação da função AND (figuras 1-25, 1-26, 1-27), e tem o mesmo efeito de uma NOT antes da saída de uma AND. A tabela verdade é o complemento da saída AND, podendo ser re

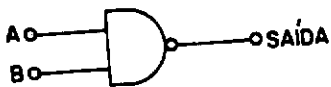


figura 1-25 - Símbolo da porta NAND

ENTRADAS		SAÍDA
A	B	$\overline{A \cdot B}$
1	1	0
0	1	1
1	0	1
0	0	1

figura 1-26 - Tabela verdade da função NAND

gistrada por uma barra acima de A.B.

A função ou porta EOR (do inglês, "exclusive or" - ou exclusivo), tem o símbolo da fig. 1-28. Também é comum representá-la por 'XOR'. Na tabela verdade (figura 1-29) observa-se que a saída é 0 quando ambas as entradas tem o mesmo nível.

Ao operar uma porta EOR com três (ou mais) entradas raciocinamos como se houvesse uma primeira combinação com duas entradas ($A \oplus B$) e o resultado combinado com a terceira entrada (saída de $A \oplus B \oplus C$). A figura 1-30 apresenta o processo, com a respectiva tabela verdade.

Qualquer que seja o número de entradas na EOR é válida a seguinte regra:

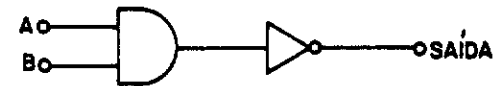


figura 1-27 - Desmembramento da porta NAND



figura 1-28 - Símbolo da porta EOR.

ENTRADAS		SAÍDA
A	B	$A \oplus B$
1	1	0
0	1	1
1	0	1
0	0	0

figura 1-29 - Tabela verdade da função EOR

* se o número de entradas nível 1 for ímpar a saída terá nível 1, se for par ela será 0. Alguns exemplos, tomando A=1, B=1, C=0, D=0, E=1, F=1



FIG. 1-30A

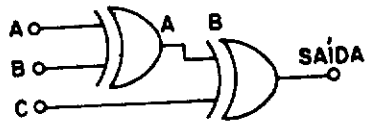


FIG. 1-30B

ENTRADAS			SAÍDA
A	B	C	$A \oplus B \oplus C$
1	1	1	1
0	1	1	0
1	0	1	0
0	0	1	1
1	1	0	0
0	1	0	1
1	0	0	1
0	0	0	0

figura 1-30 - Porta EOR (A) com três entradas
(B) desmembrada em duas portas

- (1) $A \oplus B \oplus C = 0$ (par)
- (2) $A \oplus B \oplus C = 1$ (ímpar)
- (3) $A \oplus B \oplus C \oplus D = 0$ (par)
- (4) $B \oplus C \oplus D = 1$ (ímpar)

Um circuito empregando porta EOR é mostrado na figura 1-31. Quando as duas chaves (S1 e S2) estão simultaneamente em nível 1 (5V) ou 0 (aterradas) aparecerá um nível 0 no anodo do LED, que permanecerá apagado. Entretanto, se uma das chaves for comutada para 5 V (nível 1) enquanto a outra estiver aterrada (nível 0) resultará sinal de nível 1 na saída da porta que acenderá o LED.

As portas lógicas são implementadas através de circuitos integrados que contêm várias de-

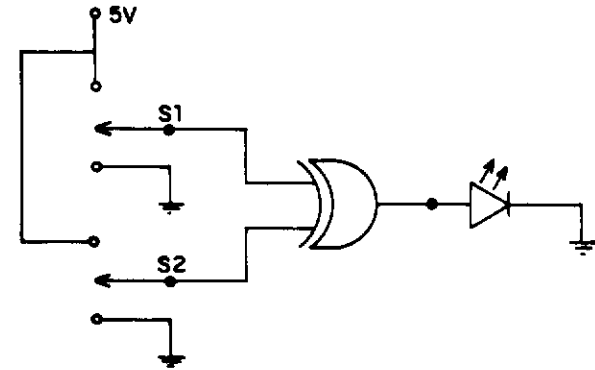


figura 1-31 - Tabela verdade da função EOR
com três entradas

las. As duas famílias lógicas mais populares - TTL e CMOS (que estudaremos nos próximos capítulos) - são numeradas respectivamente 74XX e 40XX, onde XX representa dois números que determinam quais funções estão contidas no CI. Na figura 1-32 temos o CI 7486, com 4 portas EOR, e o CI 4049 com 6 portas NOT. Qualquer que seja o fabricante, CIs com a mesma numeração são equivalentes e contêm as mesmas portas.

(3) COMBINAÇÕES LÓGICAS

As portas lógicas podem ser combinadas umas com as outras, formando cadeias e sequências. Entende-se que o nível da saída é calculado de acordo com

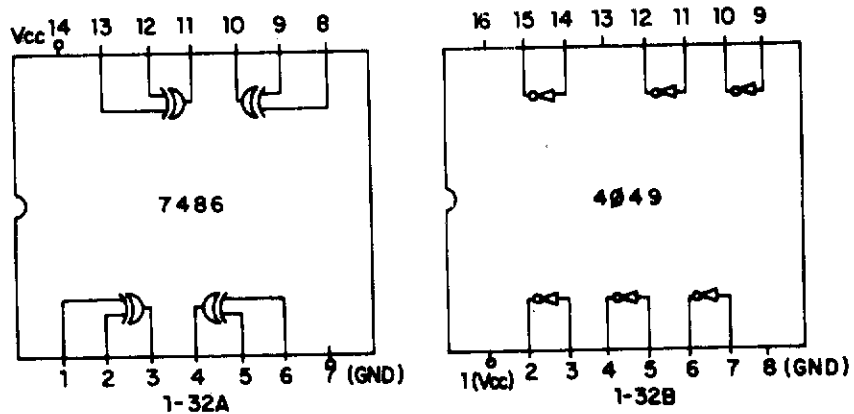


figura 1-32 CI com portas lógicas (A) 4 EOR
(B) 6 NOT

a Lógica Matemática, usando-se a Álgebra Booleana. Na prática, entretanto - tanto em nível técnico quanto em trabalhos mais sofisticados - raramente se justifica lançar mão desses complexos recursos teóricos, exceto de uns poucos teoremas que descrevemos a seguir com os



figura 1-33 - $A + \emptyset = A$

respectivos símbolos (figuras 1-33 a 1-40).



figura 1-34 - $A + 1 = 1$



figura 1-35 - $A + A = A$

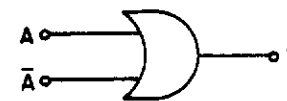


figura 1-36 - $A + \bar{A} = 1$

(a letra A indica o nível lógico do bit de entrada - \emptyset ou 1 - devendo ser substituído por ele).
A solução mais simples e rápida para

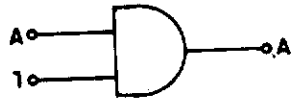


figura 1-37 - $A \cdot 1 = A$



figura 1-38 - $A \cdot 0 = 0$



figura 1-39 - $A \cdot A = A$



fig. 1-40 - $A \cdot \bar{A} = 0$

cadeias de portas lógicas é substituir cada uma, graficamente, pelo seu bit de saída até chegar ao final. Mostremos alguns exemplos, admitindo em todos eles $A=1$, $B=0$, $C=1$, $D=0$.

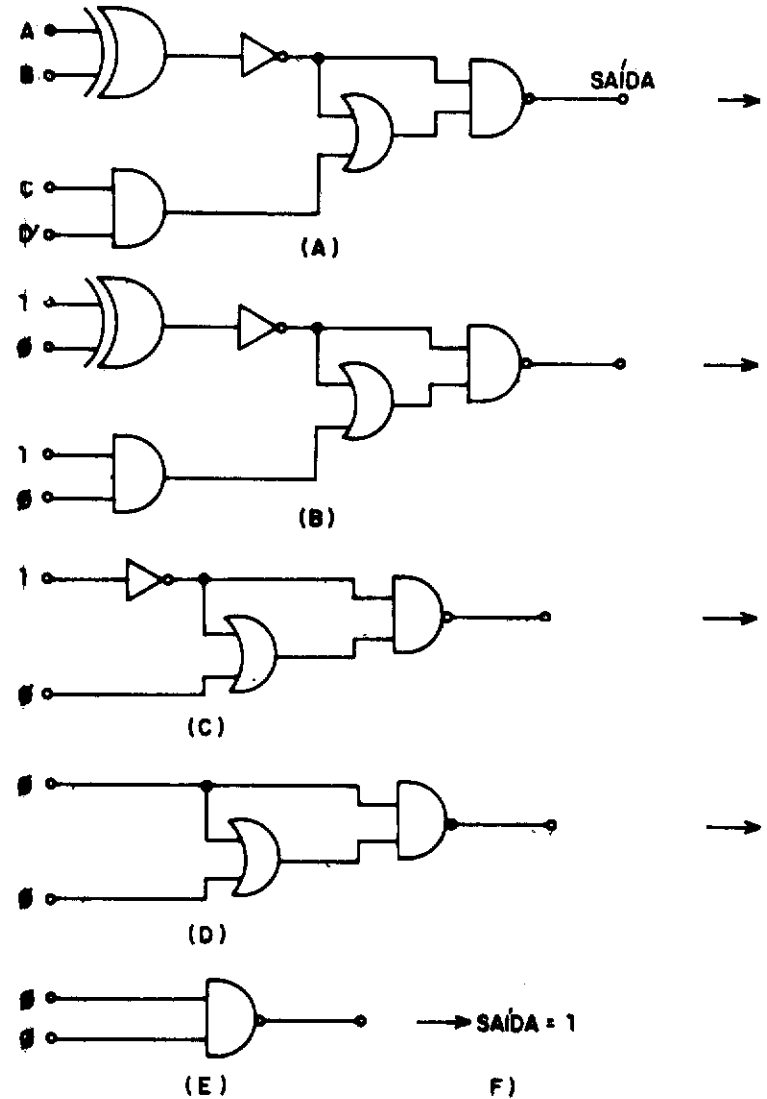


figura 1-41 - $[(A \oplus B) + (C \cdot D)] \cdot [A \oplus B]$ (contínua)

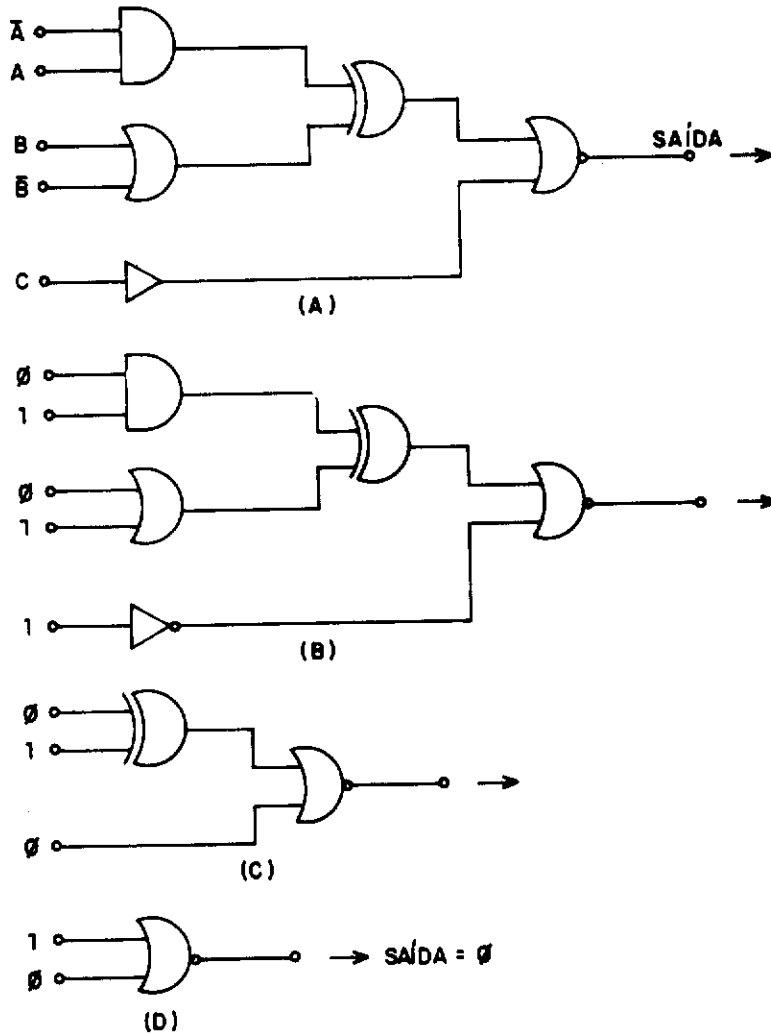


figura 1-41 - (continuação)

CAPÍTULO 2 – ARITMÉTICA DIGITAL

Dispositivos eletrônicos digitais reconhecem apenas dois estados nos sinais elétricos - 1 ou \emptyset . A aritmética que emprega somente dois números denominados "binários" - surge naturalmente como a mais adequada para Eletrônica Digital.

Estudaremos os números binários, suas relações com os decimais e diversos códigos criados para manipulá-los. Operações aritméticas com binários, incluindo processos lógico-digitais e dispositivos eletrônicos de cálculo, serão descritos e exemplificados.

(1) NÚMEROS BINÁRIOS

No sistema decimal (usado no dia-a-dia) temos 10 números ($\emptyset, 1, 2, \dots, 9$) e valores acima do décimo são representados pela posição. Ou seja:

$$3528 = (\underline{3} \times 1000) + (\underline{5} \times 100) + (\underline{2} \times 10) + (\underline{8} \times 1)$$

Lembrando que uma potência equivale ao número base multiplicado por si mesmo tantas vezes quanto for o expoente, e qualquer número elevado a \emptyset é igual a 1:

$$10^{\emptyset} = 1, 5^{\emptyset} = 1, 328956^{\emptyset} = 1,$$

podemos reescrever o exemplo acima:

$$3528 = (\underline{3} \times 10^3) + (\underline{5} \times 10^2) + (\underline{2} \times 10^1) + (\underline{8} \times 10^{\emptyset}).$$

Um outro exemplo:

$$947 = (\underline{9} \times 10^2) + (\underline{4} \times 10^1) + (\underline{7} \times 10^{\emptyset})$$

Como se observa a posição do número decimal indica por qual potência de 10 ele deve ser multiplicado, contando-se essa da direita para a esquerda

a partir de \emptyset

$$(_x1\emptyset^5)+(_x1\emptyset^4)+(_x1\emptyset^3)+(_x1\emptyset^2)+(_x1\emptyset^1)+(_x1\emptyset^0).$$

Números binários também recebem o mesmo tratamento, mas utilizando a base 2:

$$(_x2^5)+(_x2^4)+(_x2^3)+(_x2^2)+(_x2^1)+(_x2^0).$$

Devido ao extenso uso de potências de 2 em eletrônica Digital apresentamos uma tabela com os primeiros valores. O leitor ganhará muito se memorizá-la desde já.

$2^0 = 1$	$2^4 = 16$	$2^8 = 256$	$2^{12} = 4.096$
$2^1 = 2$	$2^5 = 32$	$2^9 = 512$	$2^{13} = 8.192$
$2^2 = 4$	$2^6 = 64$	$2^{10} = 1.024$	$2^{14} = 16.384$
$2^3 = 8$	$2^7 = 128$	$2^{11} = 2.048$	$2^{15} = 32.768$

Como sã temos dois números disponíveis, todas as posições devem ser preenchidas por 1 ou \emptyset . Exemplo:

$$(_x2^5)+(_x2^4)+(_x2^3)+(_x2^2)+(_x2^1)+(_x2^0) =$$

$$(_x32)+(_x16)+(_x8)+(_x4)+(_x2)+(_x1) =$$

$$32 + 16 + \emptyset + 4 + \emptyset + 1 = 53 \text{ (decimal)}$$

Tal como nos decimais podemos omitir a base 2 dos binários e subentendê-la pela posição. O exemplo anterior ficaria:

$$11\emptyset1\emptyset1.$$

Para caracterizar o tipo de numeração escreve-se um '2' como índice dos binários ou então põe-se um B. Quando não houver qualquer índice ou letra trata-se de decimal. Continuando com o exemplo aci

ma:

$$11\emptyset1\emptyset1_2 = 11\emptyset1\emptyset1_B = 53$$

Abaixo seguem mais alguns exemplos:

DECIMAL	BINÁRIO
\emptyset	$(\emptyset x 2^0) = \emptyset_B$
1	$(1 x 2^0) = 1_B$
2	$(1 x 2^1) + (\emptyset x 2^0) = 1\emptyset_B$
3	$(1 x 2^1) + (1 x 2^0) = 11_B$
4	$(1 x 2^2) + (\emptyset x 2^1) + (\emptyset x 2^0) = 1\emptyset\emptyset_B$
5	$(1 x 2^2) + (\emptyset x 2^1) + (1 x 2^0) = 1\emptyset1_B$
6	$(1 x 2^2) + (1 x 2^1) + (\emptyset x 2^0) = 11\emptyset_B$
7	$(1 x 2^2) + (1 x 2^1) + (1 x 2^0) = 111_B$
8	$(1 x 2^3) + (\emptyset x 2^2) + (\emptyset x 2^1) + (\emptyset x 2^0) = 1\emptyset\emptyset\emptyset_B$
9	$(1 x 2^3) + (\emptyset x 2^2) + (\emptyset x 2^1) + (1 x 2^0) = 1\emptyset\emptyset1_B$
10	$(1 x 2^3) + (\emptyset x 2^2) + (1 x 2^1) + (\emptyset x 2^0) = 1\emptyset1\emptyset_B$
20	$(1 x 2^4) + (\emptyset x 2^3) + (1 x 2^2) + (\emptyset x 2^1) + (\emptyset x 2^0) = 1\emptyset1\emptyset\emptyset_B$

Eletricamente, os bits 1 e \emptyset são implementados por nível de voltagem (e corrente) alto e baixo, respectivamente. Por essa convenção o circuito da figura 2-1 está indicando o número $1\emptyset1_B = 5$.

Observe que se do ponto de vista humano a numeração binária parece mais complicada, para máquinas eletro-eletrônicas que conseguem distinguir apenas dois estados (ligado x desligado, aceso x apagado, voltagem alta x voltagem baixa; esta é a linguagem ideal.

(2) Conversão de decimal em binário

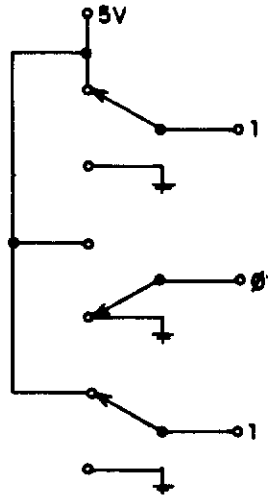


figura 2-1 - Binário 101 implementado eletricamente

Para transformar um número binário em decimal basta multiplicar o bit (1 ou 0) pela potência de 2 correspondente à posição e somar todos os produtos, como fizemos acima. A operação inversa, conversão de decimal em binário, exige um esquema especial.

Tomaremos o decimal 19 como exemplo (figura 2-2). Inicialmente ele é posto na parte superior à direita, traçando-se quadrículas de separação (figura 2-2A). Efetua-se uma divisão do decimal por 2, registrando-se o resultado (9) na próxima quadrícula superior à esquerda e o resto (1) embaixo dele (figura 2-2B). O resultado obtido é tomado como dividendo para uma nova divisão por 2, repetindo-se o processo de registro do novo resultado (4) e resto (1) como no an

					19 — DECIMAL
					2-2A
RESULTADO DA DIVISÃO POR 2			9	19	
RESTO DA DIVISÃO			1		2-2B
			4	9	19
			1	1	2-2C
		2	4	9	19
		0	1	1	2-2D
	1	2	4	9	19
	0	0	1	1	2-2E
BINÁRIO —	1	0	0	1	1
					2-2F

19 = 100118
2-2G

figura 2-2 - Processo de conversão de decimal em binário

terior (figura 2-2C). A operação é repetida (figuras 2-2D, E, F), com sucessivas divisões por 2 até que se chegue ao resultado \emptyset (lembre-se que $1 + 2 = \emptyset$ e resto = 1). Os restos anotados na parte inferior formam o binário procurado, no caso $19 = 10011B$.

Para conferir faça a conversão binário em decimal:

$$10011B = (1 \times 2^4) + (\emptyset \times 2^3) + (\emptyset \times 2^2) + (1 \times 2^1) + (1 \times 2^0) =$$

$$= 16 + \emptyset + \emptyset + 2 + 1 = 19$$

A conversão de decimal fracionário em binário é mais difícil, e na verdade raramente é necessária. Utilizando o decimal 0,8125 como exemplo, colocamo-lo no canto superior esquerdo e obteremos o binário na parte inferior, que deve começar por "0", (figura 2-3A). Na próxima quadrícula o decimal é multiplicado por 2 e se o resultado for maior que 1 subtraí-se 1; registra-se 1 na parte inferior da quadrícula e lança-se o resultado da subtração (0,625) na próxima (figura 2-3B). A operação é repetida a partir da nova quadrícula (figura 2-3C) observando-se que se na subtração o número é menor que 1 (como na figura 2-3D, "0,5 - 1 = ...") ela não é efetuada, registrando-se \emptyset na parte inferior e lançando o decimal não subtraído na próxima quadrícula. A conversão termina quando se obtém uma subtração com resultado \emptyset (figura 2-3E). A resposta final estará anotada na parte inferior, no caso $0,8125 = 0,1101B$.

(3) ADIÇÃO DE BINÁRIOS

Na adição de binários segue-se processo semelhante à de decimais (figura 2-4). Nestes, quando a adição de uma casa é maior que 9 lançamos o "vai 1" na próxima. Com binários o "vai 1" é conhecido por CARRY e ocorre quando a adição da casa excede 1. Nas figuras 2-4A até 2-4F ilustramos o processo, passo a passo. Veja pela figura 2-4G que o resultado

DECIMAL - 0,8125				
BINÁRIO - 0,	2-3A			
0,8125	0,8125x2 = 1,625 1,625-1 = 0,625	0,625		
0,	1	2-3B		
0,625	0,625x2 = 1,25 1,25-1 = 0,25	0,25		
0,	1	1	2-3C	
0,25	0,25x2 = 0,50 0,5-1 = ...	0,5		
0,	1	1	\emptyset	2-3D
DECIMAL - 0,8125	0,8125x2 = 1,625 1,625-1 = 0,625	0,625x2 = 1,25 1,25-1 = 0,25	0,25x2 = 0,50 0,5-1 = ...	0,5x2 = 1 1-1 = \emptyset
BINÁRIO - 0,	1	1	\emptyset	1
2-3E				
$0,8125 = 0,1101B$				
2-3F				

figura 2-3 - Processo de conversão de decimal fracionário em binário

$$\begin{array}{r}
 1110 \\
 1011 \\
 \hline
 \end{array} + \quad 2-4A$$

$$\begin{array}{r}
 1110 \\
 1011 \\
 \hline
 1
 \end{array} \quad 2-4B$$

$$\begin{array}{r}
 \textcircled{1} \leftarrow \text{CARRY} \\
 1110 \\
 1011 \\
 \hline
 01
 \end{array} \quad 2-4C$$

$$\begin{array}{r}
 \textcircled{1}\textcircled{1} \\
 1110 \\
 1011 \\
 \hline
 001
 \end{array} \quad 2-4D$$

$$\begin{array}{r}
 \textcircled{1}\textcircled{1}\textcircled{1} \\
 1110 \\
 1011 \\
 \hline
 1001
 \end{array} \quad 2-4E$$

$$\begin{array}{r}
 1110 \\
 1011 \\
 \hline
 11001
 \end{array} \quad 2-4F$$

BINARIO	DECIMAL
1110	→ 14
1011	→ 11
11001	→ 25

2-4G

figura 2-4 - Adição de binários

independe do tipo de números usados, isto é, convertemos os binários em decimais e efetuando a soma obtem-se uma resposta equivalente à soma em binários.

Eletronicamente soma-se binários dígito por dígito, seguindo-se a tabela verdade da figura 2-5. Analisando-a deduz-se que a soma dos dois dígitos (A+B) corresponde à tabela verdade da função EOR e o carry a AND. Ou seja (figura 2-6A)

DÍGITOS		SOMA	
A	B	A + B	CARRY
1	1	0	1
0	1	1	0
1	0	1	0
0	0	0	0

figura 2-5 - Tabela verdade de adição de binários

$$\begin{aligned}
 \text{SOMA} &= A \oplus B \\
 \text{CARRY} &= A \cdot B
 \end{aligned}$$

Dois exemplos são apresentados nas figuras 2-6B e 2-6C. Um circuito somador operando dessa maneira é denominado H.A. - HALF ADDER (do inglês, meio somador) e simbolizado pela figura 2-6D.

Uma máquina que operasse com meio - soma-

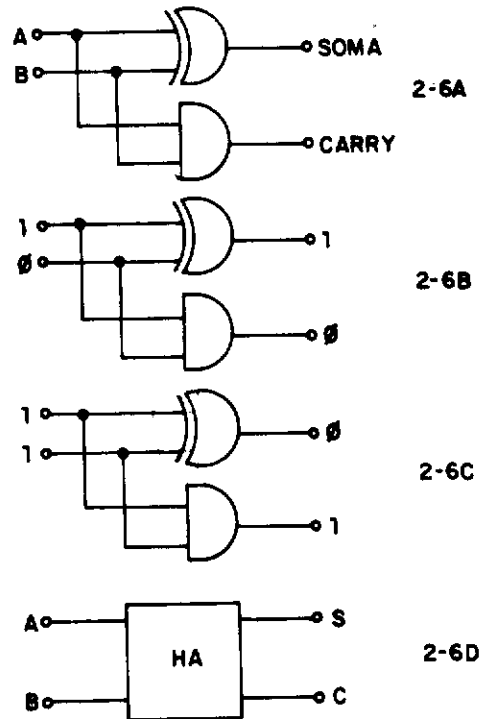


figura 2-6 - Implementação lógica da soma de binários (A) circuito lógico (B) 1+0 (C) 1+1 (D) símbolo meio somador

dores teria problemas, pois precisamos também adicionar o carry da casa anterior. F.A. - FULL ADDER (do inglês, somador completo) é um circuito que adiciona os dois dígitos mais o carry anterior, seguindo a tabela da figura 2-7. Observamos que a soma é a combinação numa porta EOR dos dois dígitos mais o carry anterior e produz um carry dado pela expressão abaixo:

$$\text{SOMA} = A \oplus B \oplus \text{CARRY ANTERIOR}$$

$$\text{CARRY} = (A \cdot B) + [(A + B) \cdot \text{CARRY ANTERIOR}]$$

O circuito que implementa um somador completo bem como o seu símbolo são mostrados nas figuras 2-7B e 2-7C. O leitor que substituir as entradas A, B e CARRY ANTERIOR por níveis lógicos obterá na saída o mesmo resultado da tabela da figura 2-7A.

De posse desse dispositivo torna-se fácil projetar uma somadora eletrônica de números binários (figura 2-8). Os bits de mesma posição, começando à direita pelo LSB, são somados dois a dois tal como no processo manual. O carry ("vai um") de uma casa é conectado ao somador seguinte, exceto o primeiro que não o tem, sendo aterrado (0). O resultado da soma será o número $C_3 S_3 S_2 S_1 S_0$.

Na figura 2-9 executamos eletronicamente a soma $1110 + 1011$ feita anteriormente por princípios aritméticos. A tabela da figura 2-7A fornece as saídas de cada F.A. Note ainda que F.A.0 corresponde à operação da figura 2-4B, F.A.1 a 2-4C, F.A.2 a 2-4D, F.A.3 a 2-4E e 2-4F.

Quando o último carry é 1 dizemos que há um "overflow". No nosso exemplo ocorreu overflow, devido ao carry do F.A.3.

Considerando que uma simples chave é capaz de emitir sinais lógicos 1/0 por controle manual e que todas as operações aritméticas podem ser reduzidas à adição, como veremos adiante, chegamos à enorme uti-

DÍGITOS		CARRY ANTERIOR	SOMA	
A	B		A + B + CARRY ANT.	CARRY
1	1	1	1	1
0	1	1	0	1
1	0	1	0	1
0	0	1	1	0
1	1	0	0	1
0	1	0	1	0
1	0	0	1	0
0	0	0	0	0

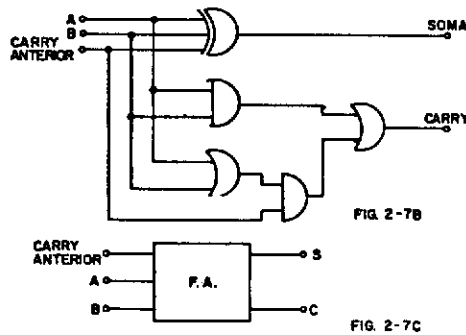


figura 2-7 - Somador completo (A) tabela verdade, (B) circuito lógico (C) símbolo

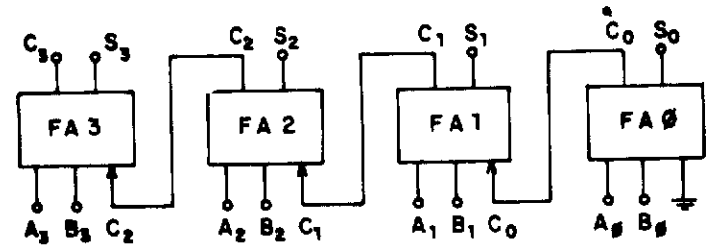


figura 2-8 - Somadora eletrônica

lidade do esquema estudado. Na verdade podemos construir uma sofisticada calculadora eletrônica com ele.

(4) SUBTRAÇÃO DE BINÁRIOS

A subtração de binários é realizada convertendo o número a ser subtraído em negativo e somando-o ao outro. Há dois tipos de binários negativos - Complemento 1 (N_1^*) e Complemento 2 (N_2^*). Nos dois casos o MSB é 1 para indicar número negativo (e 0 para positivo).

O complemento 1 é obtido pela complementação de cada bit (conversão de 1 em 0 e 0 em 1). Abaixo alguns exemplos, destacando o MSB indicador de sinal.

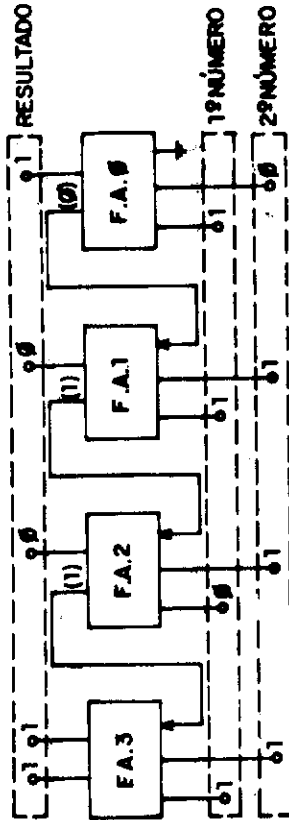


figura 2-9 - Soma eletrônica de 1110+1011

POSITIVO	NEGATIVO
+ 9 = 0 1 0 0 1 B	1 0 1 1 1 0 B = - 9
+14 = 0 1 1 1 0 B	1 0 0 0 1 B = -14
+ 2 = 0 1 0 B	1 0 1 B = - 2

Ao identificar um binário negativo complemento 1 pela presença de MSB=1 e para conhecer seu valor decimal basta complementar todos seus bits. Por exemplo: 101101B é binário negativo porque o MSB é 1; Complementando todos bits resulta 010010B= + 18; logo 101101B= - 18.

Na subtração em complemento 1 o número a ser subtraído é convertido em negativo, repetindo -se o MSB, adicionado ao outro; se houver overflow ele também é somado ao total. Resultado com MSB= 0 indicará resposta positiva e MSB= 1 negativa. Abaixo exemplo de operações em complemento 1:

+ 5 = 0101B	00101B;	- 5=1010B	11010B
+ 7 = 0111B	00111B;	- 7=1000B	11000B
+7 00111B	+7 00111B	- 7 11000B	- 7 11000B
-5 11010B	+5 00101B	- 5 11010B	+ 5 00101B
+2 100001	+12 01100B	-12 110010	- 2 11101B
	1	1	
	00010B	10011B	

No complemento 2 (N₂^{*}) todos os bits são complementados e ao resultado é somado + 1. Exemplos:

+ 5 = 0101;	complementando os bits: 1010;	somando 1:
:	1011B	
+ 7 = 0111;	complementando os bits; 1000;	somando 1:
:	1001B	

Uma maneira mais prática de se obter o complemento 2 é manter os bits a partir de LSB até a parecer o primeiro 1 e complementar os outros a partir dele:

+ 6 = 0110; os dois primeiros bits da direita para a

esquerda são mantidos e os dois outros são complementados:

$$0110 \rightarrow 1010B.$$

Para verificar o valor de um número negativo complemento 2, identificado pelo MSB=1, usa-se a mesma regra prática citada acima:

$$10010B \rightarrow 01110B = 14; \text{ deduz-se então que } 10010B = -14$$

As operações aritméticas são iguais às do complemento 1, porém se houver overflow ele é desprezado. Seguem alguns exemplos:

$$\begin{array}{r} + 7 = 0111B \text{ --- } 00111B; -7 = 1001B \text{ --- } 11001B \\ + 4 = 0100B \text{ --- } 00100B; -4 = 1100B \text{ --- } 11100B \\ \\ + 7 \quad 00111B \quad +7 \quad 00111B \quad -7 \quad 11001B \quad -7 \quad 11001 \\ + 4 \quad 00100B \quad -4 \quad 11100B \quad +4 \quad 00100B \quad -4 \quad 11100 \\ +11 \quad 01011B \quad +3 \quad 10001B \text{ ---} \quad -3 \quad 11101B \quad -11 \quad 11010B \\ \quad \quad \quad 00011B \quad \quad \quad 10101 \end{array}$$

Eletronicamente as operações com complementos são realizadas em circuitos de somador completo F.A. vistos anteriormente. Apesar de mais complicado o complemento 2 fornece um resultado mais fácil de manipular, simplesmente desprezando o overflow.

Somando-se + 1 ao complemento $1(N_1^*)$ resulta o complemento 2:

$$-7 (N_1^*) = 1000B; \text{ somando } + 1 : 1001B = -7 (N_2^*).$$

$$-4 (N_1^*) = 1011B; \text{ somando } + 1 : 1100B = -4 (N_2^*).$$

(5) SCALING

A multiplicação é efetuada por adições sucessivas, ou seja, o número é somado consigo mesmo tantas vezes quanto for o multiplicador. Igualmente, uma divisão é realizada por subtrações sucessivas até se atingir um dividendo menor que o divisor.

Mais importante é a multiplicação ou divisão de número binário por uma potência de 2, o que se denomina "scaling", de larga aplicação em cali-

bração de instrumentos de medição e controle.

Como a posição do binário determina por qual potência de 2 ele é multiplicado basta deslocá-lo uma casa para a esquerda para multiplicá-lo por 2, duas casas na multiplicação por 4, três nas multiplicações por oito, e assim por diante na seqüência de potências de 2, conforme exemplo abaixo.

2^5	2^4	2^3	2^2	2^1	2^0	
			1	0	1	= 5
		1	0	1	0	= $10 (5 \times 2^1)$
	1	0	1	0	0	= $20 (5 \times 2^2)$
1	0	1	0	0	0	= $40 (5 \times 2^3)$

Inversamente, deslocando o binário para a direita teremos uma divisão por 2^n , onde n é o número de casas deslocadas (a fórmula também é válida para a multiplicação):

2^5	2^4	2^3	2^2	2^1	2^0	
1	1	0	0	0	0	= 48
	1	1	0	0	0	= $24 (48 \div 2^1)$
		1	1	0	0	= $12 (48 \div 2^2)$

(6) CÓDIGOS BINÁRIOS

Uma seqüência de 8 bits é denominada BYTE (lê-se "baíte"). Exemplos: 1 0 0 1 1 1 0 1 B, 0 0 0 1 1 1 1 0 B, 1 1 1 1 1 1 1 1 B. Processadores digitais geralmente manipulam 1 byte de cada vez, isto é, recebem ou emitem 8 bits simultaneamente.

A palavra Kilo (do grego, "Kilo" - mil) em matemática indica o número 1.000 (Kilometro = 1.000 metros; Kilograma - 1.000 gramas). Em Eletrônica Digital, entretanto, Kilo representa $2^{10} = 1.024$. Assim, 1 Kilo bits = 1Kb = 1.024 bits e 1 Kilobytes = 1KB = 1024 bytes.

A linguagem binária é adequada para máquinas, mas algo inconveniente para humanos. Criaram-se, assim, vários códigos que convertem os binários em outras representações mais simples e inteligíveis.

No Código HEXADECIMAL cada grupo de quatro binários é representada pelo seu valor decimal. Como quatro binários podem atingir até o valor 15 (1111B = 15) e para evitar números de mais de 1 dígito (10, 11, 12, 13, 14, 15) acrescenta-se letras aos hexadecimais a partir de 10. O índice 16 ou a letra H é posposta ao número hexadecimal: 11H = 11₁₆). Alguns exemplos:

DECIMAL	BINÁRIO	HEXADECIMAL
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	0001 0000	1 0
17	0001 0001	1 1
20	0001 0100	1 4

DECIMAL	BINÁRIO	HEXADECIMAL
80	0101 0000	5 0
83	0101 0011	5 3
255	1111 1111	F F
256	0001 0000 0000	1 0 0
257	0001 0000 0001	1 0 1
428	0001 1010 1100	1 A C

Note que 1 byte contém dois hexadecimais:

$$\underline{1001\ 1101}B = \underline{9\ D}H$$

Como processadores e memórias trabalham com bytes é comum encontrar longas listas de instruções e posições em hexadecimais. Deve ser lembrado, porém, que essa numeração é apenas uma facilidade para o operador humano; a máquina trabalha realmente com números binários implementados por sinais elétricos do tipo alto x baixo (voltagem e corrente).

O código BCD ("binary coded decimal" - decimal codificado em binário) faz o inverso dos hexadecimais. Cada decimal é representado por quatro binários:

DECIMAL	B C D	DECIMAL	B C D
0	0000	10	0001 0000
1	0001	11	0001 0001
2	0010	20	0010 0000
3	0011	25	0010 0101
4	0100	43	0100 0011
5	0101	126	0001 0010 0110
6	0110	300	0011 0000 0000
7	0111	301	0011 0000 0001
8	1000		
9	1001		

Nos códigos estudados até agora a passagem de um número para o seguinte, na sequência, geralmente exige a alteração de vários bits. De 11 para 12

em binários, por exemplo, temos que alterar 3 bits:

1011B — 1100B (alteração dos 3 bits a partir de LSB)

Como dispositivos eletrônicos na maioria das vezes empregam os números em sequência e para evitar desperdício de operações em muitas alterações criou-se o CÓDIGO REFLETIVO DE GREY. Nele a passagem de um número para o próximo provoca mudança de um só bit. A montagem desse código é vista abaixo:

- (1) os dois primeiros números são escritos normalmente

<u>DECIMAL</u>	<u>GREY</u>
0	0
1	1

- (2) marca-se uma linha pontilhada ("espelho de reflexão") abaixo do último número de Grey e repete-se a sequência já escrita de trás-para-a-frente, como se fosse a imagem do espelho

<u>DECIMAL</u>	<u>GREY</u>
0	0
1	1

	1
	0

- (3) na parte superior da linha pontilhada antecede-se cada número com "0"; na parte inferior com "1". Os números são a continuada da sequência no código de Grey.

<u>DECIMAL</u>	<u>GREY</u>
0	0 0
1	0 1

2	1 1
3	1 0

- (4) os itens 2 e 3 são repetidos até se atingir o valor desejado

<u>DECIMAL</u>	<u>GREY</u>	<u>DECIMAL</u>	<u>GREY</u>
0	0 0	0	0 0 0
1	0 1	1	0 0 1
2	1 1	2	0 1 1
3	1 0	3	0 1 0
	-----		-----
	1 0	4	1 1 0
	1 1	5	1 1 1
	0 1	6	1 0 1
	0 0	7	1 0 0

É fácil verificar que a cada passo na sequência altera-se um único bit.

Letras e outros símbolos para redação de textos e comandos de operação também são codificados em binários, já que esta é a única linguagem entendida pelos dispositivos digitais. O CÓDIGO ASCII ("american standard code for information interchange" - código padrão americano para troca de informação) é o mais popular nesta categoria, sendo extensamente usado para codificação de teclado e de terminal de vídeo. Assim, ao apertar uma tecla no microcomputador o toque é convertido num conjunto de bits ASCII que circulam na barra de dados.

O código ASCII completo é apresentado no fim do livro, em hexadecimal (os primeiros caracteres se referem a comandos), compondo-se cada carácter com 7 bits.

Uma técnica muito usada para transformar valores numéricos do ASCII em decimal é subtrair 30, como na tabela abaixo;

ASCII	(HEXA)	DECIMAL
30	30-30 =	0
31	31-30 =	1
32	32-30 =	2
33	33-30 =	3
34	34-30 =	4
35	35-30 =	5
36	36-30 =	6
37	37-30 =	7
38	38-30 =	8
39	39-30 =	9

Na transmissão e recepção de números e códigos binários pode ser acrescentado um último bit de conferência. Na PARIDADE PAR o bit de acréscimo torna par a soma de todos os bits da cadeia. Por exemplo, para transmissão em paridade par do decimal 5 em ASCII teríamos 010 0011 cuja soma é 3 (0+1+0+0+0+1+1), ímpar, acrescentamos então o 8º bit 1, transformando a soma em par 1010 0011. Tomando o decimal 3 em ASCII (010 0001) vemos que a soma de seus bits já é par; logo na paridade par é acrescentado o 8º bit 0, para que assim continue (0010 0001). A PARIDADE ÍMPAR segue os mesmos princípios acima, mas o bit de conferência é acrescentado para tornar ímpar a soma da cadeia.

(7) 7-LED DISPLAY

O mais popular efeito digital é o display numérico (figura 2-10), cujo visual é imediatamente associada à Eletrônica Digital.

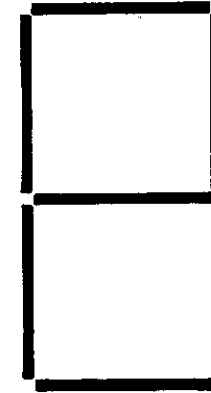


figura 2-10 - 7 LED display

Consiste de 7 LEDs (daí o nome inglês "seven LEDs display") de maneira a formar um '8'. Conforme os LEDs em condução, brilhando, forma-se um número decimal. As 10 configurações são apresentadas na figura 2-11.

Um circuito para 7-LED display é visto na figura 2-12. Geralmente o decimal é codificado BCD no input, havendo CIS próprios para decodificação. Observe que basta ao driver decodificador baixar o nível de um de seus terminais de saída para que o LED ligado a ele entre em condução; o LED fica apagado quando o respectivo terminal está em nível alto.

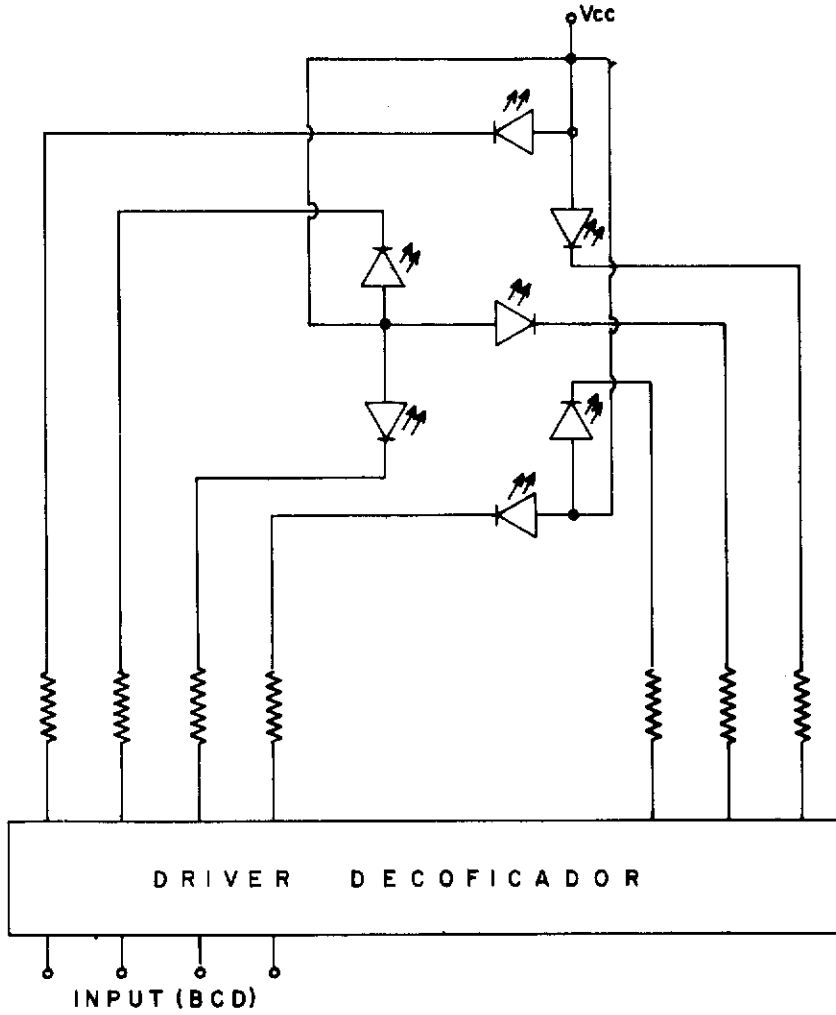


figura 2-12 - Circuito 7-LED display

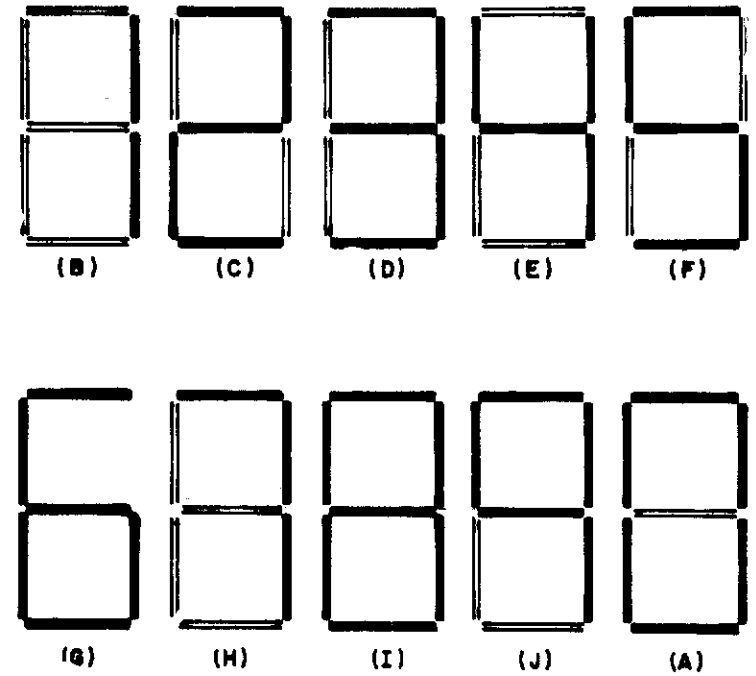


figura 2-11 - Formação de números decimais com o 7-LED

CAPÍTULO 3 – DISPOSITIVOS ELETRÔNICOS

A Eletrônica Digital é implementada (quase que exclusivamente) por semicondutores, destacando-se as tecnologias bipolar e FET que serão estudadas neste capítulo. Amplificadores operacionais não são propriamente dispositivos digitais, mas devido ao extenso emprego nesta área também os incluímos no nosso estudo.

Apresentamos os componentes na sua forma discreta (isto é, separados, fabricados isoladamente) mas em Eletrônica Digital eles aparecerão como parte de circuitos integrados, classificando-se:

- SSI ("small scale integration" - integração em pequena escala), até 100 componentes por CI
- MSI ("medium scale integration" - integração em escala média), de 100 a 1000 componentes por CI
- LSI ("large scale integration" - integração em larga escala) de 1000 a 5000 componentes por CI
- VLSI ("very large scale integration" - integração em escala muito grande) - mais de 5000 componentes por CI.

(o número de componentes por classe é aproximado, não havendo um padrão universalmente aceito).

(1) DIODOS SEMICONDUCTORES

Dividimos a voltagem aplicada ao diodo semicondutor em três regiões de operação (figura 3-1).

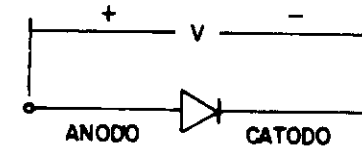


FIG. 3-1A

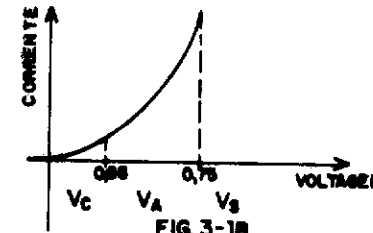


FIG. 3-1B

Figura 3-1 - Voltagem no diodo (A) Circuito (B) Gráfico

- (1) V_s (voltagem de saturação), maior que 0,75 Volts. Acima desse valor ocorre saturação, praticamente desaparecendo a resistência à corrente, como se ele estivesse em curto.
- (2) V_A (voltagem de região ativa), entre 0,65 Volts e 0,75 Volts. Nesta região o diodo comporta-se quase como uma resistência ôhmica, com passagem de corrente proporcional à voltagem aproximadamente linear.
- (3) V_c (voltagem de corte), menor que 0,65 Volts. A qui a corrente que atravessa o diodo é considerada desprezível, como se estivesse "aberto".

Quando o diodo está sob voltagem de saturação ou ativa dizemos que ele está em "polarização direta", sob voltagem de corte diz-se "polarização inversa". Observe que o diodo pode ter tensão posi

va entre anodo e catodo e ainda assim estar inversamente polarizado, basta que a voltagem seja maior que zero e menor que 0,65 Volts.

A passagem de uma região para outra não ocorre instantaneamente, especialmente quando se quer levar a corte um diodo que está saturado (de ON para OFF). O DIODO SCHOTTKY (figura 3-2) é feito exatamente para contornar esse problema, permitindo uma rápida comutação além de ter menor voltagem.

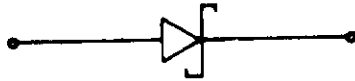


Figura 3-2 - Símbolo do diodo Schottky

(2) TRANSISTOR BIPOLAR

O transistor bipolar pode ser entendido, simplificado, como a junção de dois diodos (figura 3-3). Disso resulta que suas regiões de operação estão relacionadas às voltagens de diodo estudadas acima.



FIG. 3-3B

FIG. 3-3A

Figura 3-3 - Transistor bipolar (A) NPN (B) PNP

No transistor NPN (figuras 3-3A e 3-4A) a corrente da base (i_B) junta-se à do coletor (i_C),

formando a corrente do emissor ($i_B + i_C = i_E$). A seta no símbolo indica o sentido do fluxo da corrente (convencional, de "cargas" positivas). O valor da corrente de base é muito pequeno em relação à corrente do coletor, cerca de 100 vezes menor; considera-se assim que a corrente saindo pelo emissor é aproximadamente a mesma entrando pelo coletor.

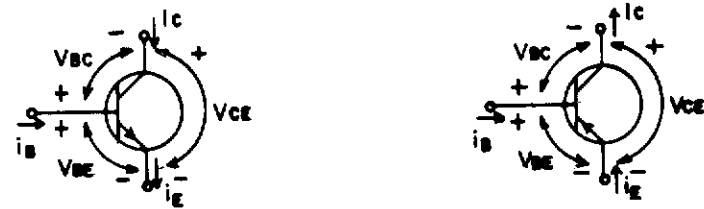


Figura 3-4 - Transistor (A) NPN (B) PNP

No transistor PNP (figuras 3-3B e 3-4B) a corrente entrando no emissor soma-se à da base, formando a corrente que sai no coletor ($i_E + i_B = i_C$).

Como nos diodos o transistor tem três regiões de operação:

- (1) CORTE - os dois "diodos" que formam o transistor estão inversamente polarizados. No NPN, como geralmente o emissor é aterrado (através de um resistor) e o coletor conectado à alimentação dc, basta colocar voltagem menor que 0,65 Volts na base para levar o transistor ao corte. Nesta condição ele se comporta como um circuito aberto, não havendo passagem de corrente do coletor para o emissor. No PNP ocorre o inverso, sendo necessária uma voltagem maior que 0,65 Volts na base para levá-la ao corte.
- (2) REGIÃO ATIVA - o "diodo" base-emissor é polarizado diretamente e o "diodo" base-coletor inversamente. No NPN aplica-se à base uma tensão de 0,65 Volts a 0,75 Volts em relação ao emissor. No PNP é colocada na base tensão negativa - 0,65 Volts a

- 0,75 Volts, em relação ao emissor.

- (3) SATURAÇÃO - os dois diodos são polarizados diretamente, no NPN com uma tensão base-emissor maior que 0,75 Volts e no PNP menor que - 0,75 Volts (menor no caso seria - 0,76 V, - 0,77 V, - 0,78V, ... porque é valor negativo). Quando saturado o transistor bipolar apresenta uma tensão de cerca de 0,2 Volts entre coletor e emissor, e essa é uma propriedade fundamental.

A região ativa é pouco utilizada em Eletrônica Digital, interessando mais os dois extremos - saturação ou corte. A figura 3-5A mostra um esquema típico para geração de sinais digitais. Com a chave da base em 1 Volt (nível 1,H) o transistor entra em saturação, apresentando 0,2 Volts (nível 0,L) na saída. Aterrando a base (nível 0,L) ocorre o corte e os 5 Volts da alimentação são integralmente apresentados à saída através do resistor, já que não há passagem de corrente provocando queda de tensão. A figura 3-5B mostra esses valores. Como se vê, este circuito é uma porta NOT.

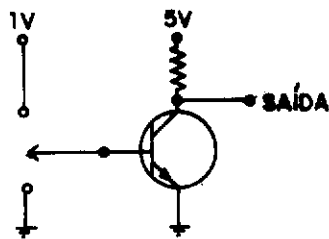


Figura 3-5A - Geração de níveis lógicos H e L com transistores.

BASE	SAÍDA
1 Volt	0,2 Volt (0,L)
0 Volt	5 Volts (1,H)

Figura 3-5B - Níveis de voltagem do Circuito

É comum o uso de transistor bipolar operando em MODO INVERSO (figura 3-6). Invertendo-se a polarização do coletor e emissor inverte-se o sentido da corrente, que é constituída principalmente pela corrente i_B e muito pouco pela corrente de emissor i_E , ao contrário da operação normal. A grande vantagem do modo inverso é a pequena voltagem de saturação V_{CE} entre coletor e emissor, da ordem de 0,5 mV.

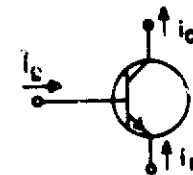


Figura 3-6 - Transistor no modo inverso

Um dado importante em tecnologia de transistores é o tempo de comutação. Em documentação dos fabricantes usa-se a seguinte terminologia (figura 3-7)

- * t_d - time delay (tempo de atraso): tempo para transistor atingir 10% da corrente de saturação
- * t_r - time rise (tempo de ascensão): tempo para sair

de 10% a 90% da corrente de saturação.

* t_s - time storage (tempo de armazenamento): tempo a partir da mudança de tensão na base, com o transistor totalmente saturado, até atingir 90% da corrente de saturação.

* t_f - time fall (tempo de queda): tempo para sair de 90% a 10% da corrente de saturação.

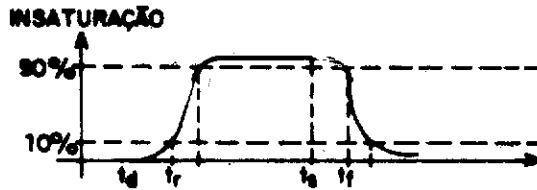


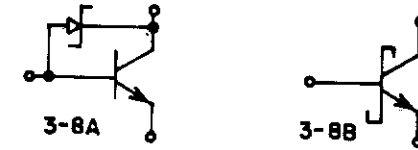
Figura 3-7 - Tempos de comutação no transistor.

O transistor bipolar é bastante rápido em comutação de estados, porém grande consumidor de energia. Em casos onde a velocidade é mais importante que o consumo é dada preferência a essa tecnologia de semicondutores.

A maior demora na comutação resulta da dificuldade de tirar o transistor do estado de saturação. Uma solução para aumentar a velocidade consiste em colocar um diodo Schottky entre a base e o coletor do transistor (figura 3-8A). Como o diodo satura com 0,3 Volts, esta passa a ser a voltagem máxima entre base e coletor do transistor, não permitindo que ele sature. O dispositivo fabricado assim é denominado TRANSISTOR SCHOTTKY e simbolizado pela figura 3-8B.

Para que o transistor em corte apresente nível 1 entre seu coletor e o resistor que o liga à alimentação (figura 3-9A) é preciso que a corrente i_L desviada preencha a capacitância da carga C_L até a voltagem ali atingir o nível alto. Chamamos esse

processo de PULL UP, e obviamente ele gasta certo tempo para se realizar.



Figuras 3-8 - Transistor Schottky (A) Construção (B) símbolo

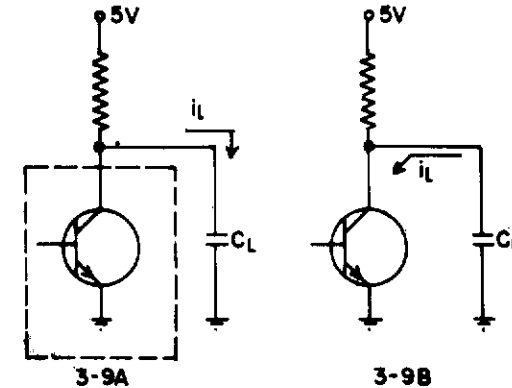


Figura 3-9A- pull-up

Figura 3-9B - pull-down

Inversamente, decorre um espaço de tempo para que o transistor saindo do corte e entrando em saturação absorva a descarga da capacitância da carga, baixando a tensão em seu coletor a 0,2V (figura 3-9B) - o que é denominado PULL DOWN.

Quando a corrente i_L para carga ou descarga da capacitância C_L deve atravessar somente componentes resistivos (de resistência R) caracteriza-se um PULL UP/DOWN PASSIVO, cujo tempo é determinado pela constante RC_L . Há vários circuitos que substituem a resistência R por outros componentes, formando um PULL UP/DOWN ATIVO, mais rápido e eficiente. Na figura 3-9 temos em (A) um pull up passivo e em (B) um pull down ativo.

(3) JFET - TRANSISTOR FET DE JUNÇÃO

Ao estudar transistores bipolares dissemos que eles poderiam ser considerados como a junção de dois diodos. O que temos na verdade são duas regiões - P e N - onde os portadores majoritários de cada uma migram para a outra (figura 3-10), onde se constituirão em cargas livres. Entre elas se forma uma ZONA DE DEPLEÇÃO, desprovida de cargas livres e usada como base. Ao polarizar diretamente este semicondutor (tensão positiva na região P e negativa na N) resulta uma corrente tanto de elétrons quanto de lacunas através da base, originando-se aí o termo "bipolar".

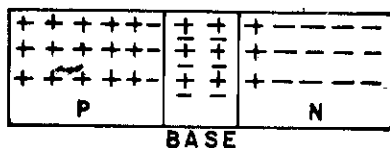


Figura 3-10 - Junção PN

No transistor JFET ("junction field-effect transistor" - transistor de efeito de campo de junção), também conhecido por FET DE JUNÇÃO, uma

região N é intercalada entre duas P (ou vice-versa), constituindo um CANAL N (figura 3-11). Polarizando-se os extremos do canal ocorre uma passagem de elétrons do polo negativo - SOURCE (fonte) - para o positivo - DRAIN (dreno), através do canal (se o canal for P haverá um fluxo de lacunas do polo positivo - que passa a ser source - para o negativo - no caso dreno. Ou seja, o tipo de canal determina qual portador se move e qual polaridade é source ou dreno). Como um só tipo de portador se move, ao contrário do transistor bipolar, dizemos que o JFET é "unipolar".

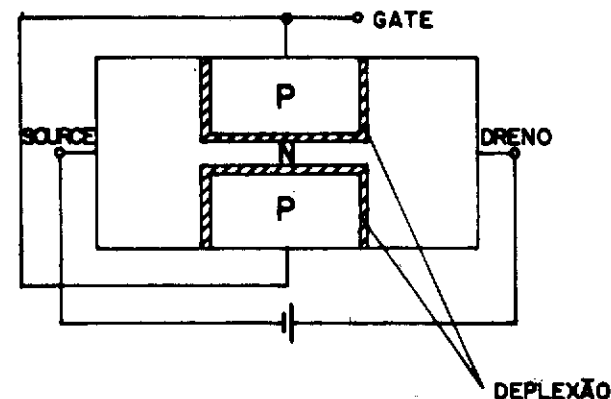


Figura 3-11 - FET de junção

O tamanho da região de depleção depende da tensão a que está submetida, aumentando com polarização inversa (tensão negativa em P e positiva em N). Se no gate do JFET canal N for aplicada voltagem negativa suficiente (as duas regiões de depleção se alargarão, unindo-se e fechando o canal (figura 3-12) efetivamente levando o transistor ao corte - não há passagem de corrente (no JFET canal P aplica-se ten-

são positiva ao gate para cortá-lo).

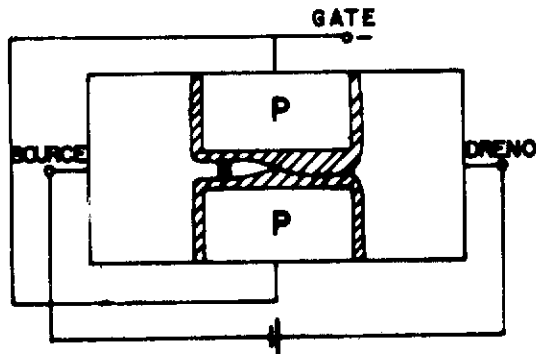


Figura 3-12 - JFET em corte

É interessante ressaltar que o JFET está "normalmente" conduzindo (ativo) e é preciso um sinal no seu gate para levá-lo ao corte. Esse processo é chamado MODO DE DEPLEÇÃO.

Na figura 3-13 vemos o símbolo do JFET sendo (A) canal N e (B) canal P (note que as setas tem sentido oposto). Lembre-se, também, que dreno e source não são fixos, bastando mudar a polaridade para alterná-los.

Dentro de certos limites de tensão o JFET tem características bem lineares, apresentando resistência ôhmica à passagem de corrente pelo canal proporcional à voltagem no gate. Isso o torna uma espécie de resistor controlado por voltagem.

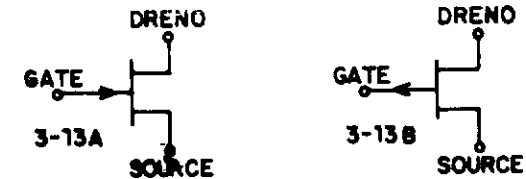


Figura 3-13 - Símbolo do JFET (A) Canal N (B) Canal P

(4) MOSFET

O MOSFET ("metal-oxide semiconductor FET" - FET de semiconductor metal-óxido) é um dos mais usados tipos de transistor em Eletrônica Digital, devido principalmente à sua grande densidade de integração (grande quantidade num pequeno CI).

O MOSFET tipo N - NMOS - consiste de um substrato P no qual se inscrevem duas regiões N, com uma camada metálica onde sobressai o gate unindo os três (no MOSFET tipo P - PMOS - temos substrato N e duas regiões P), conforme figura 3-14.

Com o gate a 0 Volts o transistor permanece cortado, pois há uma descontinuidade entre as duas regiões N. Com tensão positiva no gate atrai-se elétrons do substrato, preenchendo a área descontinua com outra região N (figura 3-15), o que permite a passagem de corrente da source para o dreno. Assim, tensão positiva no gate do NMOS coloca-o em condução (igualmente para tensão negativa no gate do PMOS).

Observe-se que, ao contrário do JFET, o MOSFET "normalmente" está cortado e precisa de sinal externo para entrar condução.

O esquema apresentado a seguir refere-se a MOSFET DE INCREMENTO, cujo símbolo é visto na

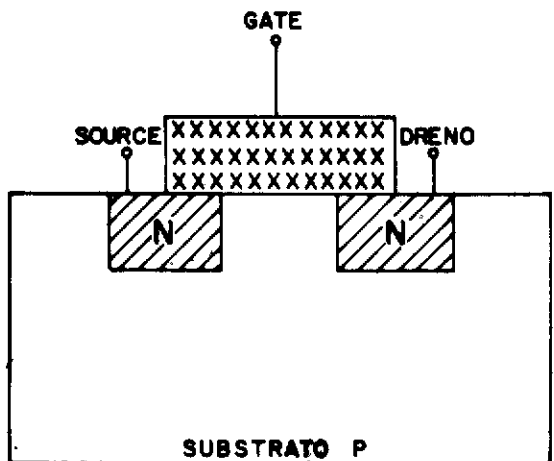


Figura 3-14 - MOSFET

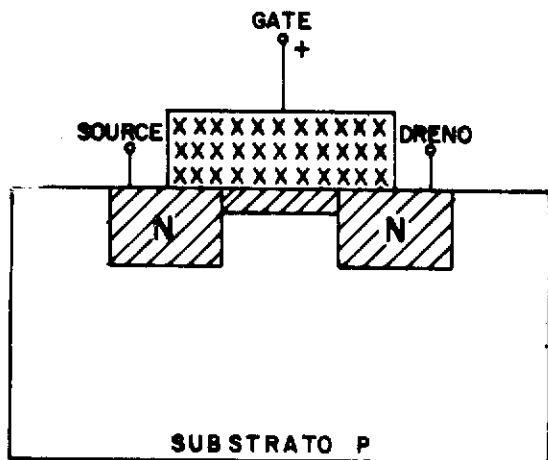


Figura 3-15 - MOSFET em condução

figura 3-16 para (A) NMOS e (B) PMOS. A nova região formada para condução é oposta à do substrato (N e P ou P e N).

Como vimos anteriormente aparecerá uma zona de depleção entre elas, cujo tamanho dependerá da tensão no substrato e determinará a condutância no canal. Por esse motivo o substrato pode ser considerado um segundo gate e é feito um eletrodo para sua ligação.

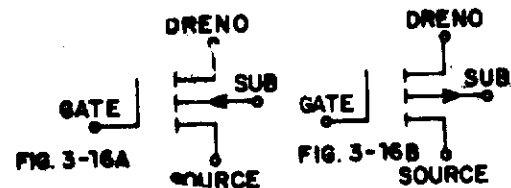


Figura 3-16

Símbolo do MOSFET (A) NMOS (B) PMOS

O MOSFET DE DEPLEÇÃO é fabricado com a região de descontinuidade já preenchida, de modo que sem aplicação de sinal externo há condução. A região de depleção entre substrato e esta área pode ser alargada por aplicação de polarização inversa no gate, tal como no JFET. O símbolo do MOSFET de depleção é vis-

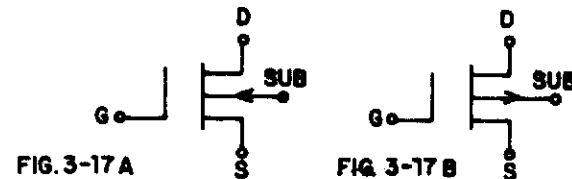


Figura 3-17

Símbolo do MOSFET de depleção (A) NMOS (B) PMOS

ro na figura 3-17, sendo (A) canal N e (B) canal P. A única diferença do símbolo de MOSFET de incremento está nos traços ligando substrato a dreno e source.

Uma simplificação do símbolo de MOSFET é apresentada na figura 3-18 - (A) NMOS e (B) PMOS, sendo bastante usada. Não é feita distinção entre modos de depleção ou incremento e sobre substrato.

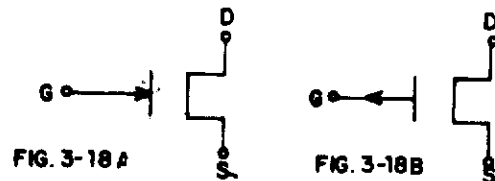


Figura 3-18
Símbolo simplificado do MOSFET (A) NMOS (B) PMOS

Uma das configurações mais comuns de MOSFET, especialmente na fabricação de microprocessadores, é mostrada na figura 3-19. Temos dois transistores de depleção, T2 denominando-se "drive" (dirigente) e T1 "load" (carga). Com o sinal de entrada (input) em nível 1 o drive T2 conduz, praticamente aterrando o output (nível 0), o que por sua vez corta o load T1. Com input = 0 o drive entra em corte, não havendo sinal no output e no gate do load T1, provocando sua condução (FET de depleção conduz sem sinal externo), e daí a output = 1. Veja que a fonte de alimentação nunca está conectada ao terra, tornando o consumo de energia muito baixo.

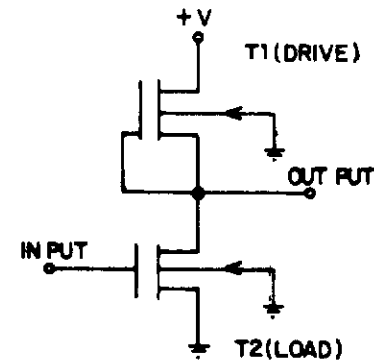


figura 3-19 - Chaveamento com dois FETs

(5) CMOS

O CMOS ("complementary MOS" - MOS complementar) conjuga um NMOS com um PMOS (figura 3-20). Com input = 1 T1 conduz, aterrando output (nível 0), ao mesmo tempo que T2 é cortado. Fazendo input = 0 T1 é cortado e T2 conduz, conectando a fonte de alimentação ao output (nível 1).

No CMOS a impedância entre gate - source chega à casa de 1 trilhão de Ohms. Por isso uma pequena corrente no input é suficiente para provocar enorme voltagem, maior que o limite máximo de aproximadamente 150 Volts, suportado pelo componente, danificando-o. Recomendações para manipular CMOS, além de extremo cuidado, é não tocar seus terminais com a mão e o instrumento (a carga estática do material é por si

são maior que a pequena corrente suportável) e soldá-lo com o ferro desligado da tomada.

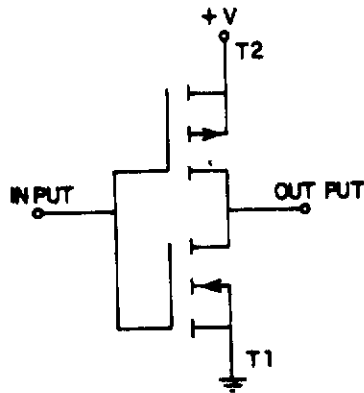


figura 3-20 - CMOS

Os CIs CMOS são disponíveis comercialmente sob a numeração 40XX ou 4502). A série CMOS da MOTOROLA, muito usada e recomendada, começa com '1' (nos exemplos acima: 14049, 14502). Note que CIs com a mesma numeração são equivalentes, independente do fabricante.

Grande esforço vem sendo desenvolvido para tornar esta tecnologia mais resistente e manipulável. Disso resultou uma classe especial de CMOS com circuito de proteção, a cuja numeração sempre é posposta à letra 'B'; ao CMOS comum é posposto 'A' ou nada é escrito (nos exemplos acima: 4049B - com proteção, 4049A ou 4049 - sem proteção; 4502B - com proteção, 4502A ou 4502 - sem proteção).

(6) AMPLIFICADORES OPERACIONAIS

(6) AMPLIFICADORES OPERACIONAIS

Amplificadores operacionais são circuitos (e não componentes) com características de alto ganho da diferença entre os sinais na entrada (figura 3-21). Um dos inputs é amplificado invertendo-se seu sinal, INPUT COM INVERSÃO (-), e o outro sem inversão, INPUT NÃO INVERSOR (+).

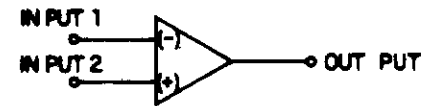


figura 3-21 - Amplificador operacional

Aplicando-se dois sinais V_1 e V_2 na entrada resulta na saída V_{output} (figura 3-22). $V_{output} = A(V_2 - V_1)$, onde A é o ganho do amplificador da ordem de 50.000. Veja que se a diferença entre V_2 e V_1 for 1 mVolt teremos no output 50 Volts. Também, se V_2 (inversor) for mais que V_1 (não inversor) o output terá o sinal inverso de V_2 .

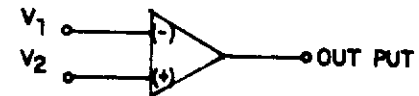


figura 3-22 - Amplificação da diferença de voltagens nos inputs.

Idealmente apenas a diferença entre as duas entradas seria amplificada. Na prática ocorre

uma pequena amplificação do próprio sinal, o que é in desejável. Chamamos CMRR ("common-mode rejection ratio", relação de rejeição modo-comum) um padrão que relaciona a amplificação da diferença com os sinais. Quanto maior o CMRR melhor o amplificador operacional, estando na casa de 100 decibéis.

Aterrando o não inversor, conectando o sinal V_{in} a ser amplificado através de um resistor R e intercalando entre a saída e o não inversor outro resistor de "feedback" (retroalimentação) R_F (figura 3-23) obtem-se um ganho e output.

$$A_F = \frac{-R_F}{R} \quad V_{output} = -\frac{R_F}{R} \times V_{in}$$

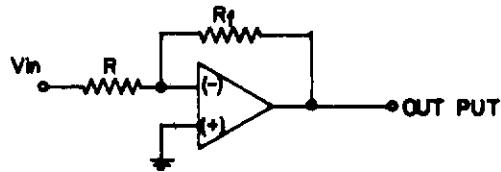


figura 3-23 - Amplificador operacional com feedback

A configuração de amplificadores operacionais do exemplo anterior é extremamente útil para calibração de instrumentos, fazendo-se R_F como potenciômetro, substituindo V_{in} por uma V_{in} de referência confiável. V_{in} ajusta-se R_F até o output atingir o valor de calibração V_{in} teoricamente esperado (figura 3-24A). Em outra aplicação o resistor de entrada R é selecionado por chaveamento, sendo um submúltiplo do resistor de feedback; V_{in} será multiplicado por um fator que dependerá da V_{in} chave. Por exemplo comutando-a para $R/2$:

$$V_{output} = -\frac{R}{R/2} V_{in} = -2 V_{in}$$

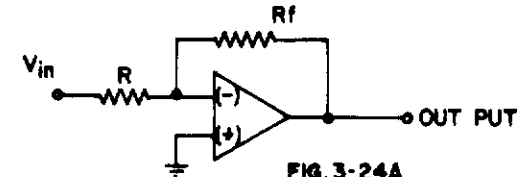


FIG. 3-24A

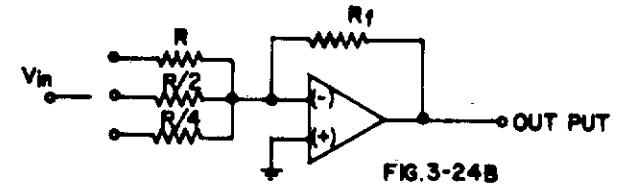


FIG. 3-24B

figura 3-24 - Aplicações do amplificador com feedback (A) calibração (B) scaling

Esta configuração ("scaling") é usada para calculadoras e computadores analógicos.

Com a entrada V_{in} no não inversor (figura 3-25A) a fórmula da saída V_{in} torna-se

$$V_{output} = + \left(1 + \frac{R_F}{R} \right) V_{in}$$

dando aproximadamente os mesmos valores da anterior. Num caso especial retira-se R_F e R (figura 3-25B) e forma-se um "amplificador de V_{in} ganho unitário", ou seja, a mesma voltagem da entrada aparece na saída. Sua utilidade é como buffer (isolação entre dois estágios), tendo cerca de 200 M Ohms de impedância na entrada e apenas 0,02 Ohms na saída.

COMPARADOR é um amplificador operacional

onde a saída pode assumir só dois níveis de tensão (V_2 e V_1), dependendo da tensão de entrada V_{in} ser maior ou menor que certa tensão de referência V_{REF} . Ou seja,

$$V_{output} = V_1 \text{ se } V_{in} \text{ menor que } V_{REF}$$

$$V_{output} = V_2 \text{ se } V_{in} \text{ maior que } V_{REF}$$

Geralmente V_1 e V_2 são os conhecidos níveis lógicos 1 e 0 (ou H e L).

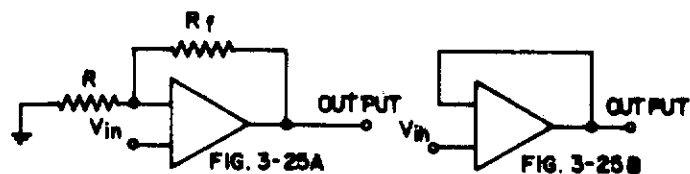


figura 3-25 - Amplificação não-inversor (A) com feedback (B) ganho unitário.

SCHMITT TRIGGER ("gatilhador de Schmitt"), é um tipo especial de comparador (figura 3-26) que altera abruptamente os níveis de saída (V_1 e V_2) conforme alteração na tensão de entrada V_{in} .

É comum representar o Schmitt Trigger pelo símbolo de amplificador operacional com um 'S' duplo no seu interior. Essa representação, entretanto, não é obrigatória.

(7) INTERFACES (BUFFER E LATCH)

Na realização prática do circuito podem parecer incompatibilidades na junção de estágios, sendo necessário intercalar dispositivos de INTERFACE pa

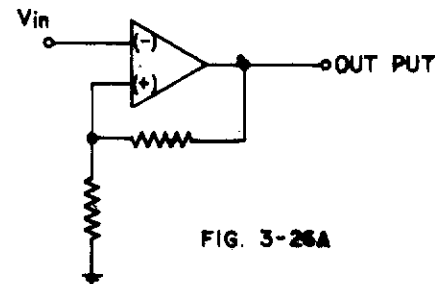


FIG. 3-26A



FIG. 3-26B

figura 3-26 - Schmitt Trigger(A)Circuito (B)Símbolo

ra torná-los compatíveis.

Quando um estágio não tem na sua saída corrente suficiente para dirigir os dispositivos que dependem dele na sequência (o que ocorre frequentemente quando um CI deve comandar vários outros simultaneamente) intercala-se um BUFFER (figura 3-27), que dispõe de potência adequada no output. Um circuito tristate também pode ser acrescentado ao buffer, de maneira que sem o sinal de habilitação EO ("enable output" - habilita output) sua saída permanece em tristate, isolada eletricamente do input. O símbolo do buffer é apresentado na figura 3-28.

COLETOR ABERTO ("open collector") é uma tecnologia especial onde o dispositivo é capaz de receber corrente, caracterizando nível 0, mas mostra-se

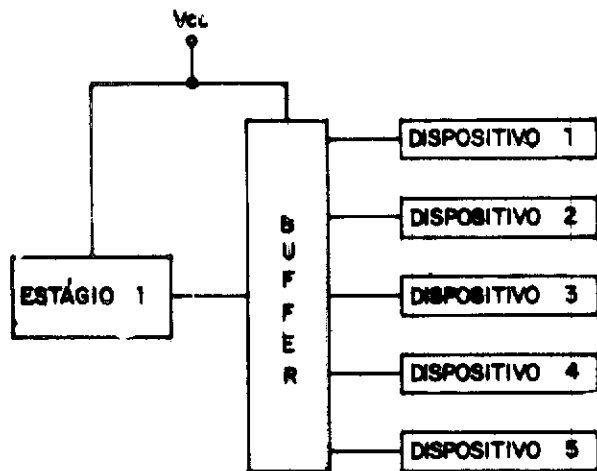


figura 3-27 - Circuito com buffer

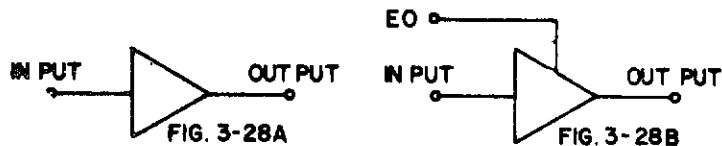


figura 3-28 - Símbolo do buffer (A) simples (B) com tristate

em tristate em nível 1 quando deveria funcionar como fonte de corrente.

Na figura 3-29 temos um circuito típico usando o buffer inversor coletor aberto 7406. Quando o sinal IN está em nível 1 o inversor apresentará ní

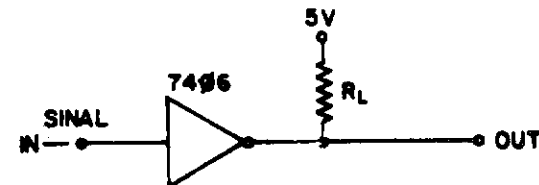


figura 3-29 - Circuito com buffer coletor aberto

vel \emptyset , recebendo corrente e provocando uma queda de tensão no resistor de carga R_L , caracterizando nível \emptyset na saída. Quando sinal IN é levado a \emptyset o buffer deveria mostrar nível 1 na saída, fornecendo corrente. Entretanto ele permanece em tristate, desligado do sistema. No nosso exemplo simples a saída OUT será levada a nível 1, por falta de queda de tensão em R_L . Mas num circuito mais sofisticado outro dispositivo (geralmente outro coletor aberto) assume o controle lógico da saída, determinando seu nível.

No LATCH os dados existentes no input no momento em que é habilitado o terminal DS ("device select", seleção de dispositivo) são armazenados no output por tempo indeterminado. Na figura 3-30 temos um latch com capacidade de 4 bits. Este esquema é muito útil na multiplexação (figura 3-31): a 1ª remessa é armazenada no latch mas não dá entrada no estágio 2 pois DS2 é mantido desabilitado; em seguida DS1 é desabilitada e DS2 habilitada, emitindo-se a 2ª remessa de bits; agora as duas remessas aparecem simultaneamente como entradas no estágio 2.

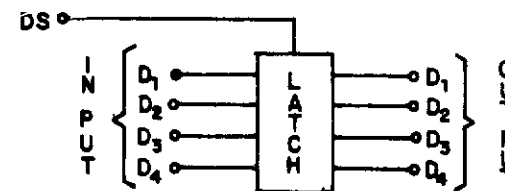


figura 3-30 - Latch

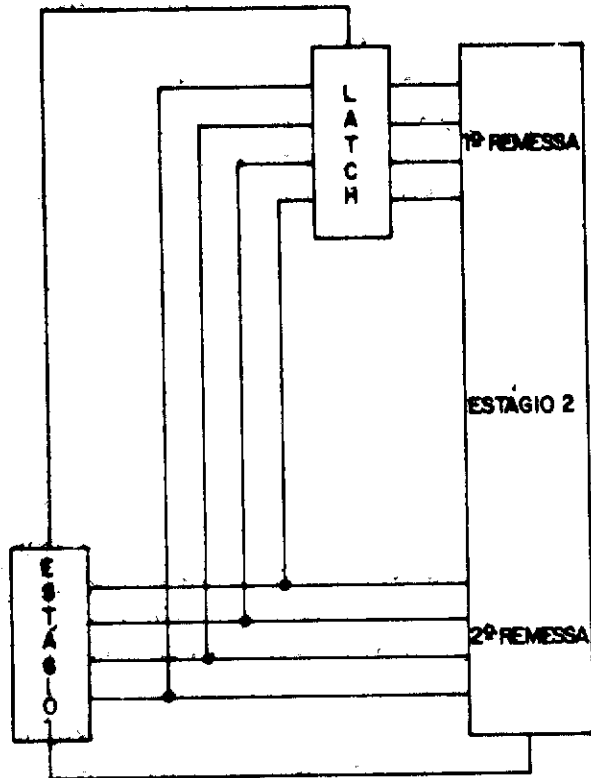


figura 3-31 - Multiplexação usando latch

CAPÍTULO 4 – FAMÍLIAS LÓGICAS

Portas lógicas, amplificadores operacionais, microcomputadores e outros dispositivos empregados em Eletrônica Digital são fabricados segundo algumas tecnologias padronizadas - denominadas "famílias lógicas".

Neste capítulo faremos um breve estudo sobre as famílias RTL, IIL, ECL e TTL. Este último, devido à sua grande importância, será visto mais detalhadamente, inclusive em seu interfaceamento com a tecnologia CMOS.

O leitor deve ter em mente que as configurações apresentadas aqui como componentes discretos são implementadas quase que exclusivamente na forma de circuitos integrados.

(1) RTL

RTL ("resistor transistor logic" - lógica resistor transistor) é a primeira família lógica e hoje não é mais usada. Entretanto, seus princípios de funcionamento são básicos para as outras famílias.

O modelo de RTL é visto na figura 4-1. Se um ou mais inputs (V_1, V_2, V_3) estiver em nível 1 resultará em saturação do respectivo transistor, levando seu coletor e o output a 0,2 Volts (nível 0). Com todos inputs em nível 0 nenhum dos transistores conduzirá e a tensão de alimentação V_{CC} aparecerá integralmente no output através do resistor R_C (nível 1). Trata-se, portanto, de uma porta NOR.

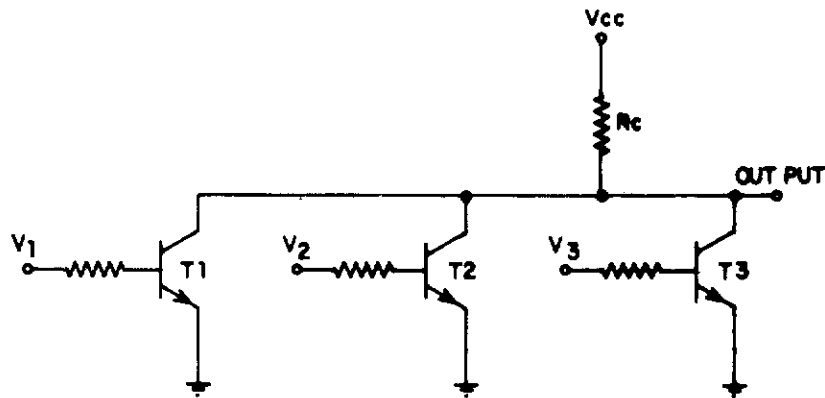


figura 4-1 - RTL

O número máximo de portas lógicas que podem ser conectadas ("fan-out") ao RTL é aumentado e o tempo de comutação diminuído pela substituição do pull-up passivo R pelo pull-up ativo da figura 4-2. Com o input V_{in} em nível 1 teremos T3 saturado e o output em nível \emptyset ($\emptyset,2$ Volts); ao mesmo tempo T1 também estará saturado, mantendo T2 cortado (nível \emptyset na base). Levando V_{in} a \emptyset resulta T3 e T1 cortados e T2 conduzindo, com a tensão no output ligeiramente inferior a alimentação (diferença de tensão entre V_{cc} e output: queda de tensão em $R_c + \emptyset,2$ Volts do coletor ao emissor de T2), caracterizando nível 1.

DCTL ("direct coupled transistor logic" - lógica de transistor acoplado diretamente) é um caso particular de RTL onde são retirados os resistores ligados à base, acoplando-se os inputs diretamente.

FIG.4-1 RTL

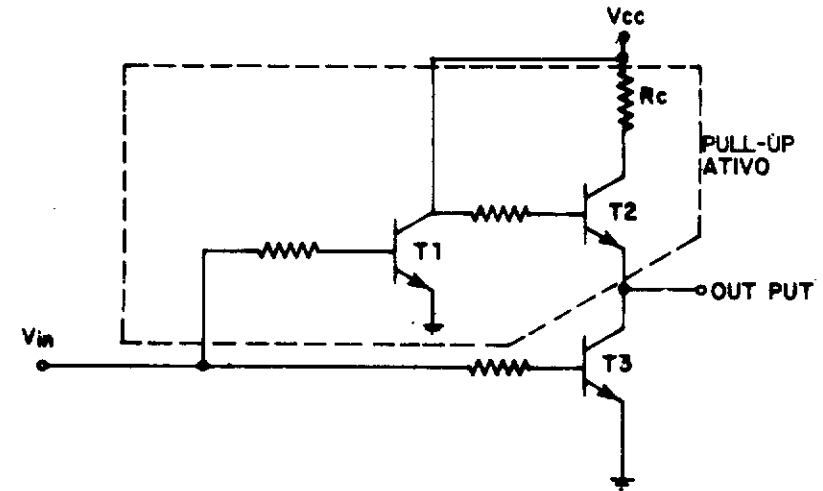
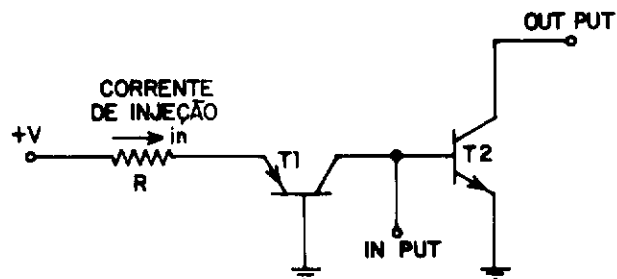


figura 4-2 - RTL com pull-up ativo

(2) IIL (ou I^2L)

IIL ou I^2L ("integrated injection logic" - lógica de injeção integrada) é uma tecnologia que permite ao usuário escolher o melhor ponto entre consumo e velocidade: aumentando um aumenta também a outra, e inversamente. Essa característica a torna adequada para sistemas alimentados por bateria, tal como calculadoras (a Texas Instruments domina e emprega largamente a IIL).

O valor do resistor R é escolhido pelo usuário e determina a intensidade da corrente de injeção que vai do emissor de T1 até a base de T2 (figura 4-3). Com o input em nível \emptyset essa corrente é desviada, deixando T2 em corte e output = 1. Com input = 1 a corrente atravessa a base de T2, saturando-o e fazendo output = \emptyset .

figura 4-3 - I^2L

(3) DTL

DTL ("diode - transistor logic" - lógica diodo-transistor) é uma tecnologia que permite grande fan-out mas apresenta baixa velocidade de operação.

O inputs (V_1, V_2, V_3) são conectados à base do transistor através de diodos (figura 4-4). Se qualquer um deles é posto em nível 0 resulta baixa voltagem na extremidade inferior de R_1 , que leva o transistor ao corte e output a nível 1 (mesma voltagem que $+V_{CC}$, pois não há corrente e respectiva queda de tensão em R_2). Com todos inputs em nível 1 seus diodos ficam em corte, permitindo à corrente que sai de $+V_{CC}$ chegar a base do transistor, saturando-o e fazendo output = 0.

HTL ("high threshold logic" - lógica de alta margem) é um tipo de DTL onde o diodo D_1 é substituído por um Zener, resultando uma tecnologia de grande tolerância a ruídos.

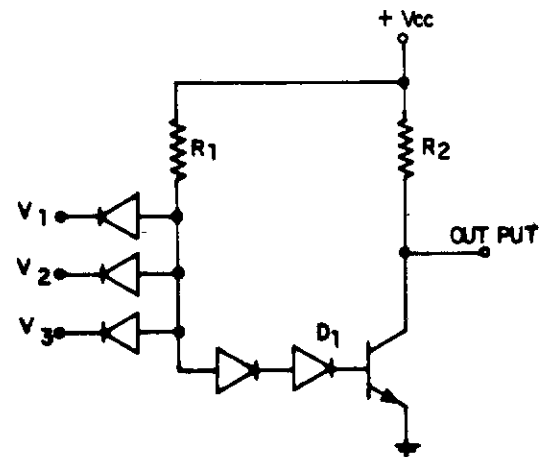


figura 4-4 - DTL

(4) ECL

ECL ("emitter coupled logic" - lógica de acoplamento de emissores) é a mais veloz das tecnologias, sendo usada principalmente em computadores de grande porte e CIs de comutação rápida.

A sua configuração básica (figura 4-5A) é semelhante a um comparador, onde uma corrente i constante é originada de outras duas variáveis (i_1 e i_2). Quando i_1 aumenta i_2 diminui e vice-versa, sendo todo o sistema projetado com valores tais que T_1 e T_2 não saturem. Com input = 1 temos i_1 muito grande e i_2 pequena, com pouca queda de tensão em R_2 , o que nos dá output = 1. Se input = 0 então i_1 é pequena e i_2 grande, ocorrendo grande queda de tensão em R_2 e consequentemente output = 0.

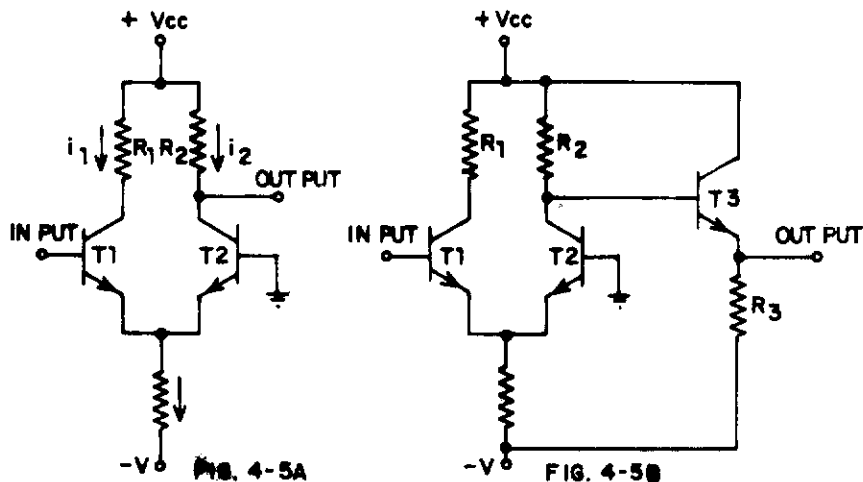


figura 4-5 - ECL (A) configuração básica
(B) com deslocador de voltagem

Para adaptar o output a níveis adequados geralmente adapta-se um deslocador de voltagem (transistor T 3, figura 4-5B), que recebendo a corrente de T 2 amplifica-a no emissor, produzindo ali um output proporcional ao valor de R 3.

Devido à sua configuração de corrente constante i as tensões de polarização entre base - e emissor do transistor de input são diferentes das usuais, valendo para V_{BE} :

$$V_{corte} = 0,70 \text{ Volts}$$

$$V_{ativa} = \text{de } 0,70 \text{ Volts a } 0,80 \text{ Volts}$$

$$V_{saturação} = 0,80 \text{ Volts}$$

(5) TTL

TTL ("transistor-transistor logic" - lógica transistor - transistor) é a mais importante família lógica em Eletrônica Digital, já caracterizada como padrão. Numera-se os CIs desta tecnologia por 74xx onde xx são dois números indicando os tipos de portas integradas.

A configuração básica TTL consta na figura 4-6. A alimentação é padronizada em 5 Volts (regulados).

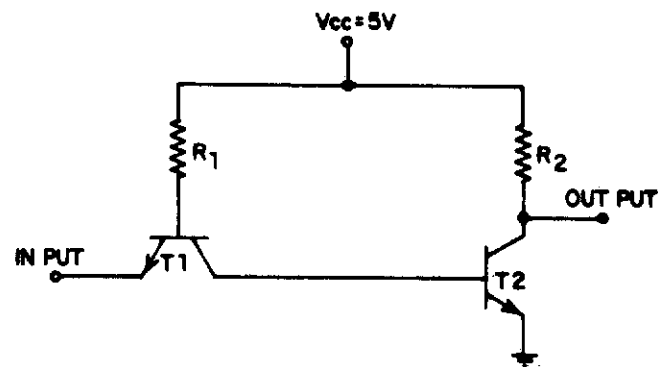


figura 4-6 - Configuração básica do TTL

Com input = 1 T 1 conduz em modo inverso recebendo as correntes do emissor (input) e base (V_{cc}) e lançando-as na base de T 2, saturando-o, o que leva a output = 0 (figura 4-7A). É importante notar que a corrente de input = 1, i_{i1} , entra no TTL, o mesmo ocorrendo com a corrente de output = 0, i_{o1} .

Com input = 0 T 1 se comporta como um diodo

do, recebendo a corrente da alimentação i_A e encaminhando-a para o emissor; seu coletor não tem de onde tirar corrente, mantendo-se cortado e cortando T 2, acarretando output = 1 (figura 4-7B). Agora tanto a corrente de input = \emptyset i_{i1} , quando a corrente de output = 1, i_{o1} , saem do TTL.

Diversos aperfeiçoamentos são introduzidos na configuração básica para dar melhor performance ao TTL, conservando entretanto os mesmos princípios de funcionamento (figura 4-8). Os transistores são fabricados tipo Schottky, para evitar saturação. O transistor de entrada, T 1, é multi-emissor (vários eletrodos de emissor, base e coletor comuns, como se fossem vários transistores com emissores separados e os outros terminais compartilhados), o que permite maior número de inputs.

T 4 é acrescentado como inversor de fase, e T 2 um pull-up ativo. Assim, se qualquer dos inputs for \emptyset ocorrerá T 4 e T 3 cortados, com output = 1. Se todos inputs são 1 T 4 e T 3 estão na região ativa e output = \emptyset . O exemplo é, portanto, uma porta NAND.

Na verdade temos três tipos de TTL:

- 74xx, modelo standard (exemplos: 7405, 74186)
- 74Sxx, "speed" - alta velocidade de operação (exemplos: 74S05, 74S186)
- 74LSxx, "low-Schottky" - Schottky de baixa potência (exemplos: 74LS05, 74LS186).

Cada tipo tem seus padrões de corrente, como registrado na tabela abaixo, mas todos com

- nível 1 : maior que 2,4 Volts.
- nível \emptyset : menor que 0,4 Volts.

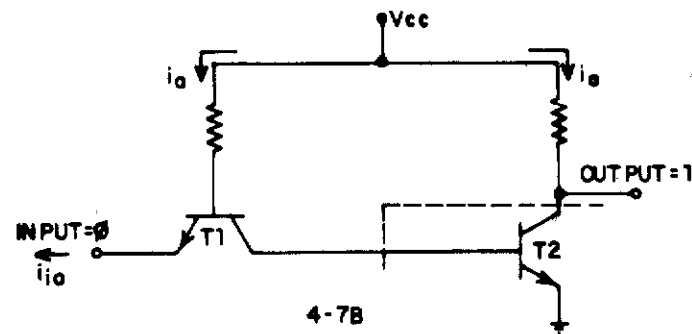
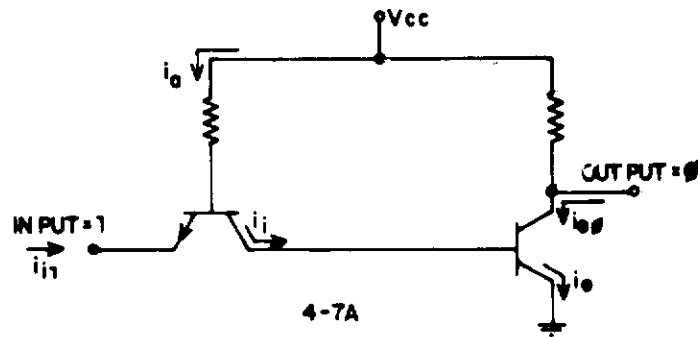


figura 4-7 - Chaveamento com TTL (A) INPUT = 1
(B) INPUT = \emptyset

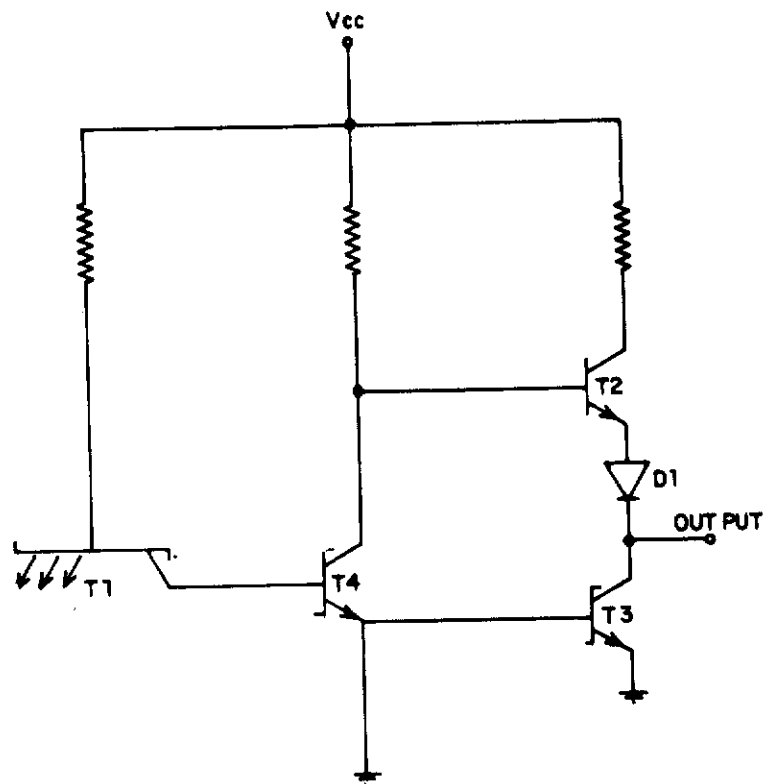


figura 4-8 - Configuração aperfeiçoada do TTL

Nível	Input	74xx	74Sxx	74LSxx
1	Corrente Máxima (Entrando)	40 μ A	50 μ A	20 μ A
0	Corrente Máxima (Saindo)	1,6mA	2,0mA	0,36mA
	Output			
1	Corrente Mínima (Saindo)	400 μ A	1000 μ A	400 μ A
0	Corrente Mínima (Entrando)	16 mA	20 mA	8 mA

A tabela acima é indispensável no interfaceamento entre CIs TTL e outros dispositivos. Por exemplo, se o outro dispositivo precisa receber 1000 μ A como input = 1 então podemos interfacear quatro componentes dele com o TTL 74xx (figura 4-9A). Entretanto, se como input = 0 ele fornece 40mA será possível interfacear apenas 2 dele com o TTL 74xx (figura 4-9B). Como a ligação deve ser adequada para os dois níveis de input, concluímos que apenas dois desses dispositivos podem ser interfaceados com o TTL 74xx.

(6) INTERFACE TTL - CMOS

TTL e CMOS são as mais importantes tecnologias de dispositivos lógicos, e interfaceamento entre as duas aparece com frequência

No primeiro caso - TTL dirigindo CMOS (figura 4-10 A) - notamos que a voltagem nível 1 de output do TTL (2,4 Volts) é insuficiente para excitar em nível 1 o input do CMOS (3,5 Volts).

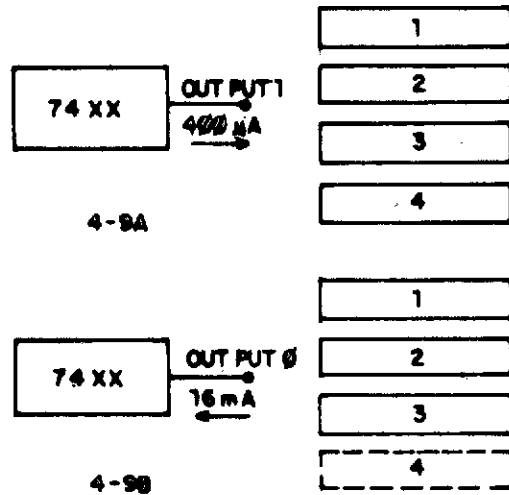


figura 4-9 - Interfaceamento do TTL standard
(A) output 1 (B) ou put \emptyset

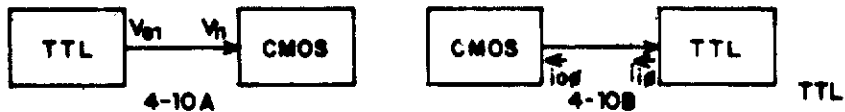


figura 4-10 - Interfaceamento TTL x CMOS
(A) TTL dirigindo CMOS
(B) CMOS dirigindo TTL

Segundo - CMOS dirigindo TTL (figura 4-10B) - a corrente máxima que o CMOS pode absorver com seu output = \emptyset ($\emptyset,4$ mA) é muito menor que a corrente fornecida pelo TTL com seu input = \emptyset (1,6 mA).

A solução para os problemas acima é intercalarmos um CI de interface (buffer) entre TTL e CMOS, o qual assentará a voltagem e corrente nos níveis necessários.

CAPÍTULO 5 – FLIP-FLOP E MULTIVIBRADORES

FLIP-FLOP é possivelmente o dispositivo mais importante da Eletrônica Digital, entrando como componente de memórias, microprocessadores, contadores, registros e outros. Consiste num multivibrador biestável, isto é, tem dois níveis de saída (ou melhor dizendo duas configurações) e permanece estável em qualquer um deles até que um sinal externo altere seu estado.

O MULTIVIBRADOR opera nos mesmos princípios do flip-flop mas não permanece indefinidamente estável nos dois estados possíveis. Se num estado ele é estável e noutro instável dizemos que é MONOESTÁVEL (ou ONE-SHOT); se instável nos dois estados é chamado AESTÁVEL.

Neste capítulo estudaremos os tipos de flip-flop existentes, analisando os detalhes de funcionamento de cada um e sua implementação prática em CIs. Serão vistos ainda modelos de multivibradores monoestável e aestável.

(1) CONFIGURAÇÃO BÁSICA E ESTADO DO FF

A configuração mais simples de FF é apresentada na figura 5-1, consistindo de duas portas NOR (poderia ser NAND), dois inputs - S e R - e dois outputs - Q e \bar{Q} .

O output superior é tomado como referência para questões de terminologia (e muitas vezes é o único usado, deixando-se \bar{Q} flutuando). Quando $Q = 0$

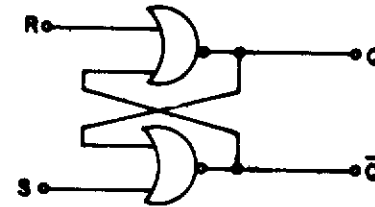


figura 5-1 - Configuração básica do flip-flop

dizemos que o FF está em "reset" - resetado - e $Q = 1$ é dito em "set" - setado.

Usando-se os inputs com níveis complementados ($S = 1$ e $R = 0$, ou $S = 0$ e $R = 1$), aquele que em nível 1 levar o FF a reset é indicado por R e para set por S. No nosso exemplo fazendo $R = 1$ obriga a $Q = 0$, que dá entrada na NOR inferior, juntamente com $S = 0$, levando $\bar{Q} = 1$, ou seja, reset. Por outro lado, se $S = 1$ então $\bar{Q} = 0$, dando entrada na NOR superior com $R = 0$, resultando $Q = 1$. As alternativas possíveis de níveis lógicos nos outputs são chamadas "estados".

Colocando simultaneamente $R = 1$ e $S = 1$ o FF é desabilitado, apresentando $Q = \bar{Q} = 0$ independente de qualquer outra condição lógica. Para $R = 0$ e $S = 0$ o FF manterá os mesmos outputs que tinha anteriormente. Surge daí a aplicação de FF como elemento de memória eletrônica: armazena-se em Q e \bar{Q} os níveis desejados (pela manipulação de R e S) e em seguida os inputs são colocados em nível 0, obrigando o FF a manter ("memorizar") o estado anterior.

A tabela abaixo registra níveis lógicos de inputs e outputs no FF - SR - NOR.

A utilização de portas NAND (figura 5-2), segue os mesmos princípios de operação, com pequenas diferenças lógicas. O input superior leva a set ($Q=1$)

INPUTS		OUTPUTS	
S	R	Q	\bar{Q}
\emptyset	1	\emptyset	1
1	\emptyset	1	\emptyset
1	1	\emptyset	\emptyset
\emptyset	\emptyset	(Estado anterior)	

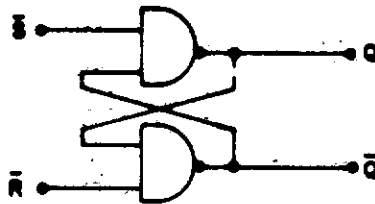


figura 5-2 - Flip-Flop SR - NAND

quando é colocado em nível \emptyset , sendo por isso representado por \bar{S} (a barra acima da letra indica que o terminal é ativo quando levado a nível \emptyset); igualmente, com $\bar{R} = \emptyset$ temos $\bar{Q} = 1$ e $Q = \emptyset$, reset. Os dois inputs em \emptyset provocam $Q = \bar{Q} = 1$ e em 1 mantém-se o estado anterior. A tabela abaixo mostra a tabela de FF-SR-NAND.

INPUTS		OUTPUTS	
\bar{S}	\bar{R}	Q	\bar{Q}
\emptyset	1	1	\emptyset
\emptyset	\emptyset	1	1
1	1	(Estado anterior)	

O símbolo do FF-SR-NOR é visto na figura 5-3A e FF-SR-NAND na figura 5-3B.

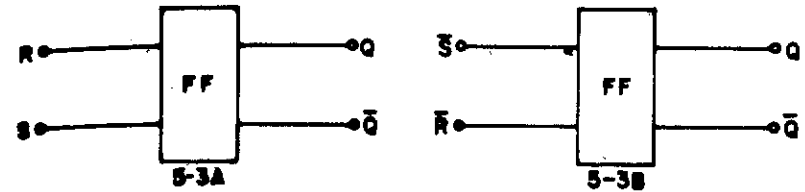


figura 5-3 - Símbolo do flip-flop SR (A) NOR (B) NAND

(2) - O CLOCK

A partir deste ponto passaremos a estudar dispositivos cujas operações dependem do fator tempo referenciado por outro sinal - processo denominado sin cronização. Chamamos ainda de "sequencial" a parte da Eletrônica Digital que trata de operações sincronizadas.

CLOCK (do Inglês, "relógio") é qualquer dispositivo emissor de sinais de sincronização. Quando não há necessidade de extrema precisão usa-se um circuito multivibrador. Nos casos onde exatidão do tempo é fundamental emprega-se um circuito com cristal, como no exemplo da figura 5-4.

O cristal (geralmente escrito "XTAL") funciona por efeito pizzoelétrico, emitindo sinais de máxima precisão na frequência para a qual foi fabricado. É disponível comercialmente em formato semelhante a um pequeno cadeado e seu símbolo é como consta na figura 5-4.

Em descrição de circuitos digitais é comum apresentar somente o terminal de output do clock (indicando-o pela letra 'C'), tal como é feito com a fonte de alimentação. Assim, para as exposições a seguir o leitor subentenda que o sinal de clock indicado pelo 'C' é gerado em algum outro ponto do sistema, através de um circuito adequado.

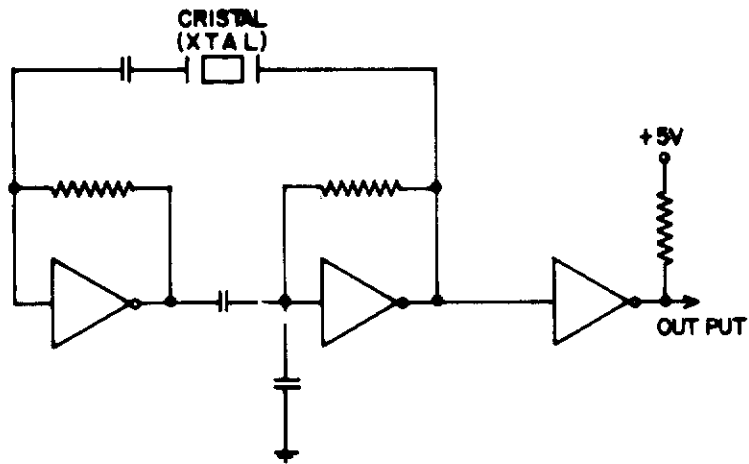


figura 5-4 - Circuito de clock com cristal

(3) FF - SR SINCRONIZADO

No FF-SR-NAND sincronizado os inputs são combinados em portas NAND com o clock (figura 5-5) Com $C = \emptyset$ o FF é desabilitado, resultando $\bar{S}' = \bar{R}' = 1$ independente de S e R, os outputs mantendo o estado anterior. Para $C = 1$ os inputs são habilitados, aparecendo nas saídas de suas respectivas portas níveis dependendo de S e R.

Na operação do FF-SR-NAND são seguidos os seguintes passos:

- (1) C é levado a \emptyset , desabilitando o FF
- (2) S e R são colocados nos níveis desejados
- (3) C é levado a 1, habilitando os inputs
- (4) Q e \bar{Q} adaptam-se aos níveis dos inputs habilitados
- (5) C volta a \emptyset , desabilitando o FF que continua mantendo os níveis de Q e \bar{Q} .

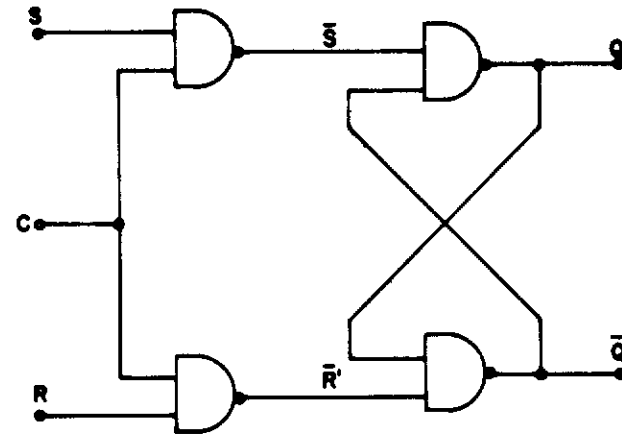


figura 5-5 - FF - SR - NAND sincronizado

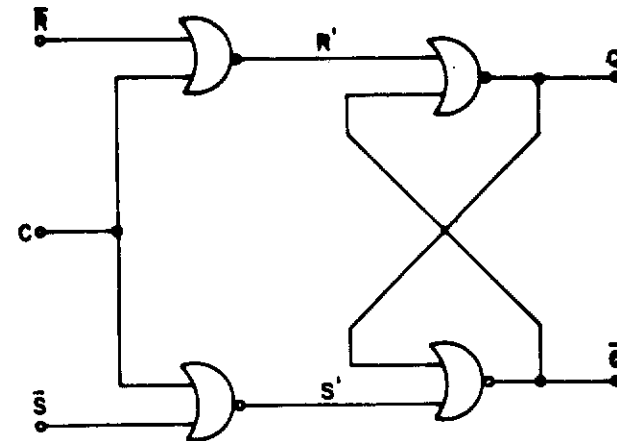


figura 5-6 - FF - SR - NOR sincronizado

No FF-SR-NOR sincronizado (figura 5-6) a desabilitação ocorre com $C = 1$, fazendo $R' = S' = \emptyset$, Q e \bar{Q} mantendo os níveis anteriores. Com $C = \emptyset$ os inputs são habilitados e passam a determinar o estado do FF. Observe que (após habilitação $C = \emptyset$) $\bar{R} = \emptyset$ resulta $R' = 1$ e $Q = \emptyset$, reset. Igualmente $S = \emptyset$ provoca $S' = 1$, $\bar{Q} = \emptyset$ e $Q = 1$, set.

(4) CONTROLE DIRETO

Muitas vezes é necessário um controle direto do FF que determine os níveis dos outputs independente dos inputs. Usa-se então uma última porta, à qual é conectado o terminal direto - R_d e S_d (figura 5-7). Se for uma OR o controle direto é feito com $R_d = S_d = 1$, que obriga $Q = \bar{Q} = 1$ apesar de quaisquer níveis nos inputs. Com $R_d = S_d = \emptyset$ o FF opera livremente segundo sua lógica. Também é possível construir controle direto com porta AND, que neste caso controla o FF com $\bar{R}_d = \bar{S}_d = \emptyset$ obrigando a $Q = \bar{Q} = \emptyset$, e deixa-a livre com $R_d = S_d = 1$.

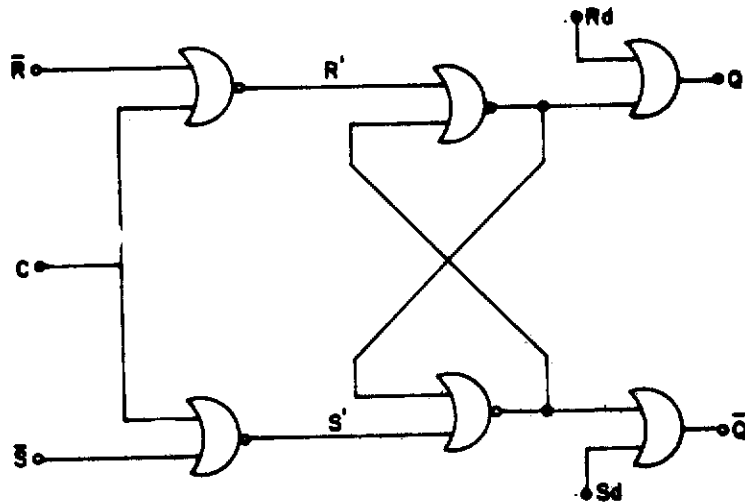


figura 5-7 - Controle direto do FF.

(5) FF - MASTER - SLAVE

Outro tipo é o MASTER-SLAVE (do inglês, "senhor - escravo"), assim chamado porque consiste num flip-flop comandando outro. Podem ser empregadas portas NOR ou NAND, sendo que usaremos as últimas como exemplo.

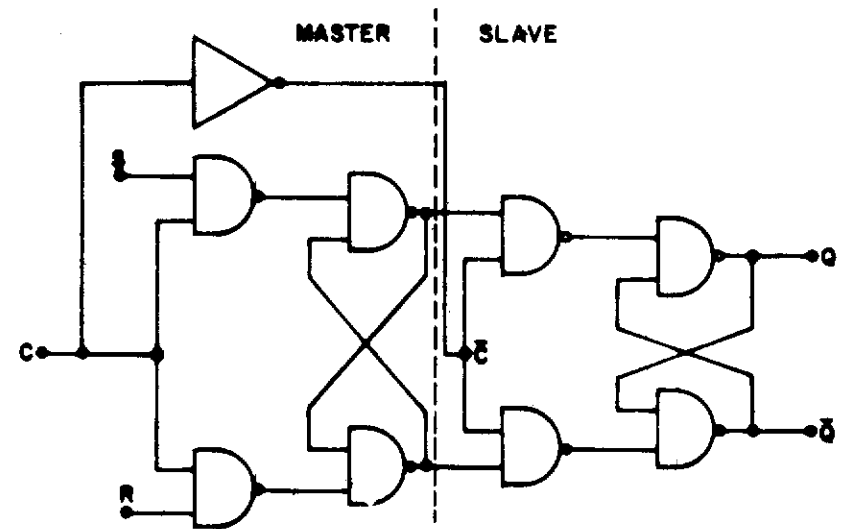


figura 5-8 - FF master-slave

O sinal de clock C é ligado diretamente ao flip-flop master, enquanto o slave o recebe invertido (\bar{C}) juntamente com os outputs do primeiro (figura 5-8). Com $C = 1$ o master é habilitado, adaptando seus outputs aos níveis de S e R . Durante este tempo o slave está desabilitado ($\bar{C} = \emptyset$), impedindo que os níveis dos outputs do master atuem sobre ele. Quando C volta a \emptyset ocorre $\bar{C} = 1$, finalmente habilitando o estado de master como input do slave.

(6) FF - AC ACOPLADO

No FF-AC ACOPLADO é intercalado um capaci-

tor entre o input e a porta, além de um resistor para retorno ao terra (figura 5-9).

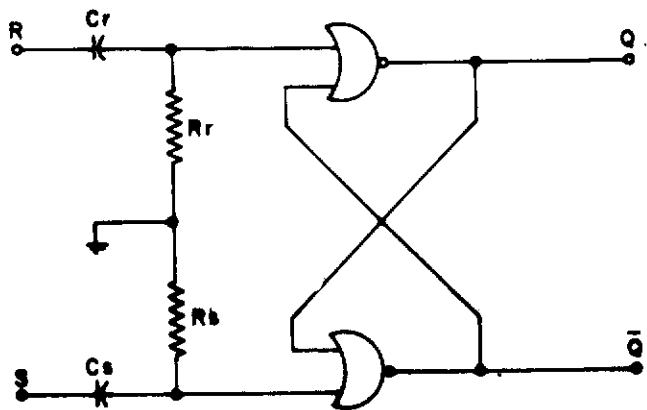


figura 5-9 - FF - AC acoplado

Na transição positiva do input (de nível \emptyset a 1), ou negativa (de nível 1 para \emptyset), o sinal atravessa o capacitor e descarrega através do resistor no terra, provocando pela Lei de Ohm uma voltagem que altera os outputs.

Levando R de \emptyset a 1 (transição positiva) a corrente (positiva) atravessa C_r e r_r , retornando ao terra. A queda de tensão no resistor resulta em voltagem positiva no extremo superior de r_r , que é input 1 para a NOR, setando o flip-flop. Com R indo de 1 para \emptyset a corrente (positiva) sai do terra, atravessa r_r e C_r em direção à entrada, provocando tensão negativa no extremo superior de r_r - nível \emptyset para a NOR que é setado (se a outra NOR, \bar{Q} , estiver resetada). O mesmo raciocínio se aplica ao input S. A tabela de inputs e estados é exatamente igual à que vimos

para o FF-SR. É possível, também, usar portas NAND.

Para sincronizar o FF-AC ACOPLADO com um clock basta acrescentar outra porta antes do capacitor (figura 5-10).

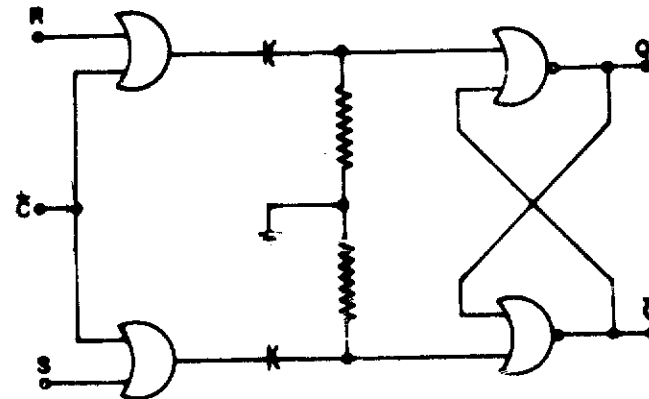


figura 5-10 - FF - AC acoplado sincronizado

Enquanto $\bar{C} = 1$ o FF está desabilitado e procede-se as alterações de níveis desejados para os inputs. Quando \bar{C} é levado de 1 para \emptyset (transição negativa) os inputs atuam sobre os outputs.

Considere que $\bar{C} = 1$, o que resulta nas duas portas OR em 1, independente dos inputs. Fazendo $R = 1$ e $S = \emptyset$ as duas portas OR continuam em 1, não havendo transição nem mudanças de estado do flip-flop. Ao levar \bar{C} de 1 para \emptyset , entretanto, a saída da OR inferior vai a \emptyset e a superior permanece em 1, resetando o flip-flop. Invertendo os níveis de R e S obtemos um set.

Observe que se \bar{C} for deixado em nível \emptyset permanentemente o flip-flop se comporta como acoplado comum, sem sincronismo.

(7) FF - ARMAZENAMENTO CAPACITIVO

O FF-ARMAZENAMENTO CAPACITIVO é projetado com duas portas recebendo os inputs e o clock, os sinais de suas saídas sendo armazenados em capacitores que são entradas para o flip-flop propriamente dito, sincronizado pelo complemento \bar{C} do clock (figura 5-11).

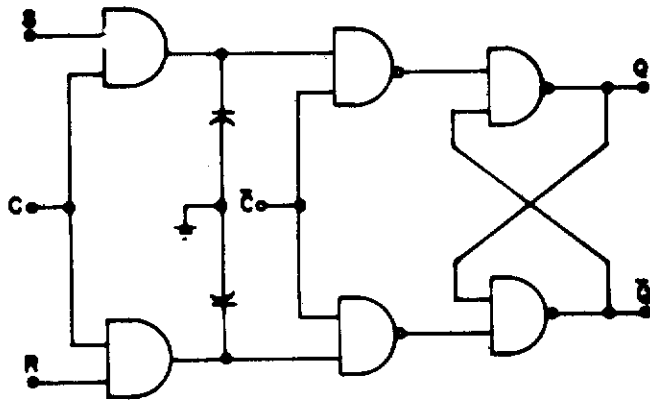


figura 5-11 - FF - armazenamento capacitivo

Com $C = 1$ os níveis de S e R são armazenados nos capacitores, mas não atuam sobre as entradas da NAND que estão desabilitadas por $\bar{C} = 0$. Quando ocorre a transição negativa de C (de 1 para 0) os inputs S e R são desabilitados e as portas NAND habilitadas por $\bar{C} = 1$, alterando-se os outputs Q e \bar{Q} segundo os níveis armazenados nos capacitores.

Para FF - armazenamento capacitivo com portas NOR basta comutar as transições do clock, fazendo \bar{C} na primeira fase e C na segunda.

(8) FF - ATRASO DE PROPAGAÇÃO

Em qualquer porta lógica o sinal não se propaga instantaneamente, demorando algum tempo para percorrer do input ao output.

Referindo-se à figura 5-12, tomemos $IN_2 = 0$ e inicialmente também $IN_1 = 0$. Neste caso a saída da NOR é 1, habilitando a AND. Se levarmos IN_1 de 0 para 1 (transição positiva) demorará certo tempo até a saída da NOR alterar, e neste ínterim ela permanece em 1. Como a transição de IN_1 é sentida instantaneamente na outra entrada da AND o output fica em nível 1 ($IN_1 = 1$ AND NOR = 1). Passando o tempo de propagação na NOR sua saída vai a 0 e também o output.

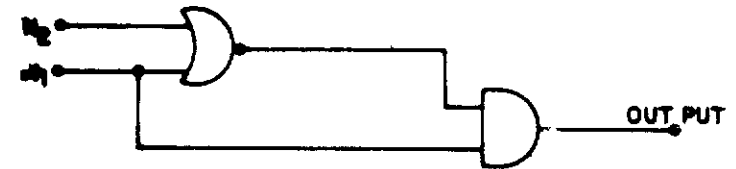


figura 5-12 - Circuito lógico de atraso de propagação

Façamos agora $IN_2 = 1$ e IN_1 inicialmente 0. A saída da NOR será 0, levando o output a 0. Quando IN_1 passar por transição positiva (de 0 para 1) não haverá alteração na saída da NOR - mesmo depois do tempo de propagação - que permanecerá em 0 e manterá o output em 0.

Portanto, o nível de IN_2 determina qual o pulso que aparecerá na saída da AND e é aproveitada no FF - ATRASO DE PROPAGAÇÃO como input, enquanto IN_1 é implementado pelo clock (figura 5-13).

Com $C = 0$ colocam-se os inputs R e S nos níveis desejados. Levando C de 0 a 1 resulta na saída

da AND um sinal correspondente à lógica de atraso de propagação, que é usado como entrada do flip-flop propriamente dito (portas NOR).

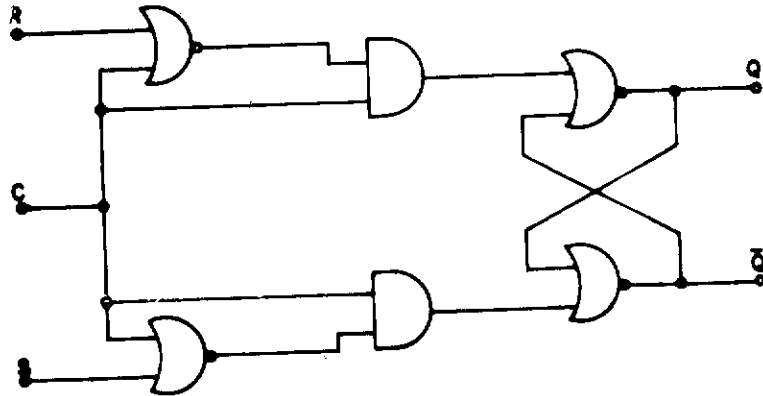


figura 5-13 - FF - atraso de propagação

Para usar portas NAND na entrada do circuito de atraso de propagação mudamos C para \bar{C} , habilitando o flip-flop na transição negativa do clock.

(9) FF - JK

O FF-JK é um sistema master-slave onde os outputs realimentam os inputs, conectando Q à entrada que gera \bar{Q} e vice-versa através de uma NAND (figura 5-14). Os inputs são denominados de J e K, para diferenciar de S e R. O símbolo do FF-JK é visto na figura 5-15.

A aplicação mais comum do JK é o processo de "toggle". Mantendo $J = K = 1$ os outputs Q e \bar{Q} muda

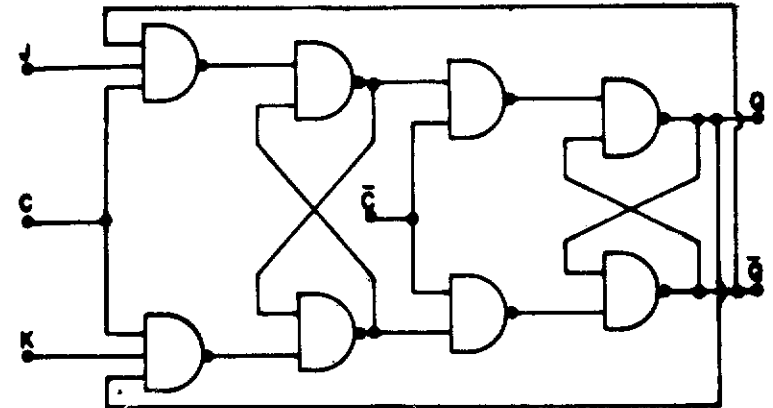


figura 5-14 - FF - JK

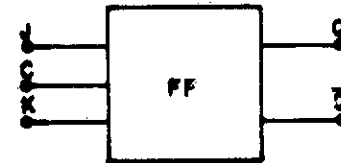


figura 5-15 - Símbolo do FF - JK

rão de nível a cada transição negativa do clock (figura 5-16). Geralmente o terminal de clock é conectado a uma fonte de sinal, que aparecerá padronizado como onda quadrada nos outputs. Isso permite que sinais irregulares ou de formas inadequadas sejam padronizadas para manipulação digital.

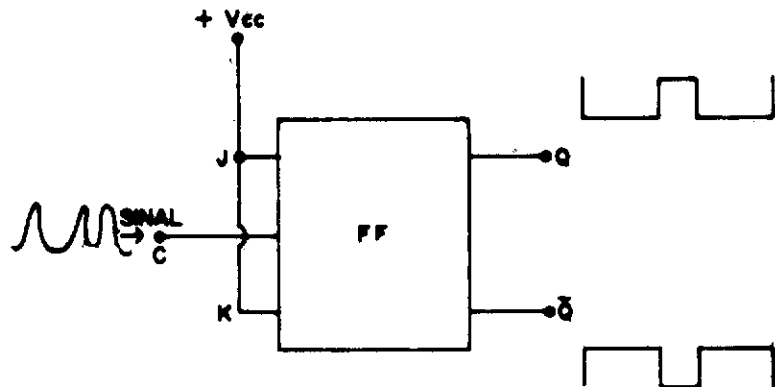
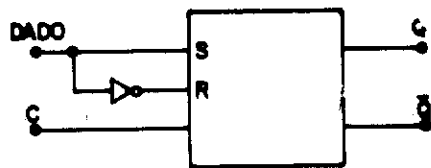


figura 5-16 - "Toggle"

(10) FF - TIPO D (DELAY-ATRASSO)

O FF - TIPO D é projetado a partir de um FF-SR, conectando os dois inputs através de um inversor (figura 5-17). O sinal colocado nos inputs aparecerá no output com atraso de um clock.



figura,5-17 - FF - tipo D

Lembremos que no FF-SR-NAND os inputs são assentados enquanto $C = \emptyset$ e aparecem no output quando C volta de 1 a \emptyset (transição negativa). Se fizermos $S = 1$ quando $C = \emptyset$ ao levar o clock a 1 teremos $Q = 1$; quando C volta a \emptyset Q mantém-se setado.

O dado a ser processado é uma forma de onda com transições sincronizadas pelo clock (figura 5-18) e dá entrada no input S. Analisando as transições positiva e negativa do clock é fácil verificar que o output terá a mesma forma de onda do sinal, mas

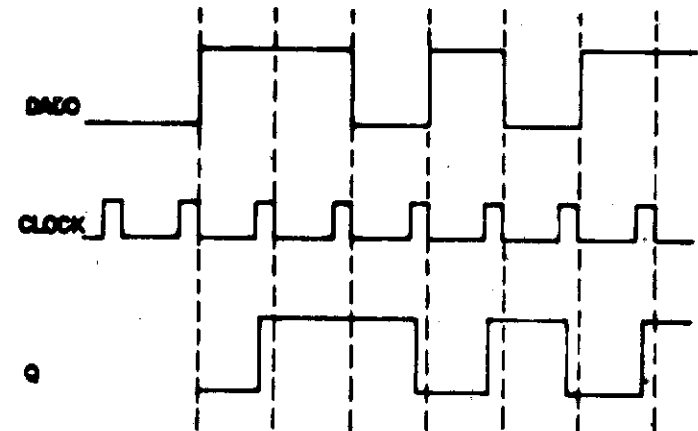


figura 5-18 - Formas de onda no RR - tipo D.

retardando um clock. A figura 5-19 apresenta o processo mais detalhadamente:

- (1) em A o clock está em \emptyset e o flip-flop recebe no input o nível \emptyset do dado neste instante; entre tanto o output não é alterado porque a porta NAND da entrada está desabilitada por $C = \emptyset$ (veja figura 5-5).

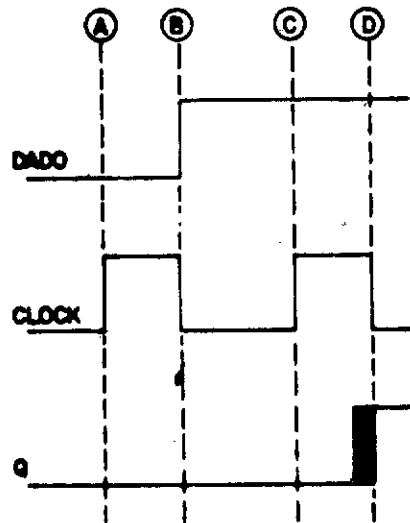


figura 5-19 - atraso no FF - tipo D.

- (2) entre A e B o clock vai a 1 (transição positiva), habilitando a NAND, o que provoca alteração em Q, levando-o a 0.
- (3) em B ocorre transição negativa do clock, estabilizando o nível lógico do output e desabilitando o input.

O FF - TIPO D não precisa ser um SR, sendo possível implementá-lo com o circuito da figura 5-20. Embora aparentemente mais complexo esse modelo é de fácil fabricação.

(11) O FLIP-FLOP NA PRÁTICA

Apresentamos em traços gerais os princípios de funcionamento dos vários tipos de flip-flop. A

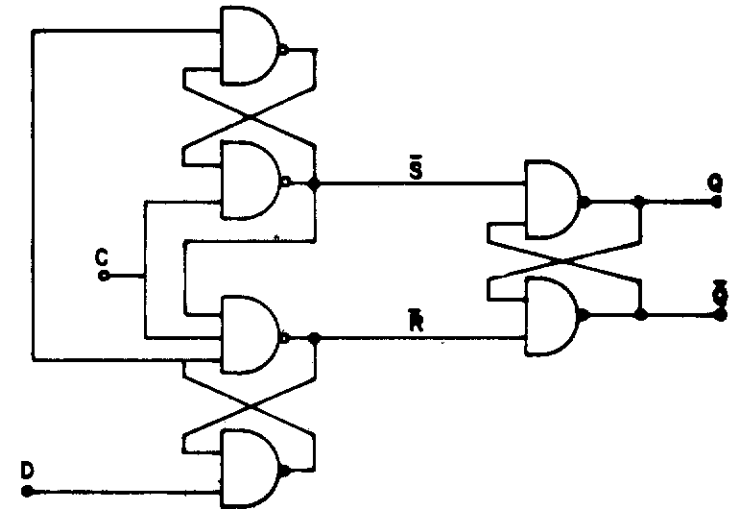


figura 5-20 - Circuito lógico de FF - tipo D

aplicação deles é tão extensa que se pode dizer ser o principal subsistema da Eletrônica Digital.

Na prática o flip-flop pode vir "pronto" no CI, como o 74LS 175 da figura 5-21. Trata-se de 4 flip-flops tipo D, o índice antes das letras diferenciando-os. Assim, 2D indica entradas de dados D do 2º flip-flop e 3Q o output do terceiro.

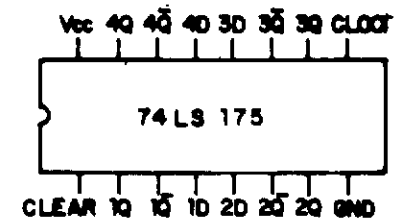


figura 5-21 - CI 74 LS 175 com 4 flip-flops tipo D.

Outra possibilidade é o próprio usuário montar o circuito do flip-flop a partir de um CI que contenha as portas lógicas necessárias, tal como o FF-SR da figura 5-22, que utiliza um 7400. Note que ainda sobram duas NAND para outro flip-flop.

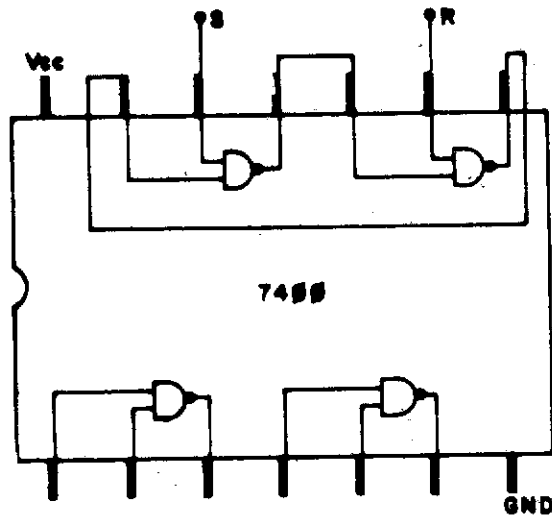


figura 5-22 - Montagem de flip-flop a partir de CI com as portas lógicas.

(12) MULTIVIBRADOR MONOESTÁVEL

O MULTIVIBRADOR MONOESTÁVEL é semelhante ao flip-flop, porém é instável num de seus estados, voltando à configuração estável sem necessidade de sinal externo. Por exemplo, ele pode ficar setado indefinidamente (estável), mas se for colocado em reset permanecerá pouco tempo nesta situação e volta espontaneamente a setado.

Devido a característica de ser levado à instabilidade por um sinal externo o monoestável é conhecido por "one-shot" ("um - tiro"). A transição de

instável para estável, produzindo um sinal em tempo determinado após o trigger externo, é usada para aplicações de circuito de atraso ("delay")

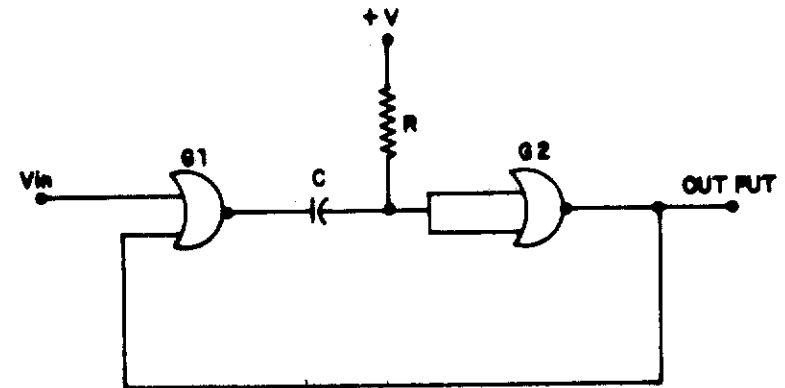


figura 5-23 - Multivibrador monoestável

Na figura 5-23 temos um monoestável típico. No estado estável as duas entradas da NOR G1 estão em \emptyset , produzindo 1 na saída que é bloqueado pelo capacitor C (somente sinais variáveis atravessam capacitores); não há corrente pelo resistor R, o que coloca +V integralmente (nível 1) nas entradas da NOR G2, resultando nível \emptyset na sua saída. Quando ocorre um pulso positivo em V_{in} (trigger) ele é convertido em nível \emptyset na saída de G1, atravessando C e entrando em G2, que passa a apresentar OUTPUT = 1 (estado instável). Após um tempo (determinado pela constante RC) o capacitor volta a se carregar com +V, levando a 1 a entrada de G2 e a \emptyset seu OUTPUT (estado estável).

Note que o tempo de atraso (estado ins-

ável) é determinado pela constante RC e não por V_{in} (figura 5-24). Fazendo R um potenciômetro obtém-se uma forma de onda no OUTPUT com um pulso de duração pré-determinada.

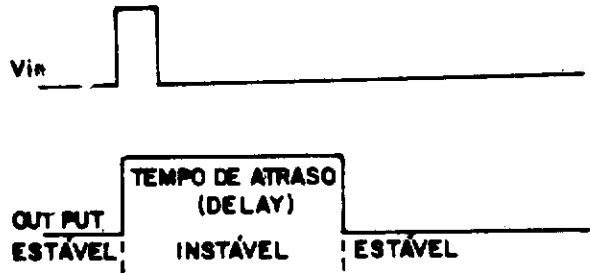


figura 5-24 - Formas de onda no multivibrador monoestável.

Outro multivibrador monoestável é visto na figura 5-25. No estado estável $S = R = 0$, $Q = 0$ e $\bar{Q} = 1$. O trigger para a instabilidade é um pulso positivo em S, levando \bar{Q} a 0 e propagando-se como sinal negativo através do capacitor C e do resistor r, resultando $Q = 1$. O capacitor volta a se carregar pelo resistor r (no tempo rC), o que provoca $R = 1$ e daí $Q = 0$, além de $\bar{Q} = 1$, como no estado inicial. No vamente pode ser usado um potenciômetro no lugar de r para controlar a largura do pulso de atraso.

(13) MULTIVIBRADOR AESTÁVEL

O MULTIVIBRADOR AESTÁVEL é um mero oscilador, apresentando instabilidade nos dois estados

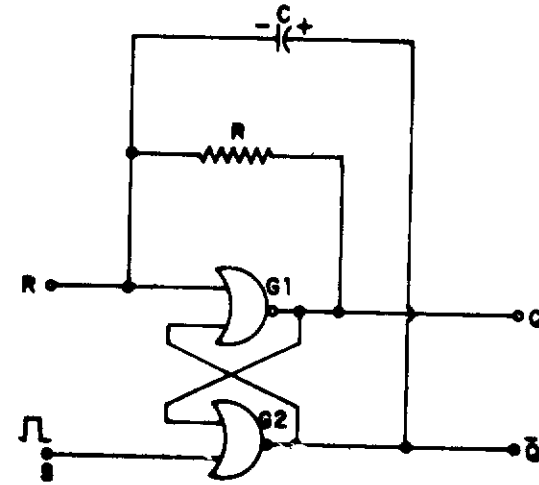


figura 5-25 - Circuito alternativo para multivibrador monoestável

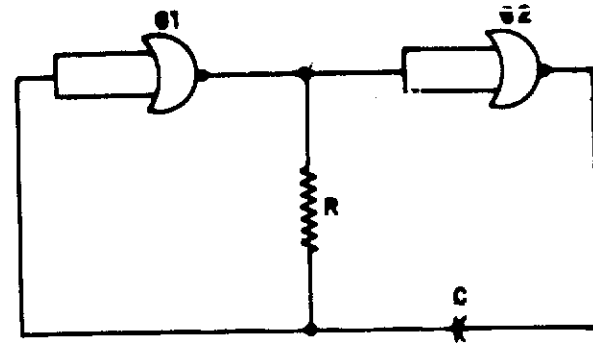


figura 5-26 - Multivibrador aestável

(set e reset). A figura 5-26 mostra um circuito que executa esta função.

Com as entradas de G 1 em nível 1 sua saída será \emptyset , levando a saída de G 2 a 1. G 1 entretanto força a placa esquerda do capacitor C ao nível \emptyset , através de r, invertendo todas as polaridades anteriores. O processo se repete indefinidamente, com o multivibrador setado e resetado alternadamente. O tempo de oscilação é controlado pela escolha dos valores de r e C.

CAPÍTULO 6 – REGISTRO E CONTADORES

Registros e contadores são aplicações de cascatas de flip-flop, servindo para armazenar e manipular bits. Na verdade o flip-flop é um registro de um só bit. Como sequências de bits correspondem a formas de ondas estes dispositivos serão usados também para manipulá-las.

Neste capítulo estudaremos o registro de deslocamento, contador de ripple, scaler, contador circular e gerador de sequência. Outras configurações existem, mas são variações dos arranjos abordados aqui

(1) REGISTRO DE DESLOCAMENTO

O REGISTRO DE DESLOCAMENTO é uma cascata de flip-flops, com o output (Q) de cada um conectado ao input (D) do próximo (figura 6-1A).

Na figura 6-1B temos as formas de onda do sinal deslocando-se no registro. Todos os flip-flops estão inicialmente resetados ($Q = \emptyset$). Ao primeiro clock o dado D_0 (nível 1, como exemplo) é transferido para Q_0 na transição negativa; todos os outros flip-flops permanecem resetados pois tinham $D = \emptyset$ durante o clock. Quando este dá o 2º pulso o FF1 recebe $Q_0 = 1$ no input D 1 e coloca seu output Q_1 em 1. Os outros flip-flops continuam resetados, pelo mesmo motivo anterior; além disso o dado D_0 baixa para \emptyset (como exemplo), resetando FF0. Esse processo se repete, deslocando para a direita de flip-flop para flip-flop a cada clock até chegar ao output do último (Q_3).

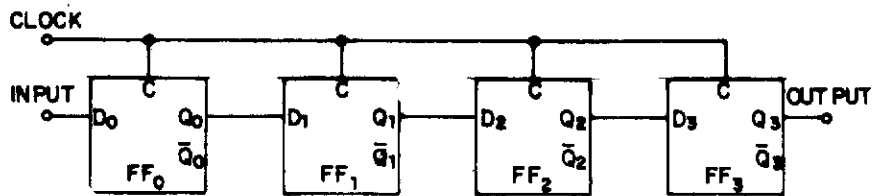


FIG. 6-1A

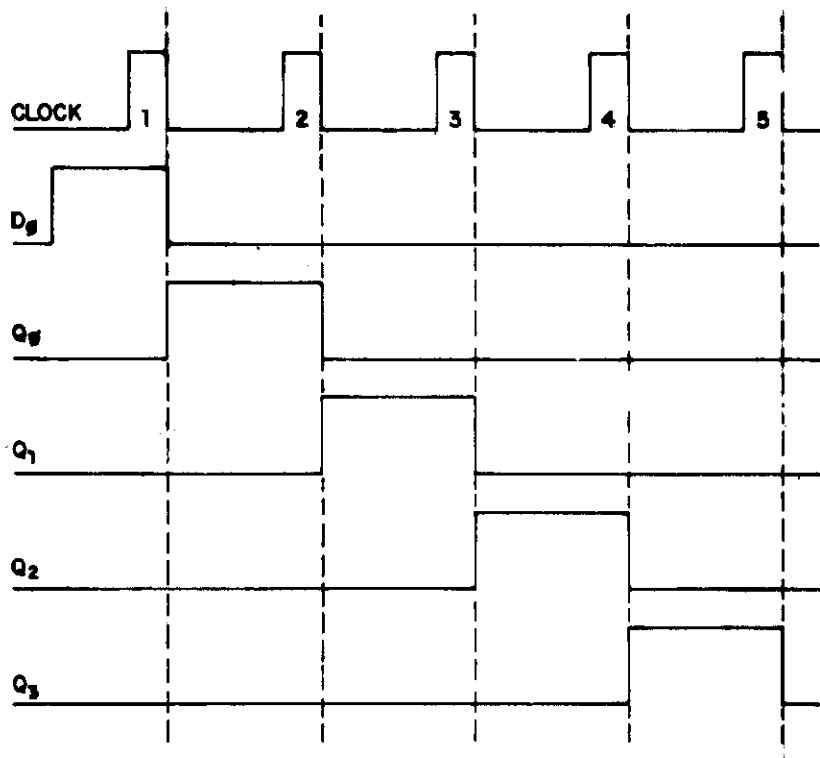


FIG. 6-1B

figura 6-1 - Registro de deslocamento (A) circuito (B) formas de onda

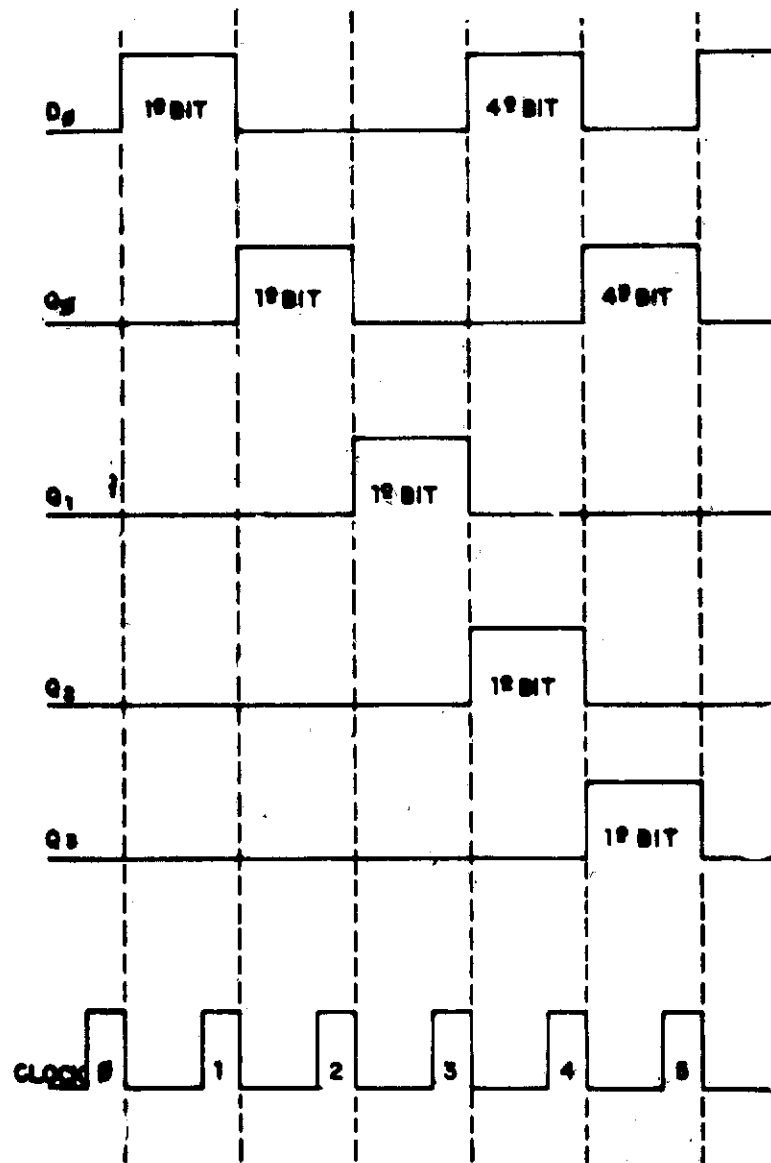


figura 6-2 - Forma de onda deslocando no registro de deslocamento.

Por simplicidade usamos um dado que é nível inicialmente e depois permanece todo o tempo em 0. Na verdade ele pode ser um sinal com qualquer forma, como na figura 6-2. Se olharmos os outputs a cada clock veremos que a forma de onda na entrada vai se deslocando para a direita.

Como se vê na figura 6-2 o primeiro bit a ser injetado é também o primeiro a aparecer no output do último flip-flop - processo que se denomina FIFO ("first in - first out", primeiro que entra - primeiro que sai).

Dizemos que o dado é "serial" quando seus bits são apresentadas em série, um após o outro. Dado "paralelo" refere-se à apresentação simultânea de todos os bits.

Uma aplicação importante do registro de deslocamento é a transformação de dado serial em paralelo (figura 6-3). Os bits dão entrada e são deslocados até todo o dado estar no registro. Nos terminais de output (Q) eles ficam disponíveis simultaneamente ou seja, em paralelo.

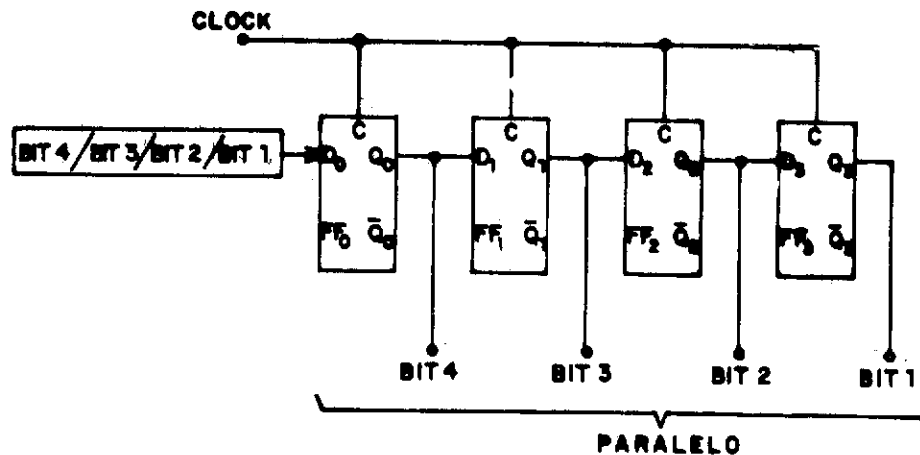


figura 6-3 - conversão serial para paralelo

Uma variação do circuito anterior é o REGISTRO DE DESLOCAMENTO ESQUERDA/DIREITA (figura 6-4). Com $R = 1$ e $L = 0$ os dados são introduzidos no terminal D_i e deslocam-se para a direita; as portas AND inferiores estão desabilitadas e não têm influência na operação. Com $R = 0$ e $L = 1$ as portas AND superiores são habilitadas. Os dados introduzidos em D_i deslocam-se da direita para a esquerda; as portas AND superiores estão desabilitadas e não têm influência na operação.

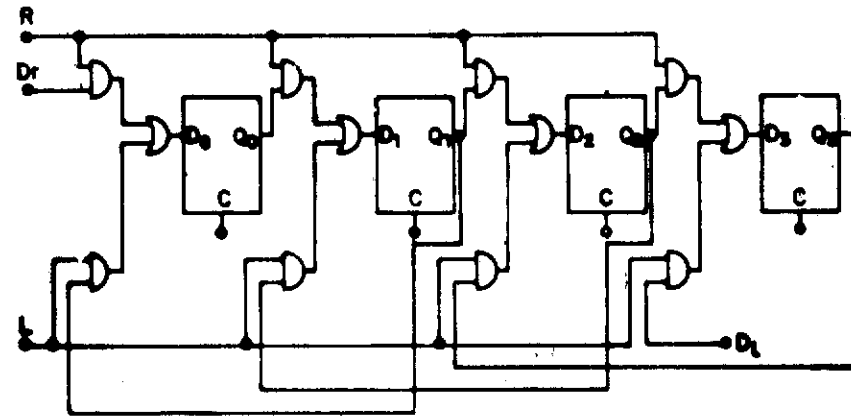


figura 6-4 - Registro de deslocamento esquerda/direita.

(2) CONTADOR DE RIPPLE

O CONTADOR DE RIPPLE é um arranjo de FF-JK (figura 6-5), com os outputs de cada um conectado ao clock do próximo; os inputs J e K são mantidos permanentemente em nível 1 e o sinal a ser "contado" é injetado no clock do primeiro flip-flop.

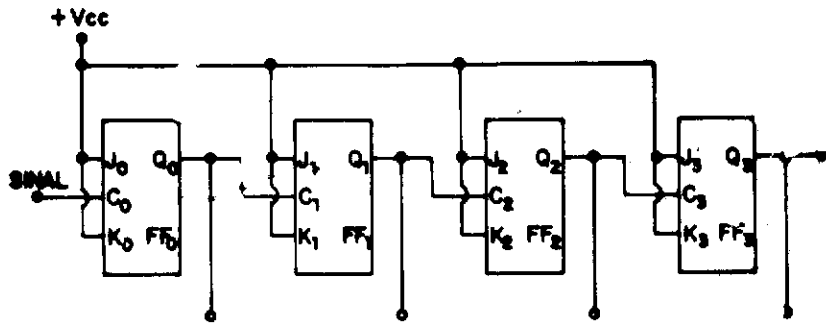


figura 6-5 - Contador de ripple

Na figura 6-6 descrevemos as formas de ondas nos outputs dos flip-flops, tomando como exemplo um sinal de clock injetado em C_0 .

O FF-JK muda de estado a CADA TRANSIÇÃO NEGATIVA de seu clock. Portanto, Q_0 terá a metade da frequência do sinal. Como Q_0 serve de clock para FF 1 teremos Q_1 com metade de sua frequência, e assim por diante, cada flip-flop dividindo a frequência do anterior por 2.

Se tomarmos os outputs dos flip-flops como bits de um binário, sendo LSB = Q_0 e MSB = Q_3 , veremos que a cada clock o número está avançando uma unidade. No nosso exemplo:

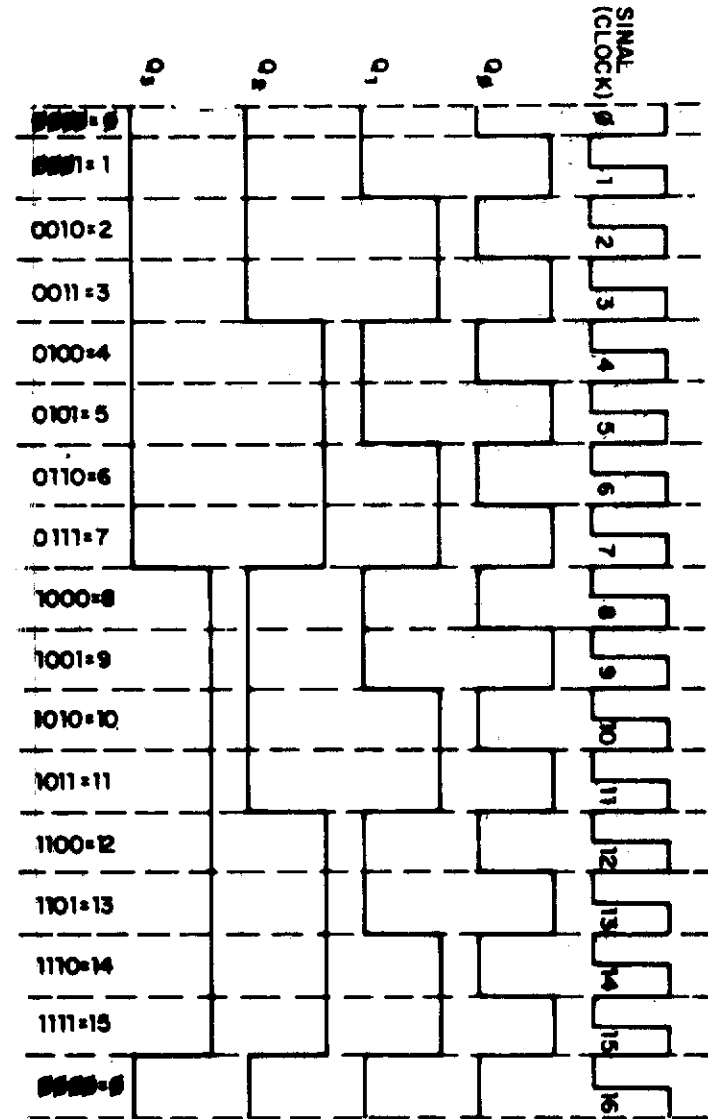


figura 6-6 - Formas de onda e contagem no contador de ripple.

O U T P U T S						
Pulso do clock (sinal)	Q ₃	Q ₂	Q ₁	Q ₀	Binário	Decimal
0	0	0	0	0	0000	0
1	0	0	0	1	0001	1
2	0	0	1	0	0010	2
3	0	0	1	1	0011	3
4	0	1	0	0	0100	4
5	0	1	0	1	0101	5
6	0	1	1	0	0110	6
7	0	1	1	1	0111	7
8	1	0	0	0	1000	8
9	1	0	0	1	1001	9
10	1	0	1	0	1010	10
11	1	0	1	1	1011	11
12	1	1	0	0	1100	12
13	1	1	0	1	1101	13
14	1	1	1	0	1110	14
15	1	1	1	1	1111	15
16	0	0	0	0	0000	0

Ou seja, o contador de ripple conta quantos pulsos entraram em seu terminal C0, apresentando o resultado em forma binária nos outputs.

Mostramos um caso onde o sinal é um clock, com repetição periódica. Na verdade não há necessidade de periodicidade e aí encontra-se a grande utilidade desse dispositivo. O medidor de fluxo da figura 6-7 é um exemplo típico.

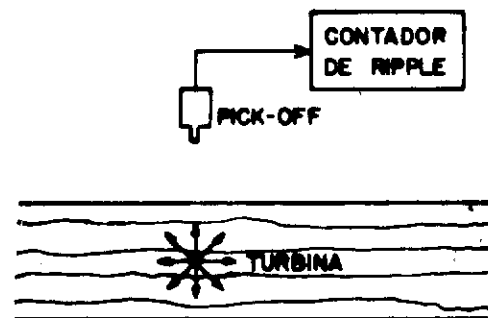


figura 6-7 - Medidor de fluxo

Uma turbina move em torno do eixo com velocidade proporcional à quantidade de fluxo que flui por ela. Quando a extremidade magnética de uma pã (indicada por setas) passa sob uma bobina pick-off é emitido um sinal que dá entrada no contador de ripple. Com muito fluxo a frequência do sinal emitido é grande, pouco fluxo provocará baixa frequência, mas em qualquer caso a contagem no contador de ripple será proporcional à quantidade de fluido que já atravessou a turbina.

Analisando a figura 6-6 e a respectiva tabela notamos que após 16 pulsos (do sinal) o contador de ripple volta a seu estado original (todos os flip-flops resetados), recomeçando-se a contagem. Dizemos que se trata de um contador MÓDULO 16. Assim, módulo é a quantidade de pulsos na entrada necessários para o contador voltar a seu estado original.

O estado do contador de ripple é determinado pelo output de seus flip-flops, que por sua vez podem assumir dois níveis - 0 ou 1. Surge daí a fórmula para determinar o módulo:

$$\text{MÓDULO} = 2^N,$$

onde N é o número de flip-flops. No exemplo anterior tínhamos 4 flip-flops, portanto

$$\text{MÓDULO} = 2^4 = 16.$$

O contador de ripple módulo 10 é denominado CONTADOR DE DÉCADA, contando de 0 a 9 e voltando a seu estado original, 0. Obviamente sua importância está ligada à sua capacidade de contar diretamente em base decimal.

O resultado da contagem é apresentado em binários nos outputs dos flip-flops. Muitas vezes é preciso habilitar outro dispositivo conforme a resposta obtida. Usa-se então um DECODIFICADOR que converte o resultado binário em um sinal elétrico numa de suas saídas.

Na figura 6-8 mostramos um contador de ripple módulo 8 e um circuito decodificador. Verifique que cada contagem habilitará sua respectiva porta AND (nível 1) e manterá todas as outras desabilitadas (nível 0). Por exemplo, para 010 somente a AND 2 será levada a nível 0. Observe que estamos usando a propriedade de \bar{Q} ser o complemento de Q.

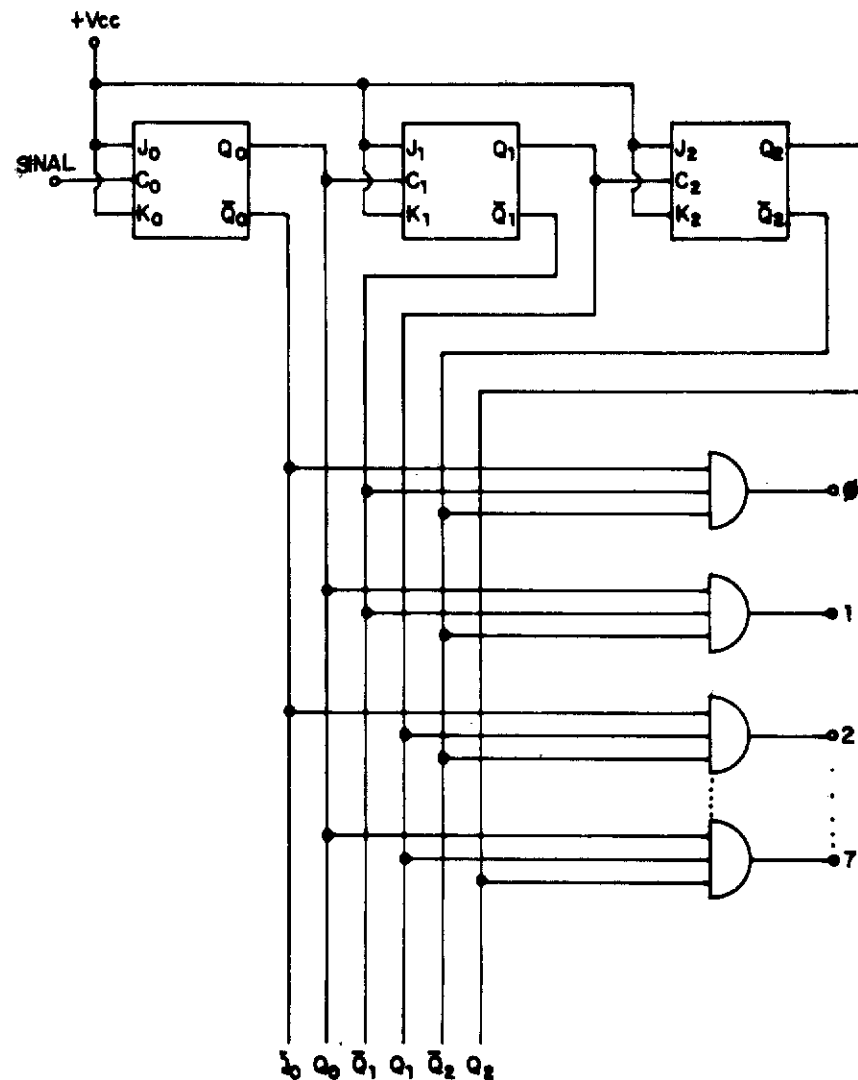


figura 6-8 - Decodificador de contador de ripple

(3) SCALER

Voltando à figura 6-6 vemos que a frequência do sinal da entrada vai sendo dividida por 2 a cada flip-flop, da esquerda para a direita. Como estudado no Capítulo 2 este é o processo de scaling, denominando-se SCALER o contador de ripple que o executa.

Na prática o contador de ripple está contido num CI, com os outputs Q disponíveis como terminais (figura 6-8). Escolhendo o terminal adequado obtemos a divisão de frequência desejada.

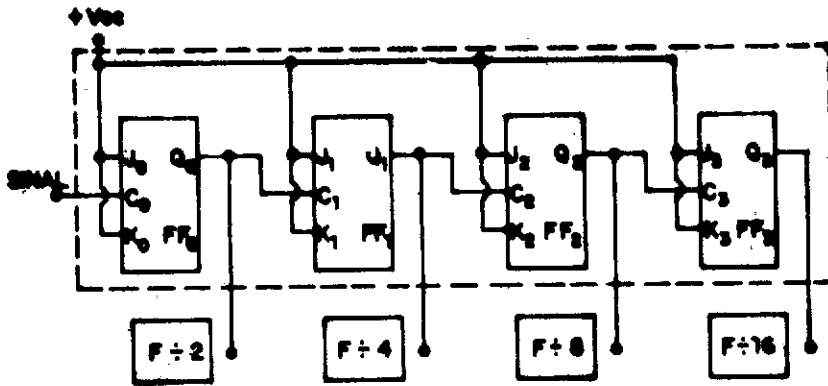


figura 6-8 - CI contendo contador de ripple.

(4) CONTADOR CIRCULAR

O CONTADOR CIRCULAR é uma cascata de flip-flops tipo D (figura 6-9). O output dá entrada como dados D no próximo flip-flop. O sinal é conectado ao terminal de clock.

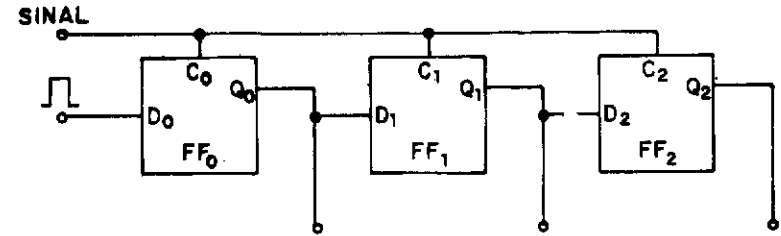


figura 6-9 - Contador circular

No primeiro pulso do sinal injeta-se o bit 1 em D_0 . A partir daí a cada pulso do sinal (funcionando como clock) o bit 1 desloca-se uma casa para a direita, tal como no registro de deslocamento.

Para saber o resultado da contagem basta verificar qual output encontra-se com o bit 1. Por exemplo, se $Q_3 = 1$ então a contagem é 3.

Embora muito simples o contador circular leva a desvantagem de usar muitos flip-flops. Será preciso tanto quanto for o maior valor da contagem.

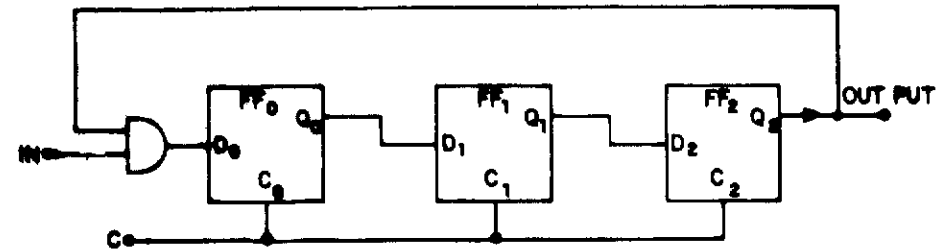


FIG. 6-10 CONTADOR CIRCULAR SWITCH-TAIL

O CONTADOR CIRCULAR SWITCH-TAIL (figura 6-9) também é uma cascata de flip-flops tipo D, conectando-se a saída do último deles à entrada do primeiro através de uma porta, que recebe ainda um sinal IN do exterior.

A cada clock o sinal IN é deslocado uma casa para a direita até chegar ao último flip-flop. IN pode ter qualquer forma de onda, digamos 110. Se após assentado o sinal no contador desabilitarmos IN (nível 1 na AND), os pulsos de clock farão que os bits circulem em looping da esquerda para a direita, conforme tabela abaixo e figura 6-10.

C L O C K	Q_0	Q_1	Q_2
0	0	0	0
1	0	0	0
2	1	0	0
3	1	1	0
(COMEÇA CIRCULAÇÃO)			
4	0	1	1
5	1	0	1
6	1	1	0
7	0	1	1
8	1	0	1
9	1	1	0
10	0	1	1

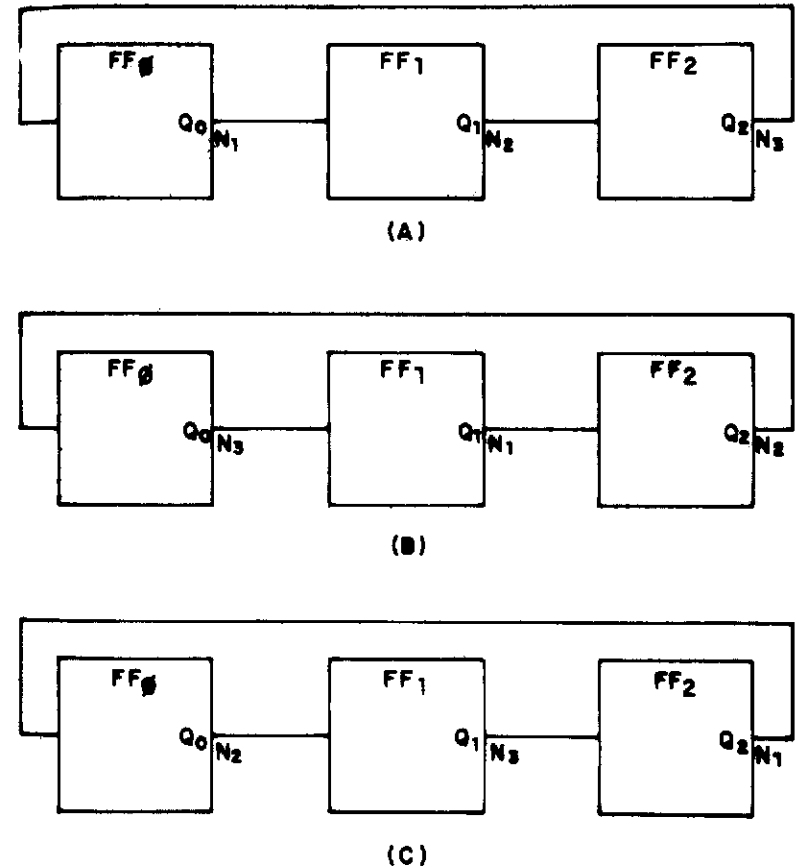


figura 6-10 - Circulação de forma de onda no contador circular switch-tail

Observe que os bits podem ser retirados no último output (Q_2) na mesma ordem em que foram armazenados no contador. Ao final de uma circulação completa todos já terão sido enviados serialmente para o exterior e ainda estarão gravados no contador, disponíveis para outras operações.

Se quisermos alterar um (ou vários) bits o nível de IN é levado a 1 ou 0, conforme necessário, o que permite atualizar os dados no contador.

(5) GERADOR DE SEQUÊNCIA

Diversos equipamentos e processos exigem uma forma de onda ou sequência de bits pré-determinadas. Por exemplo, o código 1100, 1010, 1111, 1100, 1010, 1111, 1100, ..., que se repete periodicamente após três números.

A sequência desejada pode ser obtida num GERADOR DE SEQUÊNCIA, contado com os outputs

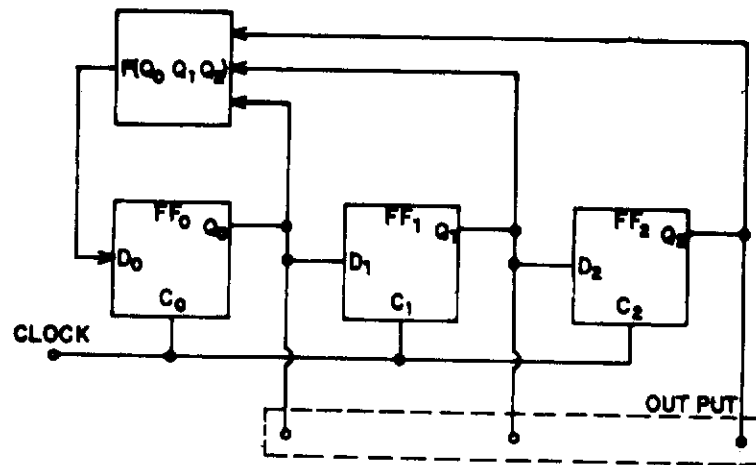


figura 6-11 - Gerador de frequência

combinados num circuito lógico (figura 6-11). O projeto do circuito capaz de gerar a sequência exige conhecimentos aprofundados de Lógica Matemática, fugindo ao escopo deste livro. Registre-se, porém, que qualquer sequência de bits, repetindo-se periodicamente, pode ser gerada por uma combinação adequada de portas lógicas.

CAPÍTULO 7 – CONVERSÃO ANALÓGICA-DIGITAL

Com raríssimas exceções processos e valores do dia-a-dia são analógicos. A importância do dado digital é sua precisão e facilidade de manipulação. Surge assim a necessidade de converter valores analógicos em digitais e vice-versa. Muitas vezes o correrá, como em telefonia moderna, a conversão de um dado analógico em digital, seu processamento e posterior reconversão do resultado digital em analógico.

Neste capítulo estudaremos as várias técnicas usadas nessas conversões, iniciando por circuitos de chaveamento eletrônico ("gates"), que são comuns e necessárias a todas. Como conversão digital para analógica (D/A) veremos os processos de resistor ponderado e o ladder. Em conversão analógica para digital (A/D) apresentaremos o comparador paralelo e o registro de aproximações sucessivas.

(1) CHAVEAMENTOS DIGITAIS

Circuitos operando com sinais analógicos são manipulados por chaves eletrônicas para fins determinados. Como a mais precisa chave eletrônica é um dispositivo digital surgem algumas técnicas mistas analógicas-digitais.

MULTIPLEXADOR (geralmente abreviado MUX) é um sistema que permite vários inputs compartilharem o mesmo canal, cada um transmitindo seu sinal durante uma parcela de tempo e ficando desabilitado nas parcelas dedicadas a outros.

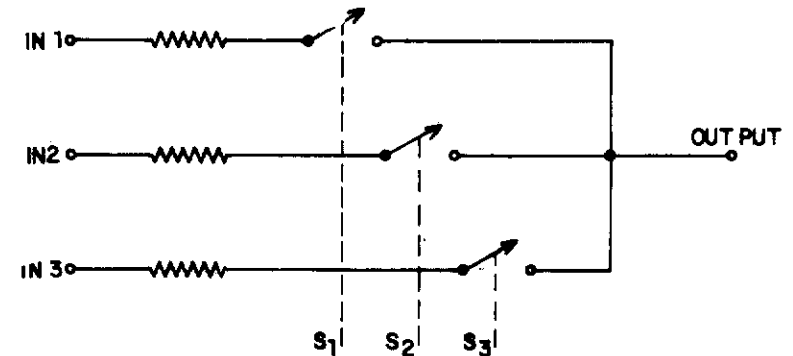


FIG. 7-1A

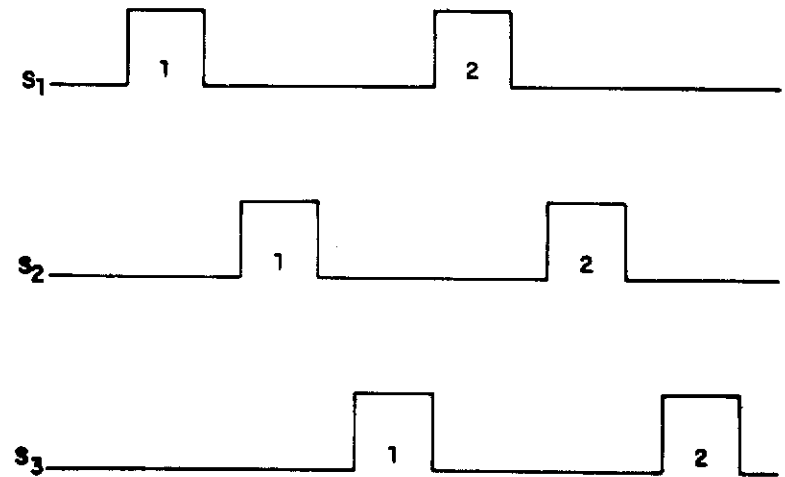


FIG. 7-1B

figura 7-1 - Multiplexador:(A) circuito (B) chaveamento

A figura 7-1 mostra um circuito multiplexador simples com três inputs. As chaves que habilitam os canais são operados por pulsos que ocorrem em tempos diferentes (figura 7-1B). As formas de ondas requeridas podem ser geradas num contador circular switch-tail como da figura 7-2.

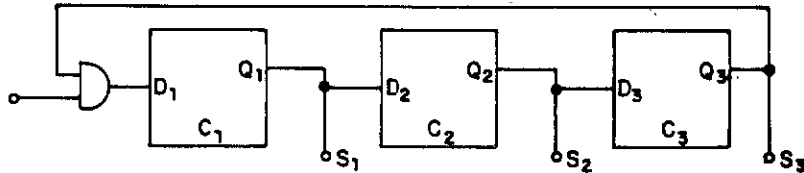


figura 7-2 - Contador circular gerador de formas de onda para chaveamento do MUX

SAMPLE AND HOLD ("amostra e conserva")

- S/H - consiste num circuito que faz a amostragem de um sinal na sua entrada por curto tempo e o mantém na saída por longo tempo (figura 7-3A).

Quando a chave digital S fecha o circuito o sinal da amostra A é rapidamente carregado através da pequena resistência R_i no capacitor hold. Ao terminar o período de habilitação a carga no capacitor descarrega lentamente através da grande resistência R_o . A saída em H é uma reconstituição aproximada do sinal na entrada (figura 7-3B).

No CHOPPER ("fatiador") um sinal de baixa frequência é modulado antes de ser processado, tornando-se uma sequência de pulsos de níveis variados, cujas amplitudes o reconstituem (figuras 7-4, 7-5 A e B)

O sinal dá entrada num amplificador. Entretanto, pulsos digitais na frequência da modulação atuando na chave S o colocam em curto com a terra, transformando-o numa sequência de pulsos cujos picos são proporcionais ao sinal naquele instante.

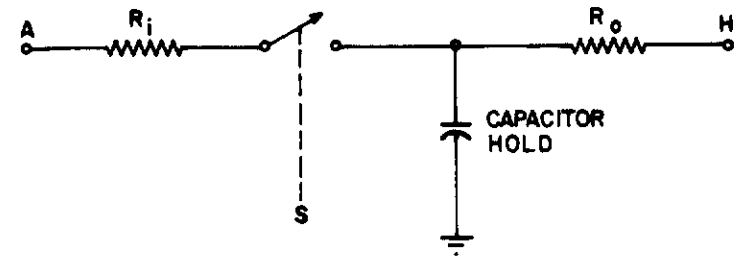


FIG. 7-3A

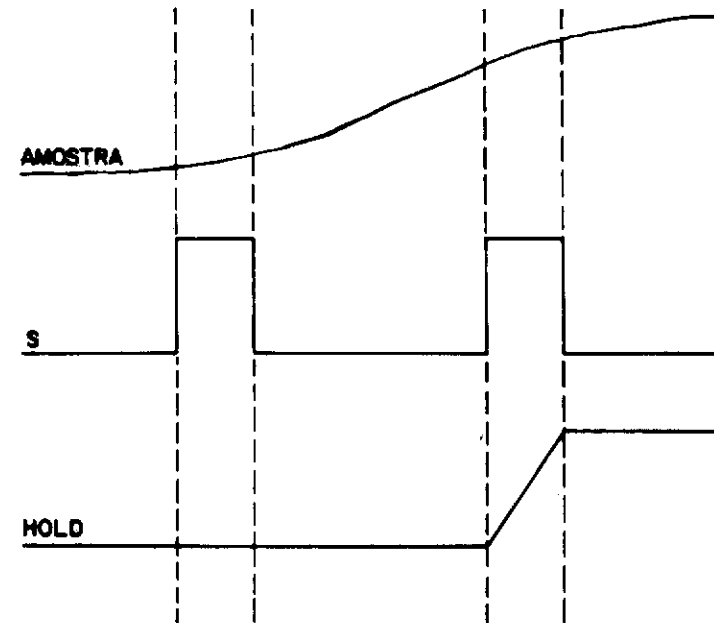


FIG. 7-3B

figura 7-3 - Sample and hold-S/H (A) circuito (B) formas de onda

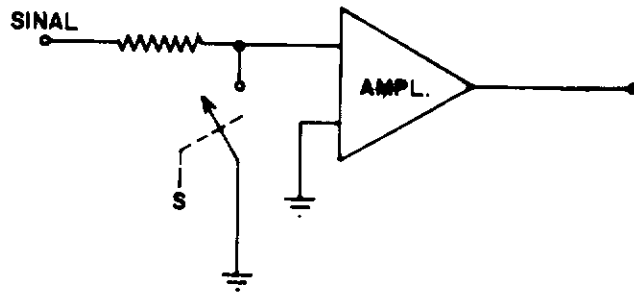


figura 7-4 - Chopper



FIG. 7-5A



FIG. 7-5B

figura 7-5 - Formas de onda no chopper (A) entrada analógica (B) saída

As chaves eletrônicas citadas acima ("gates") geralmente são transistores. A característica de $-0,5\text{m Volts}$ entre coletor e emissor (V_{CE}) durante a saturação do transistor bipolar no modo inverso é usada para transformá-lo numa chave de rápida comutação.

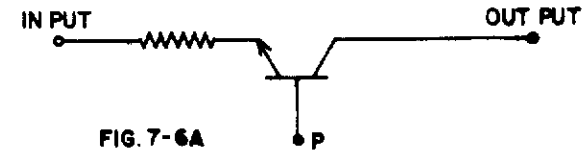


FIG. 7-6A

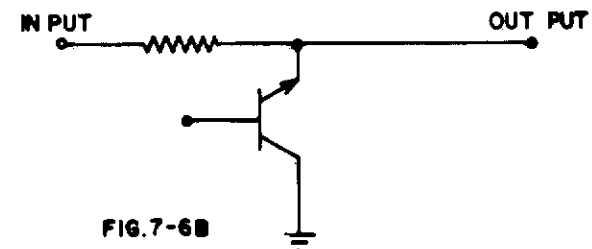


FIG. 7-6B

figura 7-6 - Transistor bipolar em modo inverso como gate (A) série (B) paralelo

As duas configurações da figura 7-6 são possíveis. Em (A) quando o transistor é levado à saturação, pela aplicação do sinal de controle na base a voltagem existente no INPUT (emissor) é transferida para o OUTPUT (coletor), com uma ligeira diferença de $-0,5\text{ m Volts}$, como uma chave fechada coloca em curto dois pontos; cortando o transistor a voltagem do INPUT é isolada do OUTPUT, semelhante a chave aberta. Em (B) com o transistor saturado a voltagem no OUTPUT é $-0,5\text{ m Volts}$, independente do INPUT (chave aberta); cortando o transistor a voltagem do INPUT é transferida para o OUTPUT (chave fechada). Nos dois casos lembre-se que o transistor funciona no modo inverso.

O FET também é usado como gate, apresentando várias vantagens em relação ao transistor bipolar nesta função. As figuras 7-7A e 7-7B correspondem ao mesmo processo visto para transistores bipolares.

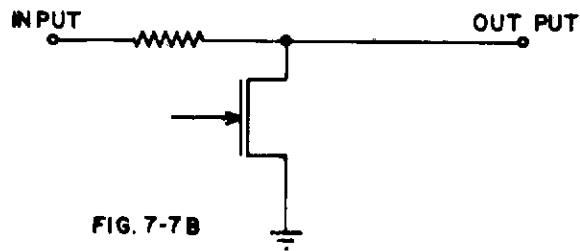
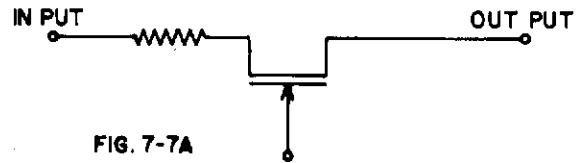


figura 7-7 - FET como gate (A) série (B) paralelo

(2) CONVERSÃO DIGITAL PARA ANALÓGICO - D/A

RESISTOR PONDERADO D/A é a mais fácil técnica de conversão D/A (figura 7-8). O OUTPUT é conectado a um sistema de chaveamento através de resistores cujos valores formam uma progressão geométrica de razão $1/2$, sendo o LSB = R e as seguintes divisões sucessivas por múltiplos de 2 até chegar ao MSB. As chaves podem ser manuais ou eletrônicas (gates), conforme a aplicação, fechando contacto dos resistores com +V (nível lógico 1) ou aterrando-os (nível 0).

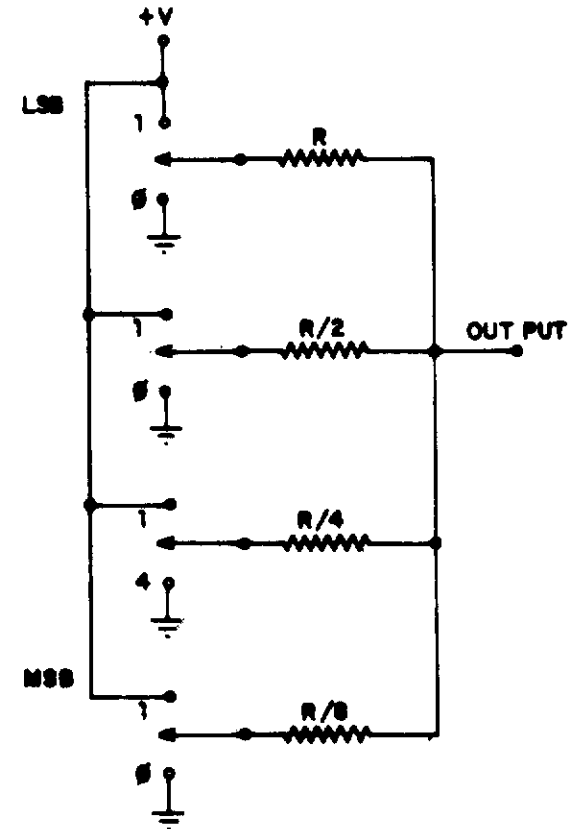


figura 7-8 - Resistor ponderado

Pela Lei de Ohm quanto menor a resistência maior será a corrente que o atravessa, para uma dada tensão. Portanto, o LSB irá gerar o menor sinal no OUTPUT devido ao grande valor do seu resistor, cabendo ao MSB fornecer a maior corrente. O sinal no OUTPUT será exatamente uma resultante desses valores ponderados. Veja que o número pode ter tantos bits quanto for preciso.

Um aperfeiçoamento do sistema de resistor ponderado consiste em acrescentar um amplificador operacional, como na figura 7-9. Como o output é dado por

$$V_o = - \frac{R_F}{R_i} V,$$

onde R_i representa a resistência ligada ao não inversor (e resistências aterradas são praticamente desprezadas), basta chavear os resistores ponderados para se obter output analógico correspondente ao valor digital no input.

LADDER é outro sistema de conversão D/A, também baseado na Lei de Ohm. Somente dois valores de resistores são necessários (R e $2R$), o que torna mais fácil a fabricação em circuito integrado. O chaveamento conecta o resistor de $2R$ ao OUTPUT através de tantos resistores R quanto for a posição do bit em referência (figura 7-10).

Ao sinal injetado no LSB cabe atravessar o maior número de resistores, sendo bastante atenuado até chegar ao OUTPUT. O MSB terá a menor atenuação, ligando-se ao OUTPUT através de apenas um resistor $2R$.

O OUTPUT é centralizado em torno de uma voltagem por meio da VOLTAGEM OFFSET (V_{OFF}), que substitui o terra nos exemplos anteriores ou é injetada no não inversor do amplificador operacional (figura 7-11), nos dois casos através de um potenciôme-

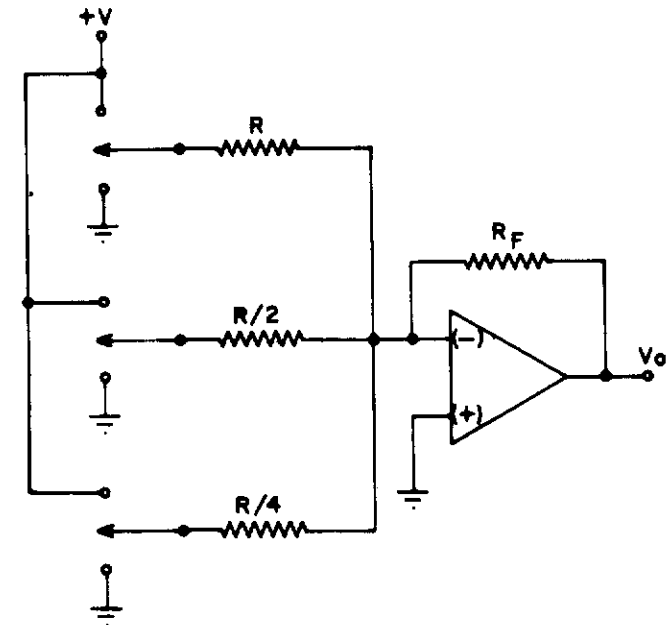


figura 7-9 - Resistor ponderado com amplificador operacional.

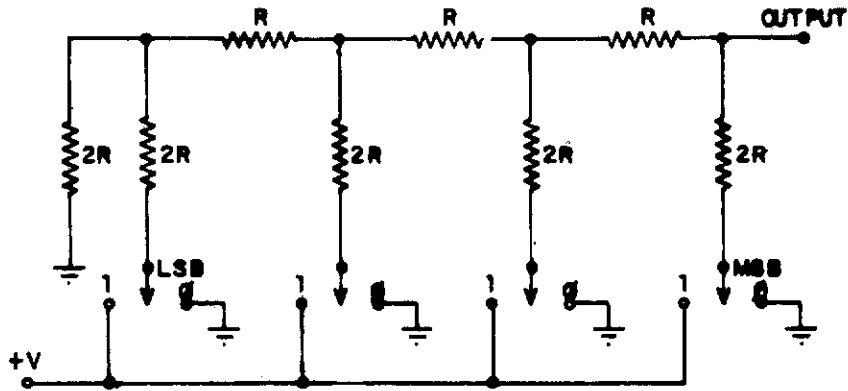


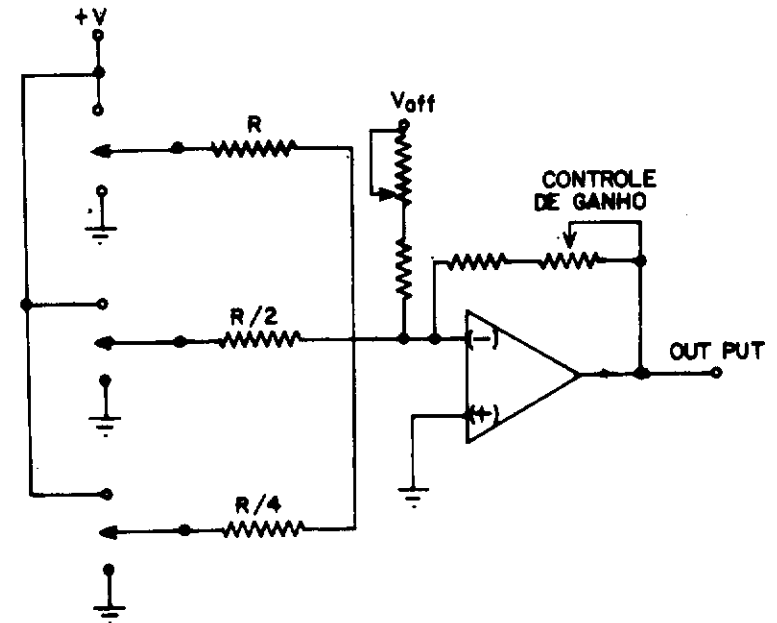
figura 7-10 - Ladder

tro de calibragem. CONTROLE DE GANHO é outro potenciômetro ajustado para manter o OUTPUT do amplificador operacional dentro da escala de ganho desejada (V_{OFF} geralmente é manualmente controlada pelo usuário enquanto o controle de ganho é feito por seleção de um resistor de valor apropriado).

Há um grande número de CIs que executam todas as tarefas D/A, bastando completar as ligações externas com valores adequados. Um dos mais famosos é o CI 1408 (figura 7-12), que converte 8 bits em voltagem analógica, dispondo de terminais para calibração de V_{OFF} .

(3) CONVERSÃO ANALÓGICA PARA DIGITAL - A/D.

A CONVERSÃO ANALÓGICA PARA DIGITAL A/D é bem mais complexa do que o inverso, visto ante

figura 7-11 - V_{OFF} e controle de ganho. Voltagem de offset em conversões D/A.

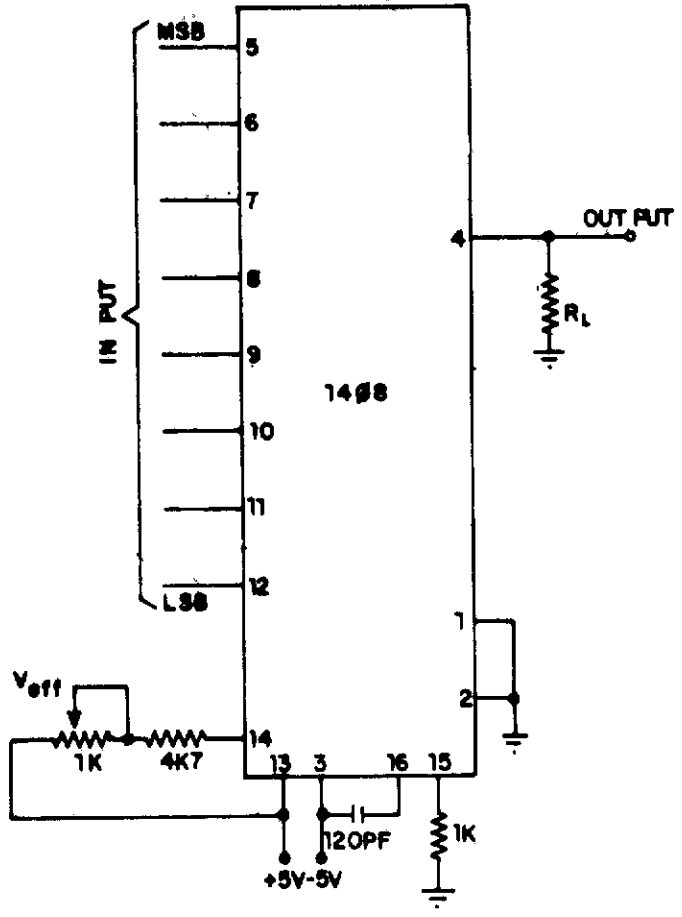


figura 7-12 - CI 1408 conversor D/A

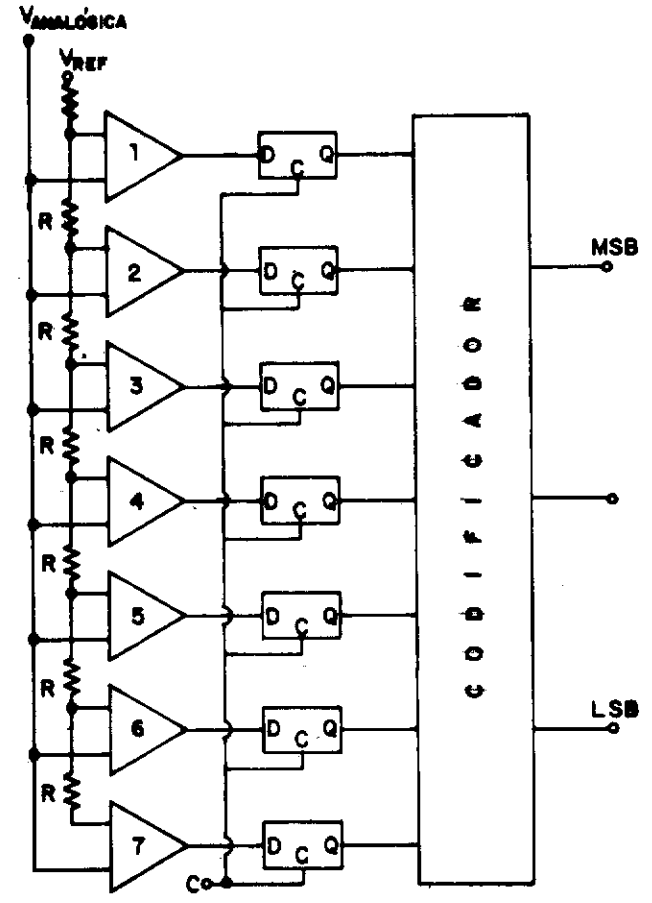


figura 7-13 - Comparador paralelo

riormente. Existem ainda um grande número de circuitos possíveis, com um sem-número de variantes. Abordaremos apenas os dois mais usados.

No COMPARADOR PARALELO a voltagem analógica que se quer digitalizar é comparada com uma parte da voltagem de referência V_{REF} . A posição na série de resistores em que é retirada a parte da voltagem de referência determina o peso ponderado do resultado naquele ponto (os resistores funcionam como divisores de voltagem). A saída nos comparadores (V_{ALTA} ou V_{BAIXA} , ou seja, 1 ou 0) dá entrada nos flip-flops que os armazena e os encaminha ao codificador, onde finalmente aparecem como bits.

Considere a figura 7-13, onde são usados 7 compensadores. Isso permite comparar a voltagem analógica com 8 valores digitais (o 8º é o máximo), que são codificados em 3 bits. A cada clock ocorrerá uma conversão, de maneira que uma voltagem analógica variável será amostrada e convertida em bits periodicamente.

No processo de APROXIMAÇÕES SUCESSIVAS (figura 7-14) um dispositivo RAS (registro de aproximações sucessivas) emite um número binário que é transformado por um D/A em voltagem analógica, a qual é comparada com a voltagem que se quer digitalizar. Conforme a comparação resulta um nível 1 ou 0 que realimenta o RAS, fazendo-o reajustar o número binário até que se obtenha o valor analógico exato.

Na primeira tentativa do RAS o MSB é elevado a 1 e todo os outros permanecem em 0, perfazendo o binário 10000.... Se a saída no D/A for maior que a voltagem de amostra ocorre um nível 0 no comparador e na entrada do RAS, que baixa o MSB para 0 e leva o próximo bit para 1, resultando o binário 01000..... Entretanto, se o D/A tem saída menor que a amostra o comparador é levado a 1 e o RAS mantém o MSB em nível alto e eleva também o pró

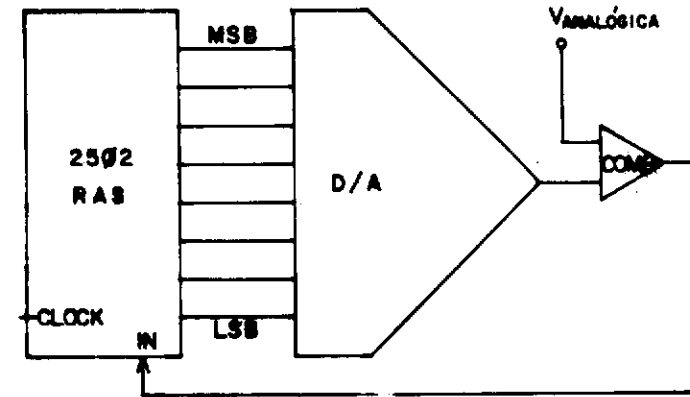


figura 7-14 - Conversão D/A por aproximações sucessivas.

ximo bit, apresentando na sua saída 11000.... O processo prossegue, testando-se bit por bit a cada clock no RAS na direção do MSB para o LSB. O número binário final registrado no RAS é o valor digital correspondente à amostra analógica.

Um RAS bastante usado é o CI 2502 (o D/A poderia ser o CI 1408 visto na seção anterior deste capítulo). Em sistemas microprocessados o microprocessador devidamente programado pode substituir o RAS, porém sendo mais lento que ele na operação.

CAPÍTULO 8 – MEMÓRIAS SEMICONDUTORAS

Memórias semicondutoras são dispositivos que armazenam dados (bits), dividindo-se em três ramos: memória de acesso sequencial, RAM e ROM.

Neste capítulo apresentaremos resumidamente as características e princípios de operação de cada um dos três tipos de memórias semicondutoras.

(1) MEMÓRIA DE ACESSO SEQUENCIAL

A MEMÓRIA DE ACESSO SEQUENCIAL caracteriza-se pela apresentação sequencial (um após outro) dos bits armazenados, em posição correspondente à ordem de armazenamento (um gravador, por exemplo, também é uma memória deste tipo: os dados são apresentados, um após outro em sequência, conforme a fita passa diante do cabeçote do play back).

Implementa-se memória de acesso sequencial semicondutora com registros de deslocamento (figura 8-1). A "palavra" a ser memorizada terá tantos bits quantos registros forem utilizados (no caso, 3 bits). A cada clock ocorre um deslocamento, aparecendo na saída um novo conjunto de bits que formam uma nova palavra. Por exemplo, no 8º clock aparecerá a 8ª palavra de 3 bits. Observe que o bit na saída é reintroduzido no registro, circulando da esquerda para a direita. Quando WRITE ENABLE ("habilita escrita") está em nível 0 as portas AND que entram na parte inferior das OR são desabilitadas, impedindo a entrada de dados d_0 , d_1 , d_2 . e habilitando as AND

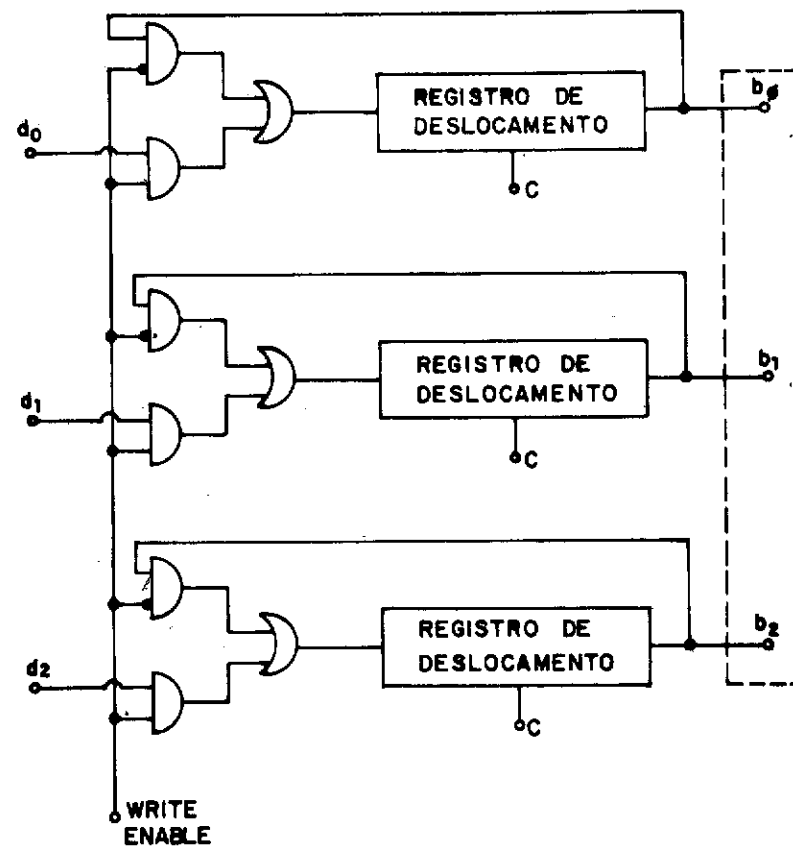


figura 8-1 - Memória de acesso sequencial

superiores para circulação de bits já armazenados. Com WRITE ENABLE = 1 habilita-se as AND inferiores para dados a serem armazenados nos registros, enquanto é cortada a circulação dos bits antigos, que são substituídos pelos novos.

O exemplo acima usa o sistema FIFO ("first in first out" - primeiro a entrar, primeiro a sair). Uma memória de acesso sequencial tipo FILO ("first in last out" - primeiro a entrar, último a sair) é visto na figura 8-2.

Os dados d_0 , d_1 , d_2 são introduzidos nos registros, deslocando-se (para cima, na figura) a cada clock. A leitura é feita nas saídas Q_0 , Q_1 , Q_2 dos mesmos flip-flops de entrada, de modo que a última palavra escrita está na saída de leitura, enquanto a primeira encontra-se nos flips-flops mais afastados da entrada (na parte superior da figura).

A leitura de uma palavra exige a emissão de pulsos de clock até que ela posiciona-se na saída, o que torna o esquema de acesso sequencial de morado (correspondendo a enrolar a fita do gravador até que o dado desejado posicione-se em frente ao cabeçote).

(2) RAM

RAM ("random access memory" - memória de acesso randômico) é um dispositivo que armazena bits mantendo-os disponíveis para leitura, sendo o acesso para escrita ou leitura independente da posição do bit.

Os dados são armazenados na RAM por posições disponíveis - ENDEREÇOS - indo de \emptyset até o maior endereço. A habilitação de um determinado endereço para leitura ("read") ou escrita ("write") é comunicada à RAM através de sinais digitais (1 ou \emptyset)

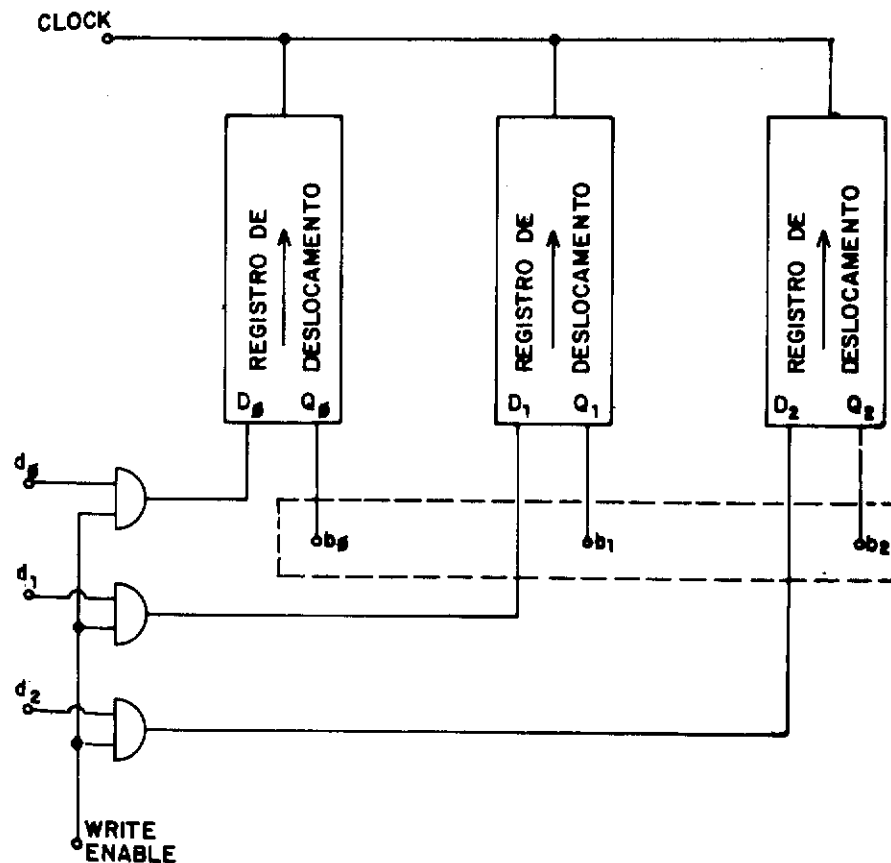


figura 8-2 - Memória de acesso sequencial FILO

nos seus terminais, Assim, na figura 8-3 fazendo $A_7=0$, $A_6=0$, $A_5=1$, $A_4=0$, $A_3=0$, $A_2=0$, $A_1=0$, $A_0=1$, ou mais simplesmente 00100001 $B = 33$, este endereço estará disponível. Se o terminal W/\bar{R} estiver em nível 0 será habilitada a leitura (R, read) e o dado anteriormente gravado naquele endereço aparecerá no pino DADO (como nível 1 ou 0); se W/\bar{R} for levado a 1 habilitar-se-á a escrita (W, write) e o nível 1 ou 0 apresentado ao pino DADO ficará gravado no endereço que estiver presente nos terminais A_0 a A_7 .

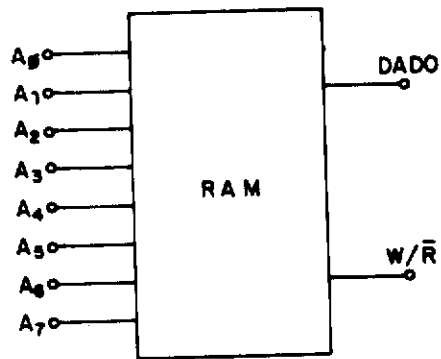


figura 8-3 - RAM

O número de posições disponíveis para escrita e leitura na RAM - sua capacidade - é dada pela fórmula

$$\text{CAPACIDADE} = 2^N,$$

onde N é o número de terminais de endereço (A_0 até A_{N-1}). No exemplo anterior

$$\text{CAPACIDADE} = 2^8 = 256.$$

A quantidade de bits por endereço - "tamanho da palavra" - não é padronizada, sendo comum 1, 2, 4 e 8. Ainda no exemplo anterior mostramos uma RAM de 256 endereços, cada um com um único bit, o que é representado por "256 x 1". Na figura 8-4, temos um exemplo de 256 endereços por 8 bits cada, ou seja, 256 x 8. Quando o endereço é assentado os 8 bits de dados tornam-se disponíveis simultaneamente, para escrita ou leitura.

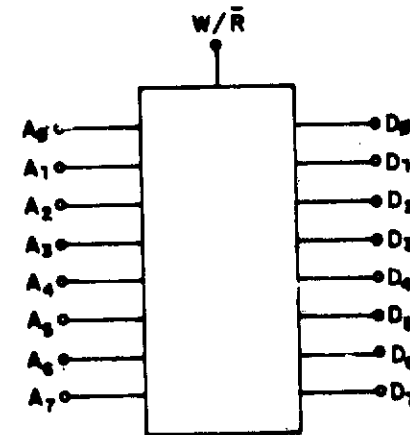


figura 8-4 - RAM 256 x 8

Como 8 bits = 1 byte, a capacidade 256x8 bits também pode ser escrita "256 bytes". Está se tornando comum abreviar bit por "b" e byte por "B", do que resulta

$$\begin{aligned} 256 \text{ bits} &= 256 \text{ b} \\ 256 \text{ bytes} &= 256 \text{ B} \\ 256 \text{ B} &= 256 \times 8 \text{ bits} \end{aligned}$$

Lembre-se ainda que em Eletrônica Digital $2^{10} = 1.024 = K$ ("Kilo"), daí surgindo a nota-

ção bastante popular Kilobyte = KB = 1024 bytes. Por exemplo, 64 KB = 64 x 1024 bytes = 65.536 bytes.

CI's RAM com poucos bits por endereço podem ser ligados em cascata, formando uma memória com tamanho de palavra maior. Bastante comum com microprocessadores é a cascata formando 1 byte de dados (figura 8-5). Quando um endereço é emitido nas linhas A_0 até A_7 ele dará entrada em todos os CI's ao mesmo tempo e cada um apresentará nos pinos de dados sua contribuição à formação do byte.

Além de terminais de dados e endereços encontramos diversos outros pinos de controle. WRITE-W e READ-R são indispensáveis, aparecendo isolados ou conjugados (W/\bar{R}) um como complemento do outro. OE ("output enable", habilita saída) e DS ("device select", seleção de dispositivo) também são comuns, servindo para habilitar e selecionar o CI. Em todos os casos é largamente utilizado o nível \emptyset para ativação o que é simbolizado por uma barra acima do símbolo ou uma bolinha no terminal.

RAM ESTÁTICA (SRAM - "static RAM") e RAM DINÂMICA (DRAM - "dynamic RAM") é uma importante classificação, referindo-se ao tempo que a RAM é capaz de manter os dados gravados. Na RAM estática os bits escritos permanecem gravados enquanto houver ten são alimentando-a. Na RAM dinâmica a capacidade de conservação do bit escrito é de apenas alguns milisegundos, sendo necessário um circuito controlado por sinal externo ("refresh", refrescamento-ou restauração) para manter o nível do dado. Nos dois casos os dados são perdidos quando ocorre o corte na tensão de alimentação.

Todas as tecnologias de semicondutores são empregadas na fabricação de CI's RAM. As bipolares são mais rápidas, porém mais caras. As RAM-MOS, especialmente às NMOS, são bem mais baratas e com maior densidade (mais capacidade de armazenamento por CI).

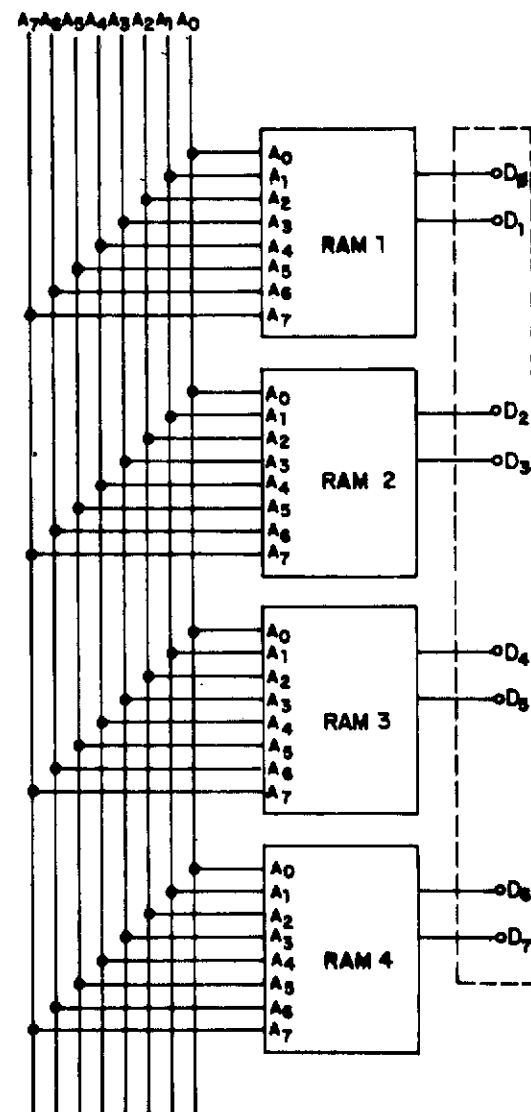


figura 8-5 - Cascata de CI's RAM

(3) ROM

A ROM ("read only memory" - memória de apenas leitura) é um tipo especial de memória de acesso randômico. Sua característica principal - e diferença da RAM - é a conservação dos bits gravados mesmo quando a tensão é desligada.

Ao contrário da RAM onde o usuário pode gravar (escrita) a qualquer momento, inclusive em substituição a um dado anteriormente escrito no endereço, na ROM é feita uma gravação definitiva e indelével na fábrica.

Para evitar o rigor da ROM foi criada a PROM ("programmable ROM" - ROM programável), que é programável pelo próprio usuário. O processo de programação depende da marca usada, mas em geral consiste em fornecer voltagem relativamente alta (20 Volts) a um terminal especial; os níveis presentes nos pinos de dados serão gravados no endereço que estiver assentado nos respectivos pinos.

A EPROM ("erasable PROM" - PROM apagável) é uma ROM que além de programável também pode ter os dados apagados. Entretanto o processo de apagamento é bem mais complexo que na RAM, geralmente usando-se uma luz ultravioleta.

Como na ROM só há leitura não é necessário terminal do tipo READ/WRITE. Basta endereçá-la (pela aplicação de níveis lógicos adequados nos terminais de endereço) e aparecerá nos pinos de dados os bits gravados. É comum, tal como na RAM, encontrar terminais OE e CS.

A ROM é empregada para manter gravados dados de uso constante, como tabelas e programa monitor do aparelho. Dentro de um esquema bem genérico o microprocessador lê instruções e dados gravados pela

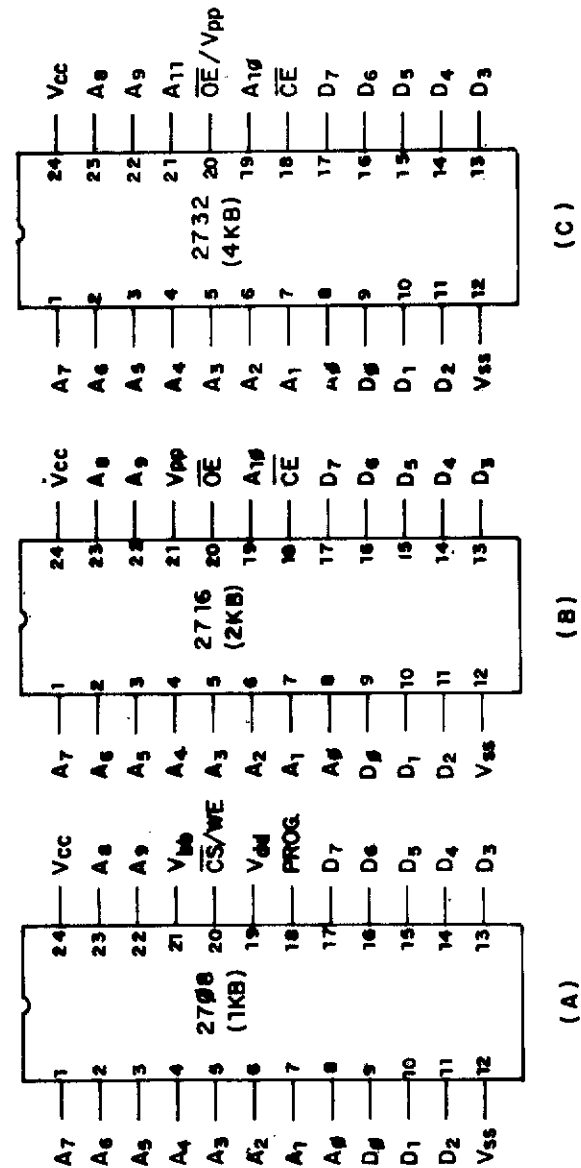


figura 8-6 - EPROM 27xx(a) 2708 - 1KBytes
(b) 2716 - 2KBytes - (c) 2732-4KBytes

fábrica na ROM, transforma-os de acordo com parâmetros transitórios acrescentados pelo usuário na RAM e escreve o resultado das operações nesta última.

A ROM mais popular, encontrada em quase todos microcomputadores, vídeo-games e instrumentação, é a família 27xx da INTEL (figura 8-6). A marcação 'xx' indica a capacidade total em Kilobits da memória:

2708	- 8	Kilobits	=	1 Kilobytes (1 KB)
2716	- 16	Kilobits	=	2 Kilobytes (2 KB)
2732	- 32	Kilobits	=	4 Kilobytes (4 KB)
2764	- 64	Kilobits	=	8 Kilobytes (8 KB)

Na 2708 temos os 9 pinos A₁ - A₉, permitindo o endereçamento de 0 até 512, resultando o total de 1024 (1 Kilo) endereços. Encontramos ainda 8 terminais de dados (D₀ - D₇), alimentação (V_{CC} = + 5 Volts, V_{DD} = - 5 Volts, V_{DD} = - 12 Volts) e terra. A seleção do CI é feita pelo terminal CS com lógica negativa (ativo 0). Na programação dessa EPROM o pino 18 - PROGR. - recebe cerca de 100 pulsos de + 25 Volts, assentados endereços e dados desejados e uma tensão de + 12 Volts no terminal CS/WE ("write enable" habilita escrita).

O desenvolvimento da microeletrônica possibilita dobrar a capacidade da EPROM 2708 e eliminar a voltagem - 5 Volts e - 12 Volts, levando à pinagem da 2716. A programação passou a ser feita com um único pulso (V_{pp}), restando dois terminais de habilitação (OE e CE). Pelo acréscimo do pino A₁₀ o endereçamento aumentou para 2.048 bytes (de 0 a 2047). Observe que os outros pinos são mantidos no mesmo lugar.

Novo progresso se fez dobrando a capacidade da 2716, resultando a 2732 de 4 Kbytes. O pino de programação V_{pp} foi posto junto com OE e no seu lugar acrescentou-se A₁₁, com endereçamento de 0 a 4095 (4.096 bytes).

CAPÍTULO 9 – MICROPROCESSADORES

Os microprocessadores tornaram-se ponto central da Eletrônica, "cérebros" de microcomputadores, vídeo-games, instrumentação eletrônica industrial e atualmente começam seu domínio também sobre televisores e automóveis.

Pelos milagres da Microeletrônica conseguiu-se colocar num único CI (VLSI) quase todos os estágios de um computador inteiro, de maneira simplificada, tornando os poderosos meios da Informática acessíveis às atividades mais variadas.

Apresentaremos uma visão genérica dos microprocessadores, sua evolução, arquitetura e processos de operação. Deve ser notado que grande número de dispositivos e funções anteriormente citados são substituíveis pelo microprocessador e, inversamente, na arquitetura deste encontraremos muito do que vimos até aqui. Ao final do capítulo descreveremos o Z-80, o mais popular microprocessador de 8 bits.

(1) EVOLUÇÃO DOS MICRO PROCESSADORES

A primeira geração de microprocessadores começou com o 4004 da INTEL em 1969, sendo logo em seguida substituído pelo 4040. Esses dois modelos são classificados como "microprocessadores de 4 bits", ou seja, tinham 4 pinos de dados.

A mesma fábrica lançou o 8080, um aperfeiçoamento dos anteriores dispondo de 8 pinos de da

dos - "microprocessador de 8 bits ". A ZILOG projetou o Z-80, o mais popular microprocessador de 8 bits (usado na grande maioria de microcomputadores brasileiros), totalmente compatível com o 8080 e dispondo de maior lista de instruções. Ainda nesta geração encontramos o 6800 da MOTOROLA e o 6502 da MOSTEK.

Os microprocessadores citados até aqui necessitam de dispositivos externos onde esteja gravada a sequência de instruções que irão executar (ROM, PROM, EPROM) e também onde possam armazenar dados temporariamente (RAM). A INTEL lançou o microprocessador 8048 que tem internamente sua própria ROM e RAM, ou seja, é um único CI com quase todos os estágios de um computador interno. Esse microprocessador é o pai da geração denominada "one single chip" ("uma única pastilha").

Da mais recente evolução na área surgiram os "microprocessadores de 16 bits", com 16 pinos de dados (na verdade já estamos entrando na geração de "microprocessadores de 32 bits"), sendo o 68000 da MOTOROLA o mais destacado, além do Z-8000 da INTEL.

(2) ARQUITETURA DO MICROPROCESSADOR

Na sua arquitetura (figura 9-1) os microprocessadores dispõem de registros de uso geral (A - acumulador) e de uso específico (IR - "instruction register" - registro de instruções; PC - "program counter" - contador de programa); no nosso exemplo limitamo-nos aos registros mais indispensáveis, mas geralmente aparecem em número bem maior (registros são cadeias de flip-flops, como estudado no capítulo 6).

No PC fica registrado (em binário) o endereço da instrução a ser executada, gravada na memória. Após a execução de cada uma o conteúdo do PC é incrementado + 1, o que leva a uma execução sequencial do programa gravado. Instruções especiais de "jump" (salto) carregam qualquer endereço desejado no PC, permi-

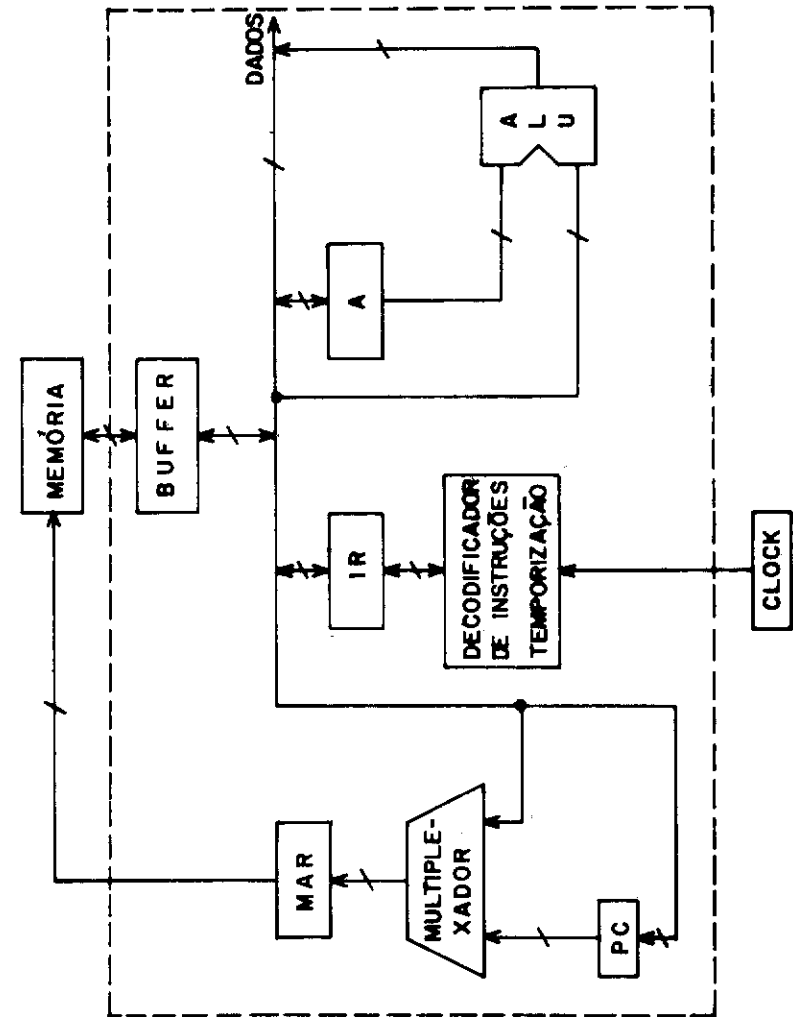


figura 9-1 - Arquitetura de microprocessador

tindo escapar da sequência.

O conteúdo do PC é carregado no MAR ("memory address register" - registro de endereço de memória), que o apresenta como endereço para a memória externa ao microprocessador. Esta coloca na linha de dados, através de um BUFFER, o conteúdo do endereço solido citado pelo PC, que é armazenado no IR ("instruction register" - registro de instruções). O DECODIFICADOR DE INSTRUÇÕES e o controle de TEMPORIZAÇÃO decodificam a instrução IR, para que ela possa ser executada.

O processo acima é uma leitura de memória. Na ESCRITA (WRITE) o PC é desativado pelo MULTIPLEXADOR, que apresenta ao MAR o endereço onde se deseja "escrever" (gravar bits, numa RAM por exemplo).

A ALU ("arithmetic-logic unit" - unidade lógica - aritmética) realiza todas as operações lógicas e aritméticas requeridas pelo sistema. Observe que um dos membros da operação é tomado da linha de dados e o outro do acumulador; o resultado obtido pela ALU é posto na linha de dados e geralmente escrito no acumulador. Isso explica duas características de programas para microprocessadores: (1) carregar o acumulador com os dados que se quer operar e (2) transferir para outro registro o conteúdo do acumulador, deixando-o livre para receber outros resultados.

(3) TEMPORIZAÇÃO DO MICROPROCESSADOR

As operações do microprocessador são realizadas em sincronia com o sinal externo de clock. A cada pulso dele ocorre mais um passo na execução das instruções, classificando-se os tempos do microprocessador, em três categorias (figura 9.2):

- (1) CICLO DE CLOCK propriamente dito (T1, T2, T3, etc);
- (2) ESTADO DE MÁQUINA (M1, M2, M3), composto por 3 a 5 ciclos de clock que indica em qual passo da instrução o microprocessador se encontra. M1 sempre

apresenta o início de busca de instrução gravada na memória para ser executada e os microprocessadores costumam emitir um sinal indicando que estão neste estado (por exemplo, sinal $\overline{M1}$ - pino 27 - no Z-80).

- (3) CICLO DE INSTRUÇÃO, compreendendo de 3 a 5 estados de máquina, que englobam desde a busca de instrução (M1), até sua decodificação e execução (M2, M3).

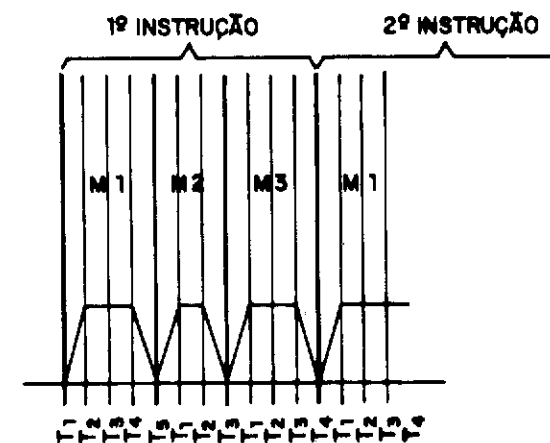


figura 9-2 - Tempos na execução de instruções

O sinal do clock varia de 1 MHz a 10 MHz, dependendo da marca e modelo do microprocessador. Além do mais muitos deles tem um limite máximo para a frequência do clock mas funcionam normalmente abaixo deste valor.

Adotamos uma média de 2 MHz para o clock e também 20 ciclos para a execução de uma instrução (4 estados de máquina x 5 ciclos cada). Disso resulta que o microprocessador é capaz de executar 1000000 instruções por segundo!

(4) BARRAS DE ENDEREÇOS, DADOS E CONTROLES

O microprocessador se comunica com os outros estágios através de trilhas de circuito impresso, por onde a corrente elétrica circula indicando os níveis lógicos. Elas são divididas em 3 grupos, denominadas BARRAS.

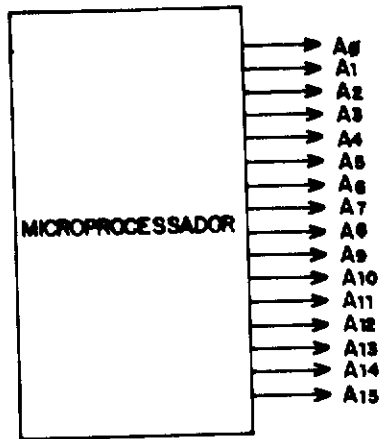


figura 9-3 - Barra de endereços

A BARRA DE ENDEREÇOS (figura 9-3) transporta sinais no sentido do microprocessador para o resto do sistema, na forma de bits compondo um número binário, indicando com qual endereço ele quer se comunicar. Na maioria dos microprocessadores há 16 pinos de endereço, resultando uma barra com 16 trilhas de circuito impresso, o que possibilita o endereçamento de $65.536 = 64 \text{ K}$ posições (o cálculo é feito baseado na fórmula $\text{TOTAL DE POSIÇÕES} = 2^N$, onde N é o número de pinos de endereço).

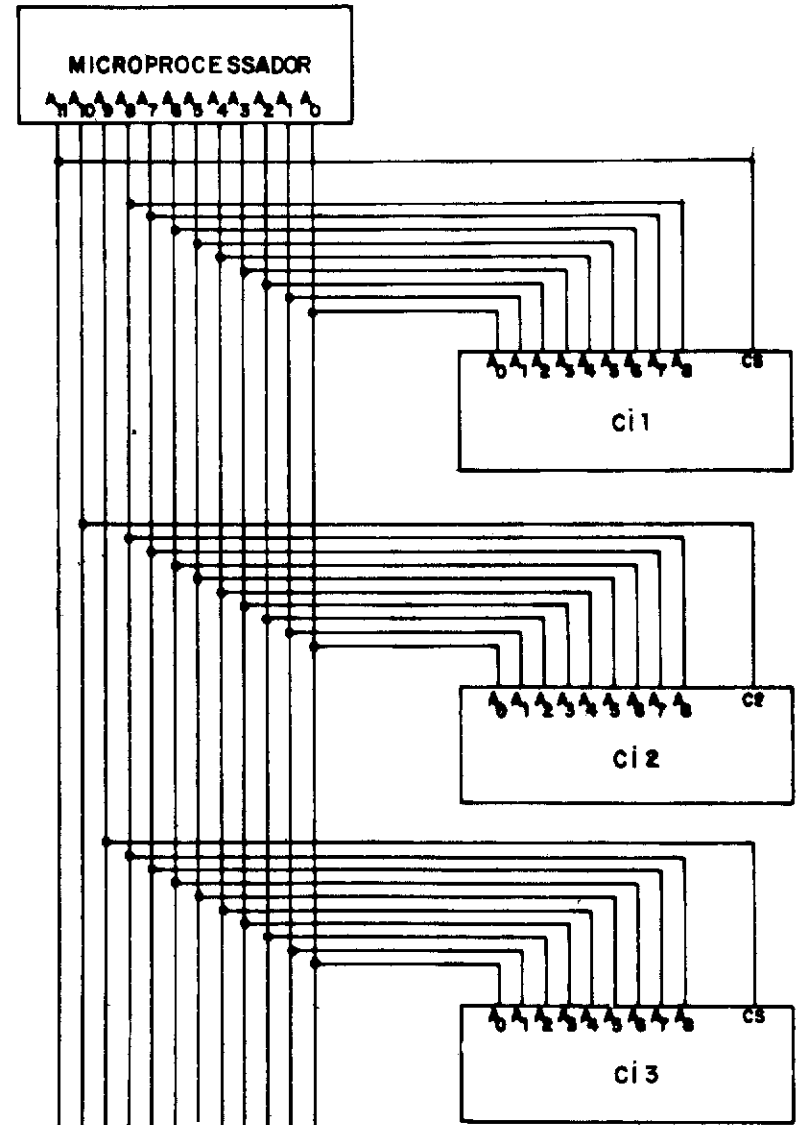


figura 9-4 - Distribuição de endereços por CI

Para diferenciar os vários componentes com os quais o microprocessador se comunica, permitindo que cada um tenha seu próprio endereço, todos eles são conectados aos bits menos significativos da barra de endereços e os bits mais significativos são distribuídos para diferenciação. A figura 9-4 mostra um exemplo típico. Os nove bits menos significativos (A_0 até A_8) são comuns aos três CIs. A_9 é conectado ao pino CS ("chip select", seleção de chip) do CI 3, A_{10} ao CI 2 e A_{11} ao CI 1. Portanto, o endereço existente de A_0 até A_8 só pertencerá ao CI 3 quando A_9 estiver em nível 1. Disso resulta que este CI conterá os endereços $A_9 = 1$, $A_8 = 1/0$, $A_7 = 1/0$, $A_6 = 1/0$, $A_5 = 1/0$, $A_4 = 1/0$, $A_3 = 1/0$, $A_2 = 1/0$, $A_1 = 1/0$, $A_0 = 1/0$, ou seja de $10\ 0000\ 0000\ B$ até $11\ 1111\ 1111\ B$ (em hexadecimal de $A00H$ até $BFFH$), que na numeração decimal corresponde de 512 até 1023).

Num exemplo, o microprocessador apresenta na barra de endereços $A_8 = 0$, $A_7 = 1$, $A_6 = 0$, $A_5 = 0$, $A_4 = 0$, $A_3 = 0$, $A_2 = 0$, $A_1 = 0$, $A_0 = 1$ ($0100000001\ B = 129$). As respectivas posições 129 dos três CIs serão ativadas, mas só será efetivamente habilitada aquela pertencente ao CI que tem o MSB em nível 1; as outras duas posições não se comunicarão com o microprocessador.

A BARRA DE DADOS (figura 9-5) é composta de trilhas de circuito impresso que levam e trazem a corrente elétrica contendo os bits de dados.

Observe que a barra de endereços conduz sinais sempre na direção do microprocessador para o resto do sistema - dizemos que é unidirecional - enquanto na barra de dados o fluxo pode ser tanto saindo quanto chegando ao microprocessador - bidirecional.

A BARRA DE CONTROLES (figura 9-6) é usada para controlar os sinais de outras barras, indicando sentido de fluxo, tempo em que cada sinal deve ser considerado e outros comandos específicos de cada micro -

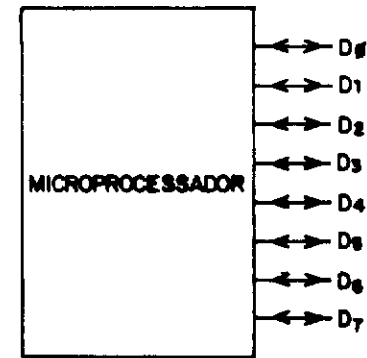


figura 9-5 - Barra de dados

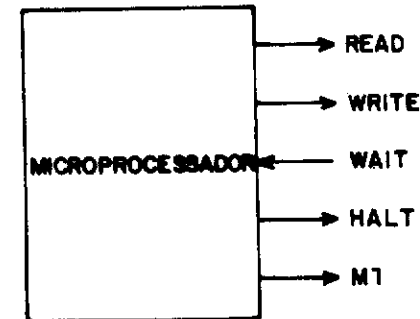


figura 9-6 - Barra de controles

processador. O número de pinos de controle, e correspondente de trilhas de circuito impresso, depende da marca do microprocessador e situa-se na faixa de 3 a 8.

De importância fundamental são os controles READ (leitura) e WRITE (escrita). Quando o microprocessador emite o sinal READ (figura 9-7) significa que ele está lendo os bits da barra de dados originados no endereço indicado pela barra de endereços (o fluxo é no sentido resto do sistema para o microprocessador), ou ordenando à interface que receba os dados dos periféricos (INPUT). A emissão de WRITE pelo

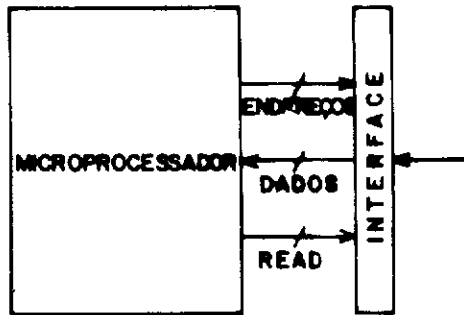


figura 9-7 - Sinalização de leitura/input

microprocessador (figura 9-8) indica que ele está enviando dados para serem gravados na posição indicada pela barra de endereços (fluxo no sentido microprocessador para o resto de sistema), ou ordenando que a interface envie dados para os periféricos.

Além de emitir sinais de controle o microprocessador também pode recebê-los. Um deles - READY

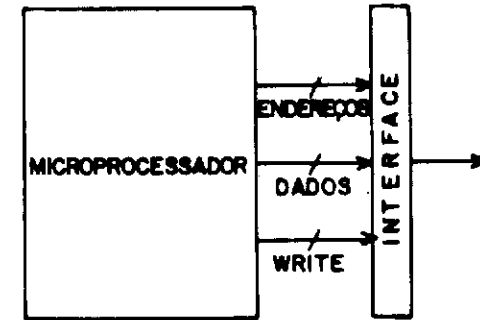


figura 9-8 - Sinalização escrita/output

(preparado, pronto; também pode aparecer como WAIT - espere) é de grande importância em trabalhos de análise e manutenção. Quando esse sinal de controle chega ao microprocessador este pára de executar as instruções (permanece indefinidamente no estado M 1), até que o sinal seja retirado.

É necessário considerar com atenção o nível lógico que ativa o terminal do microprocessador. Assim, se ele tem o pino READ (ativação em nível 0 indicado pela barra) significa que ocorrerá uma leitura (ou input) quando o respectivo pino estiver em nível 0 e NÃO ESTARÁ ocorrendo leitura (ou input) quando o nível for 1. Um caso interessante é a existência do terminal READ/WRITE; o pino em nível 1 indica leitura e em 0 escrita.

(5) MICROCONTROLADORES

Chamamos microcontrolador um microprocessador de estrutura bem simples, capaz de executar apenas uma seqüência fixa de instruções. Atualmente eles estão sendo largamente empregados em eletrodomésticos por exemplo, em máquina de lavar programável.

(6) MICROPROCESSADOR Z-80

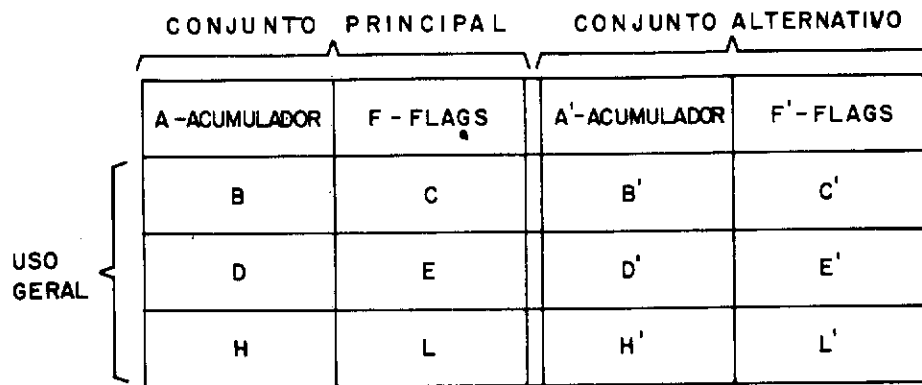
O Z-80 é um microprocessador de 3a. geração, sendo o aperfeiçoamento do 8080, inclusive totalmente compatível com ele (OZ-80 tem todas instruções do 8080 e mais algumas). É o mais popular microprocessador de 8 bits, tendo sido adotado pela grande maioria de microcomputadores brasileiros.

Na sua estrutura interna encontramos 22 registros (figura 9-9), sendo 6 de uso específico e 16 de finalidades gerais. Estes últimos formam dois conjuntos iguais - 8 registros principais e 8 alternativos.

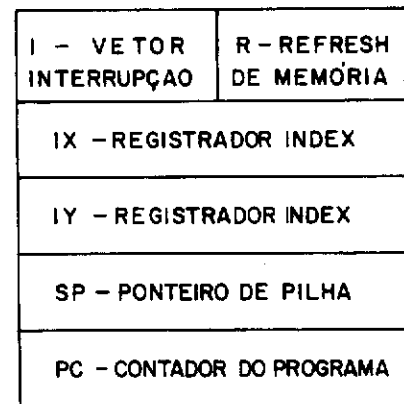
Através de uma instrução do programa seleciona-se um par ou outro, mas não os dois simultaneamente. Os registros de uso geral são de 8 bits, mas B-C, D-E, H-L podem ser manipulados como par de 16 bits. Os registros de uso específico tem 16 bits, com exceção de I e R, de 8 bits cada.

No acumulador (A) são registrados resultados de operações e dados recebidos da memória ou de portas. Os flags F são 6 registros de um só bit cada (mais 2 não usados) que indicam resultados de operações lógicas e aritméticas executadas pelo microprocessador (figura 9-9C):

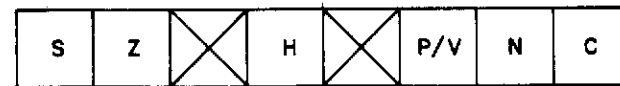
- S : sinal (+ ou -)
- Z : zero ou não zero
- H : existência de carry para o bit 4
- P/N: paridade (par ou ímpar) ou overflow
- N : subtração
- C : carry



(A)



(B)



(C)

figura 9-9 - Estrutura interna do microprocessador Z-80 (a) registros de uso geral (b) registros de uso específico (c) registros de flags.

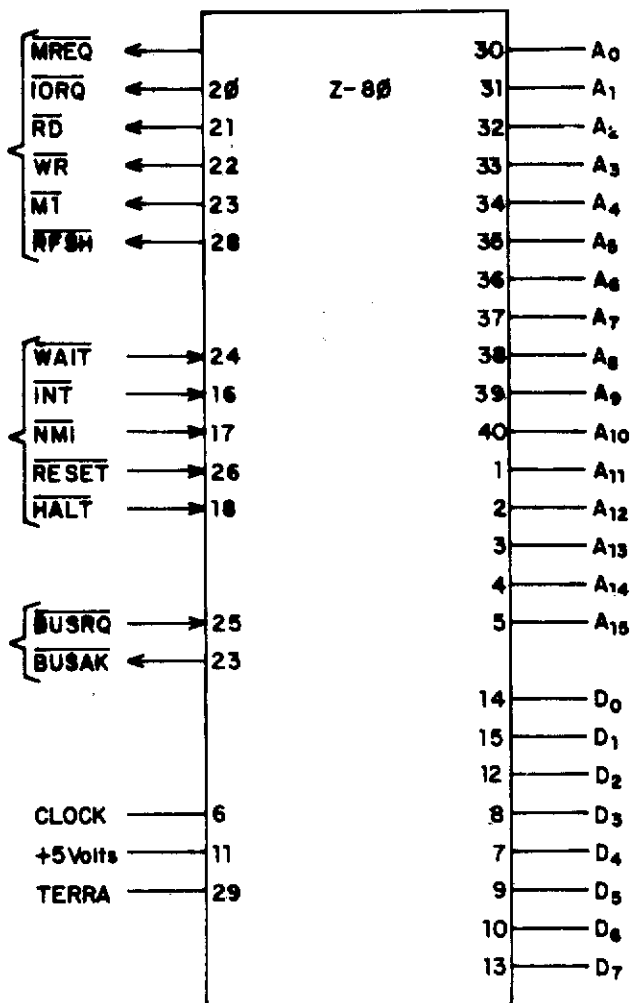


figura 9-10 - Pinagem do microprocessador Z-80

I - vetor de interrupção - contém 8 bits que são juntados a mais 8 bits fornecidos pelo dispositivo interruptor para formar o endereço da rotina de interrupção. R - REFRESH - é usado para endereço de refrescamento automático de memórias dinâmicas. IX e IY são somados ao byte fornecido pela instrução para formar um endereço indexado. SP - stack pointer, ponteiro de pilha - contém o endereço da "pilha" (na memória RAM) onde podem ser guardados temporariamente conteúdos de outros registros. PC-program counter, contador de programa - é um contador cujo conteúdo indica o endereço da próxima instrução a ser executada; após a execução ele é automaticamente incrementado.

O Z-80 funciona sincronizado pelo clock e cada sinal deste é chamado "estado de máquina" - T1, T2, T3. "Ciclo de máquina" M1, M2, M3 é um conjunto de estados de máquina (sinais de clock) usados pelo microprocessador para executar uma instrução. No primeiro ciclo de máquina M1 ele endereça a posição de memória onde vai buscar ("fetch") a instrução, que nos outros ciclos será lida, decodificada e executada.

Os pinos do Z-80 (figura 9-10) tem as seguintes funções:

- A₀ / A₁₅ : 16 terminais de endereço; em tristate quando não usados
- D₀ / D₇ : 8 terminais de dados; em tristate quando não usados
- TERRA
- + 5 VOLTS (regulados)
- CLOCK: frequência dependente do modelo (entre 2 MHz e 4 MHz)
- MREQ : ("memory request", requisição de memória) para operações de memória
- IORQ : ("input output request", requisição de entrada e saída) para operações com portas de entrada e saída

- \overline{RD} : ("read", leitura) leitura de dados da memória pelo microprocessador ou entrada de dados do exterior pelas portas IO
- \overline{WR} : ("write", escrita) escrita de dados na memória pelo microprocessador ou saída de dados para o exterior pelas portas IO
- $\overline{M1}$: ("machine 1") sinal emitido sempre que é executado o 1º ciclo de máquina (portanto, quando é lido o primeiro byte de uma instrução)
- \overline{RFSH} : ("refresh", refrescamento) sinal emitido pelo Z-80 para refrescamento automático de memórias dinâmicas
- \overline{WAIT} : ("wait", aguarde) enquanto estiver ativo provoca estado de espera do microprocessador, que fica sem executar instruções
- \overline{INT} : ("interrupt", interrupção) indica periférico querendo atenção do microprocessador
- \overline{NMI} : ("no maskable interrupt", interrupção não mascarável) interrupção de atendimento obrigatório, para casos de emergência
- \overline{RESET} : ("reset") o contador de programa PC volta a 0000, reiniciando-se o programa
- \overline{HALT} : ("halt", parada) indica que o microprocessador executou uma instrução "halt" e precisa do "INT" para recomeçar
- \overline{BUSRQ} : ("bus request", requisição de barras) periférico quer controlar as barras, em operação DMA ("direct memory access")
- \overline{BUSAK} : ("bus acknowledge", reconhecimento de barras) indica que o Z-80 aceitou o pedido \overline{BUSRQ} .

CAPÍTULO 10 – INSTRUMENTOS PARA TESTE E MANUTENÇÃO DE APARELHOS DIGITAIS

A manutenção e teste de equipamentos digitais tem características bem diferentes do conserto de equipamentos analógicos, exigindo instrumentação própria. Assim o tradicional multímetro tem aplicação muito limitada nesta área, enquanto o osciloscópio raramente é útil.

Neste capítulo apresentaremos os principais instrumentos específicos para teste e manutenção de aparelhos digitais: ponta de prova lógica, tester estático, analisador de níveis lógicos, analisador de assinaturas, frequencímetro, proto-board e wire-wrap. Veremos seus traços principais e operação, destacando vantagens e desvantagens de aplicação.

(1) PONTA DE PROVA LÓGICA

Essa é a ferramenta mais comum e usada em Eletrônica Digital. Sua finalidade é indicar qual o nível lógico existente num determinado ponto, se 1 ou 0 (H ou L). Deve ser notado que só é possível aplicar a ponta de prova lógica em condições estáticas (os níveis lógicos estáveis por tempo relativamente grande). Além dos dois níveis 1 e 0 ela poderá também registrar o tristate, cobrindo assim todos os casos possíveis.

Um esquema comum de ponta de prova é apresentado na figura 10-1. Consiste de um oscilador (multivibrador), três inversores (I_1 , I_2 , I_3), um LED L indicador de nível baixo ("low", 0) e outro LED H

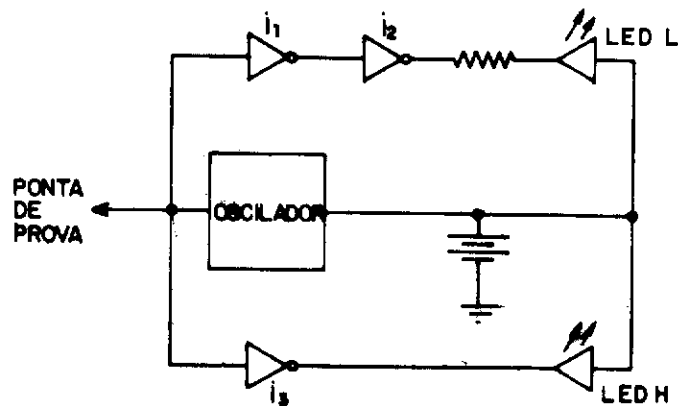


figura 10-1 - Ponta de prova lógica para nível alto ("high", 1).

Com a ponta de prova flutuando (sem receber sinal com nível padrão H ou L) o sinal do oscilador atinge I_1 e I_3 . Na sua fase positiva ele é invertido por I_3 , chegando ao cátodo do LED H como nível baixo, fazendo-o brilhar; ao mesmo tempo ele é invertido e "desinvertido" por I_2 , aparecendo ao cátodo do LED L como nível alto, o que o mantém apagado. Na fase negativa o sinal do oscilador é invertido por I_3 e torna-se nível alto para o cátodo do LED H, que se mantém apagado; nos inversores I_1 e I_2 ele é transformado em níveis positivo e negativo, respectivamente, aparecendo no cátodo do LED L como nível baixo, provocando condução elétrica e o brilho. Assim, os dois LEDs brilham alternadamente com a mesma frequência do oscilador.

Quando a ponta de prova é encostada num ponto com nível lógico baixo (L) o sinal do oscilador é cancelado. I_3 inverte o sinal da ponta de prova e provoca apagamento do LED H. I_1 e I_2 , invertendo e desinvertendo o sinal da ponta de prova fazem com que o LED L brilhe (sozinho e ininterruptamente), indicando nível baixo L do ponto em teste.

Se tivermos nível alto H na ponta de prova o oscilador também será cancelado, I_1 e I_2 conservarão o LED L apagado e I_3 inverterá o sinal, fazendo o LED H brilhar (sozinho e ininterruptamente), indicando nível alto H para o ponto em teste.

(2) TESTER ESTÁTICO (COMPUTESTER)

TESTER ESTÁTICO é um instrumento que injeta dados, endereços e controles na entrada e faz a leitura do nível lógico na saída. Tem a grande vantagem de ser universal, aplicável a qualquer dispositivo digital, além de baixo custo e não exigir conhecimentos teóricos minuciosos por parte do operador.

Sob o nome de COMPUTESTER a PETIT EDITORA comercializa um modelo de tester estático, inclusive por reembolso postal. A aplicação do COMPUTESTER é apresentada em detalhes nos livros "MANUTENÇÃO DE MICROCOMPUTADORES" e "ELETRÔNICA DE VÍDEO-GAMES", do mesmo autor e editora.

Os endereços, dados e controles do COMPUTESTER (figura 10-2) são chaveados no seu painel frontal, aparecendo como níveis lógicos nos seus respectivos terminais, que são conectados ao aparelho em teste. O resultado é avaliado através de uma ponta de prova lógica. Há 16 terminais de endereços, 8 de dados e 3 de controles; a alimentação e o terra são retirados do próprio equipamento sendo testado.

(3) LSA - ANALISADOR DE NÍVEIS LÓGICOS

O LSA ("logic state analyzer") - ANALISADOR DE NÍVEIS LÓGICOS - é um sofisticado e caro ins-

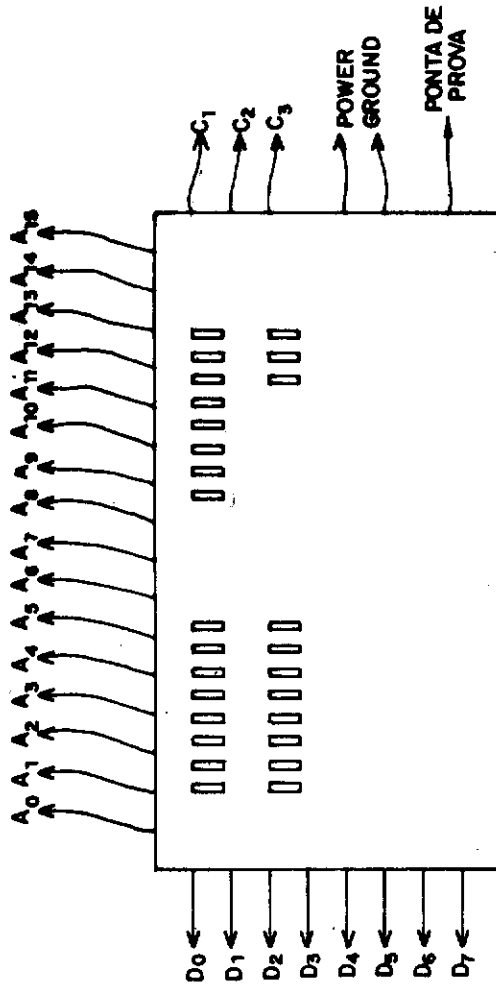


figura 10-2 - Tester estático - Computester

trumento para desenvolvimento e manutenção de dispositivos digitais. Consiste no registro do fluxo de sinais numa parte selecionada da operação. Apesar do alto custo e exigir conhecimentos aprofundados sobre o sistema em teste é bastante eficiente e preciso.

O LSA (figura 10-3) dispõe de um conector com cerca de 28 canais (16 de endereços, 8 de dados e 4 de controles), trigger (conjunto de chaves no painel frontal), qualificador e display. O conector é ligado às barras de endereços, dados e controles do equipamento em teste. A gravação e apresentação no display começa quando aparecer nas barras de endereço o mesmo valor chaveado no trigger. Toda vez que for emitido o sinal ligado ao qualificador é registrado no display o conjunto de níveis lógicos existentes nas barras.

(4) ANALISADOR DE ASSINATURAS

O ANALISADOR DE ASSINATURAS é uma espécie de voltímetro de sinais digitais, registrando um valor relacionado à sequência de bits em cada ponto. É necessário ter o registro de aparelho semelhante aquele em teste, que esteja em perfeito estado de funcionamento. Os dois valores são comparados, verificando-se discrepâncias (tal como medir voltagem ou formas de onda num televisor e comparar com a medição correta do esquema).

No analisador de assinaturas (figura 10-4) encontramos um display e uma ponta de prova. Esta última é conectada com o ponto em teste do aparelho, fornecendo um valor correspondente ao fluxo de bits ali, que é registrado no display. No circuito impresso do equipamento em teste é gravado o valor correto que deveria ser lido pelo analisador de assinaturas ou então a comparação é feita com leitura em equipamento similar em bom funcionamento.

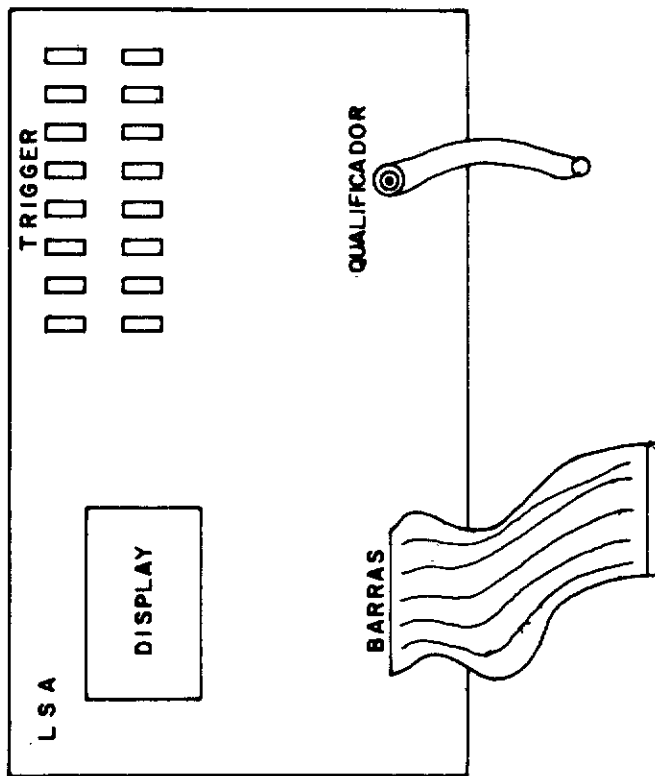


figura 10-3 - LSA - Analisador de níveis lógicos

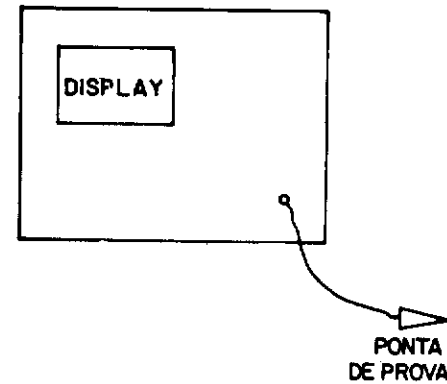


figura 10-4 - Analisador de assinaturas

(5) FREQUENCÍMETRO

O FREQUENCÍMETRO mede a quantidade de pulsos emitidos por um dispositivo, num espaço de tempo relacionado pelo operador.

O ponto do circuito a ser analisado é conectado pela ponta de prova do frequencímetro (figura 10-5) e a base de tempo selecionada através de botão ou chave no painel frontal. A quantidade de pulsos emitidos na base de tempo (frequência) é registrada no display.

(6) PROTO-BOARD

O PROTO-BOARD ("prototype board", prancheta de protótipos) é utilizada tanto em Eletrônica Analógica quanto Digital. Devido à grande padronização

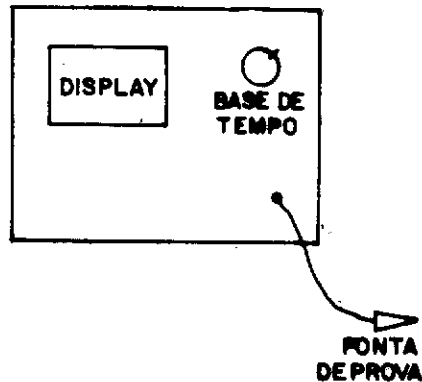


figura 10-5 - Frequencímetro

da parte digital, com uso quase que exclusivo de circuitos integrados, este instrumento adapta-se muito bem a ela, transformando-se num meio rápido e fácil de testar circuitos digitais.

O protoboard consiste numa prancheta cheia de "furinhos" onde são encaixados os CIs. Cada furi - nho é eletricamente conectado a outro que fica vago , disponível para ligações. Estas podem ser feitas por fios ("cabinhos"), que são colocados ou retirados sem necessidade de solda.

(7) WIRE-WRAP

No WIRE-WRAP (arame enrolado) os CIs são encaixados em soquetes especiais de pinos grandes. O operador manipula uma pistola (em indústria pode ser um maquinário de grande porte) que enrola arames fi-

nos, conectando-os eletricamente.

Tanto no Wire-wrap quanto no protoboard não há preocupação inicial quanto ao lay-out ("dese - nho") do circuito impresso. Somente depois de testa - do e aprovado o esquema ele é realizado.

CAPÍTULO 11 – CIRCUITOS ILUSTRATIVOS

A melhor maneira de aprender Eletrônica é seguir circuitos - no papel (esquema) e no aparelho. O leitor terá neste capítulo alguns exemplos de circuitos digitais, tirados das áreas de video-games, micro computadores, instrumentação e televisão.

(1) JOYSTICK DO ODISSEY

Os dois joysticks (dispositivo disponível para participação dos jogadores) do video-game ODISSEY são acoplados ao microprocessador através de dois buffers-tristate; cada buffer e respectivo joystick é selecionado alternadamente por um decodificador, de modo que a cada instante o microprocessador receba os dados de um jogador ou de outro (figura 11-1).

O joystick propriamente dito é composto por um sistema de 5 contatos, com os polos conectados ao buffer "naturalmente" em 5 Volts-nível 1. Quando o jogador aciona um dos contatos a linha correspondente é aterrada, caracterizando-se nível 0 nessa entrada do buffer, que é transmitido para o microprocessador. Observe que temos UP (para cima), DOWN (para baixo), RIGHT (direita), LEFT (esquerda) e ACTION (ação), correspondendo a instrução de jogo para alteração da tela.

O buffer 74 LS 365 (figura 11-2) que faz a interface entre o joystick e o microprocessador mantém seus pinos de saída (13, 11, 9, 5, 7) em tristate enquanto não receber nível 0 nos pinos 1 ($\overline{G1}$) e 15 ($\overline{G2}$). Quando isso ocorre ($\overline{G1} = \overline{G2} = 0$) os níveis existentes nas

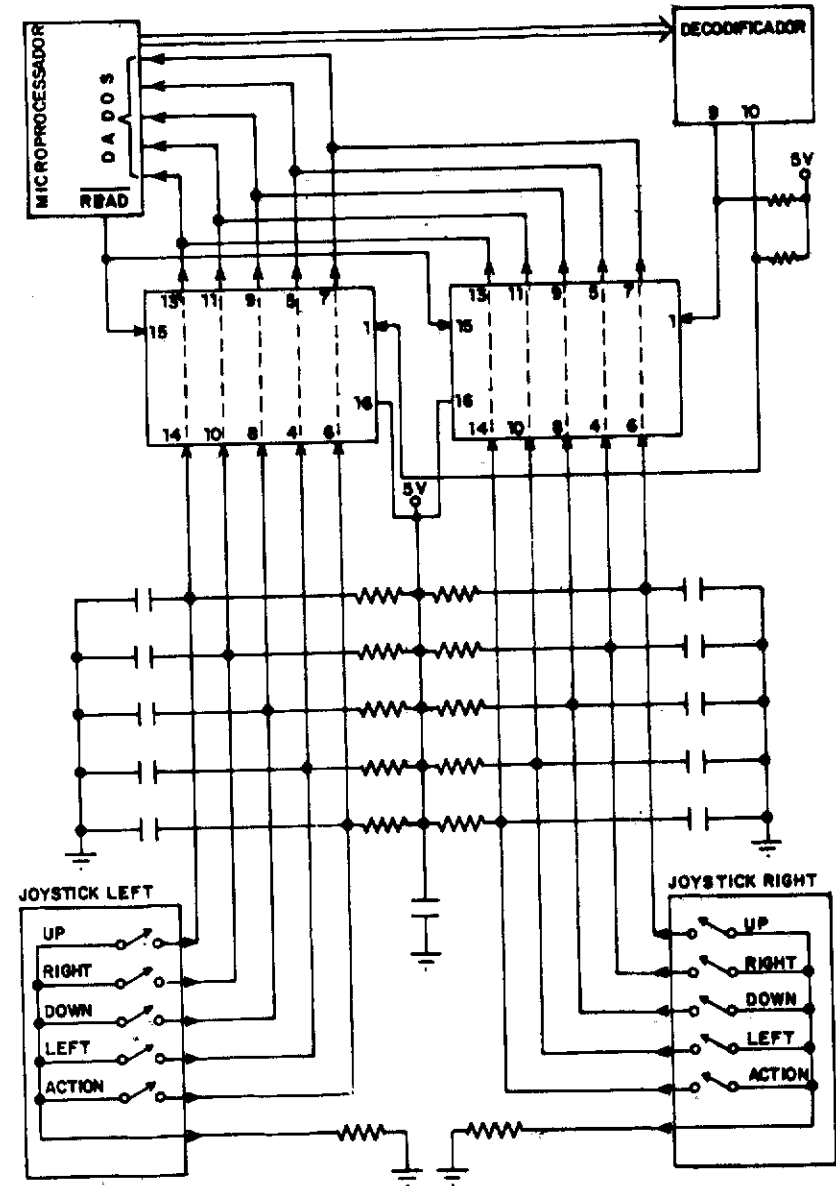


figura 11-1 - Circuito de joystick do video-game ODISSEY

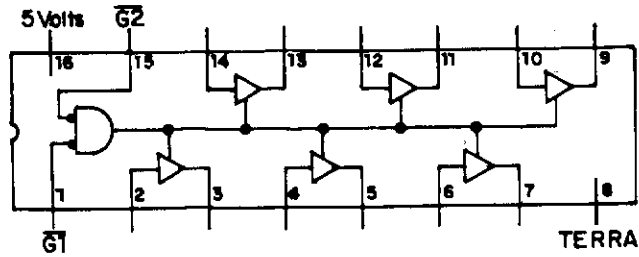


figura 11-2 - CI -74 LS 365

entradas de cada buffer (pinos 14,12,10,4,6) são transmitidos às respectivas saídas. Note que a condição de tristate do buffer é controlada por uma porta AND com entradas negadas (o que é indicado pelas duas bolinhas)

O microprocessador coordena todo o processo de recepção dos dados.

Quando deseja dar entrada nos dados do joystick seu terminal READ (leitura) é levado a nível 0, o que habilita os gates G2 dos dois buffers.

Entretanto, os dados não são transmitidos pois os gates G1 permanecem em nível 1, desabilitados. O decodificador, controlado pelo microprocessador, baixa a nível 0 o pino 9 ou o 10, habilitando o gate G1 de um dos buffers e permitindo que os dados desse joystick selecionado cheguem à barra de dados e sejam processados.

(2) ENDEREÇAMENTO DE ROM DO TK

Um circuito razoavelmente complexo é visto na figura 11-3, sendo uma simplificação do esquema de endereçamento da ROM da linha de microcomputadores SINCLAIR (no Brasil TK). Por envolver muitos conceitos importantes vale a pena analisá-lo.

As operações com a ROM ("read only memory" - memória para gravação de dados não alteráveis) podem ser divididas em dois tipos: (1º) leitura "normal" de dados e instruções; (2º) leitura de caracteres a serem apresentados na tela. Para dar máximo aproveitamento ao

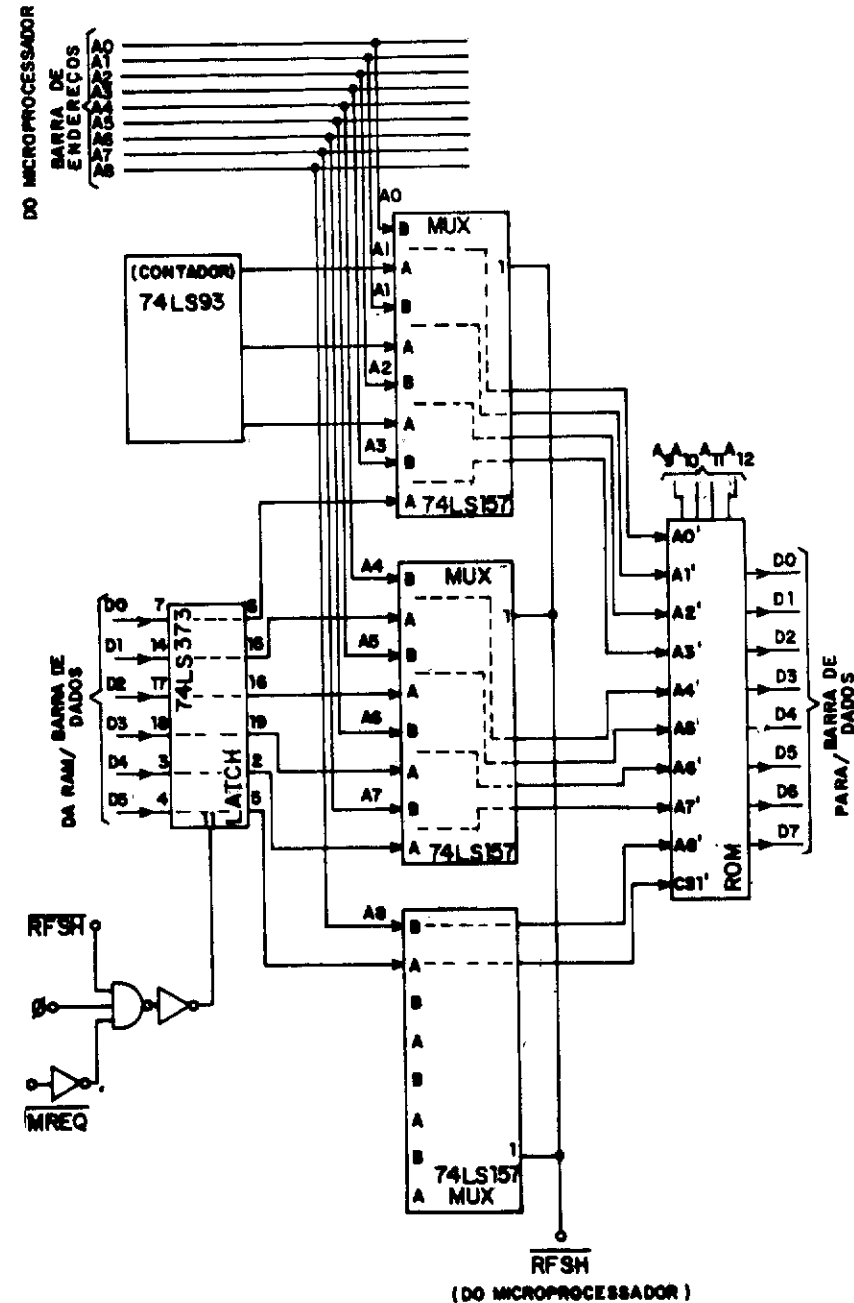


figura 11-3 - Circuito de endereçamento do TK

microcomputador projetou-se um circuito com 3 multiplexadores (ou multiplexers-MUX, 74 LS 157) que permite ao microprocessador executar operações do 1º tipo enquanto simultaneamente e automaticamente são realizadas as operações do 2º tipo. Ou seja, enquanto o microprocessador processa um programa o circuito cuida do vídeo.

O carácter a ser apresentado na tela é escrito (conforme desenvolvimento do programa) na RAM (memória de escrita e leitura), como 1 byte - sendo 6 bits para identificar o carácter e 2 bits de controle.

O microprocessador executa a operação de leitura (na RAM) do byte a ser apresentado na tela naquele momento. Para isso seu terminal MREQ é levado a \emptyset . Como neste instante RFSH=1 só resta o pulso positivo do clock para que todas as entradas da porta NAND estejam em nível alto, resultando \emptyset em sua saída, que é transformado em nível 1 pela inversora, habilitando o latch 74 LS 373. Os dados D_0 a D_5 que estão sendo lidos na RAM são armazenados ali, convertendo-se em saídas para os multiplexadores (MUX-74 LS 157). Entretanto esses bits do latch ainda não tem efeito sobre os MUXs, pois as portas 'A' estão desabilitadas pela seleção (pino 1-RFSH-nível alto habilita portas 'B', nível baixo habilita portas A). Eles são meramente armazenados para as operações seguintes.

No início do processo o microprocessador endereça a ROM, fazendo uma leitura de dados que o informam sobre qual instrução executar. Nesta condição seu pino RFSH ("refresh", refrescamento) está em nível 1 habilitando as entradas 'B' dos multiplexadores e bloqueando aquelas marcadas 'A'. Com essa divisão apenas os endereços recebidos do microprocessador são percebidos pela ROM, ficando sem efeito para ela os sinais originados no contador (74 LS 93) e no latch.

Após ler uma instrução da ROM o microprocessador passa a decodificá-la internamente, e todo o sistema deveria ficar ocioso esperando que ele termi-

ne essa tarefa. Entretanto, nesta condição o pino RFSH é levado a nível \emptyset , habilitando as entradas 'A' recebidas do latch e do contador e bloqueando em dereços vindos do microprocessador. Agora o código do carácter torna-se um endereço para a ROM.

Cada carácter é gravado como uma sequência de 8 bytes na ROM, isto é, uma matriz de 8 bits x 8 bits. Bit = 1 significa um ponto de tela "aceso" e bit = \emptyset ponto "apagado". O código de 6 bits recebidos da RAM permitem distinguir

$$2^6 = 64$$

caracteres na ROM. Cada um deles é mostrado linha por linha (num total de 8) na tela, através de um sistema de ilusão ótica usado pelo televisor. O contador fornece os três primeiros bits que formam o endereço da ROM (A_0, A_1, A_2), e são eles que determinam qual das 8 linhas do carácter deve ser mostrado naquele momento. Observe que

$$2^3 = 8.$$

(3) DETECTOR/CONTADOR

O detector/contador é um circuito largamente empregado na indústria e em instrumentação. Seu circuito (simplificado) é visto na figura 11-4. podendo ser expandido para quantos dígitos for necessário.

O circuito é baseado num fototransistor (transistor que conduz quando recebe luz na região da base, permanecendo cortado em caso contrário) acoplado a um LED. Na realidade os dois formam um único dispositivo numa embalagem de plástico com uma fenda (região pontilhada na figura) entre eles.

Mantendo a região da fenda livre a luz do LED atinge a base do transistor, fazendo-o conduzir e provocando uma queda de tensão no resistor R_L

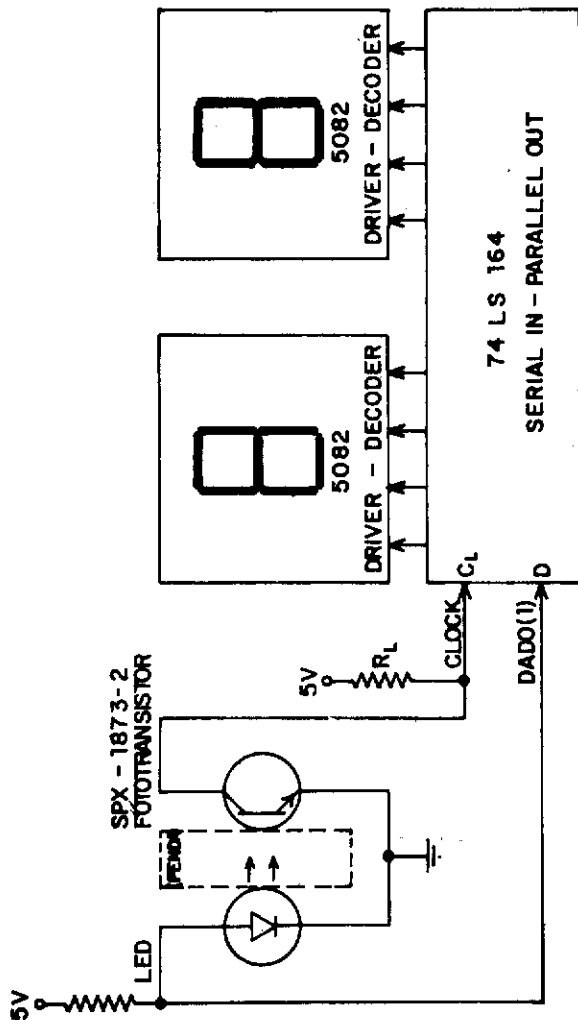


figura 11-4 - Circuito detector/contador

que é vista como nível \emptyset no clock do registro de deslocamento serial in - parallel out (74 LS 164). Quando um objeto é interposto na região da fenda o fototransistor deixa de receber a luz do LED, entrando em corte e levando o nível lógico do clock a 1 (não há queda de tensão em R_L). Após a retirada do objeto o fototransistor volta a conduzir, restabelecendo nível \emptyset no clock, o que completa um ciclo (figura 11-5).

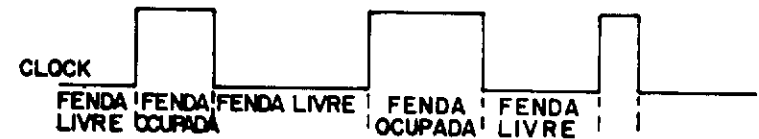


figura 11-5 - Posição na fenda e nível lógico do clock

A cada ciclo do clock o dado (mantido permanentemente em 1) é deslocado no registro de deslocamento 74 LS 164, aparecendo como dois números BCD (binary coded decimal) na saída, que são decodificados e transformados em dígitos pelo driver-decoder.

Observe que as únicas limitações do circuito são os tempos de corte do fototransistor e pull-up do resistor R_L , na casa de nanosegundos (1 segundo dividido por 1 bilhão). Isso permite contagens de dezenas de milhões por segundo com precisão quase absoluta.

(4) SELETOR DE CANAIS DE TV

Nos televisores antigos a seleção de canais era feita por um eixo que, girando, fazia contato entre uma bobina (indutor) selecionada e um capa-

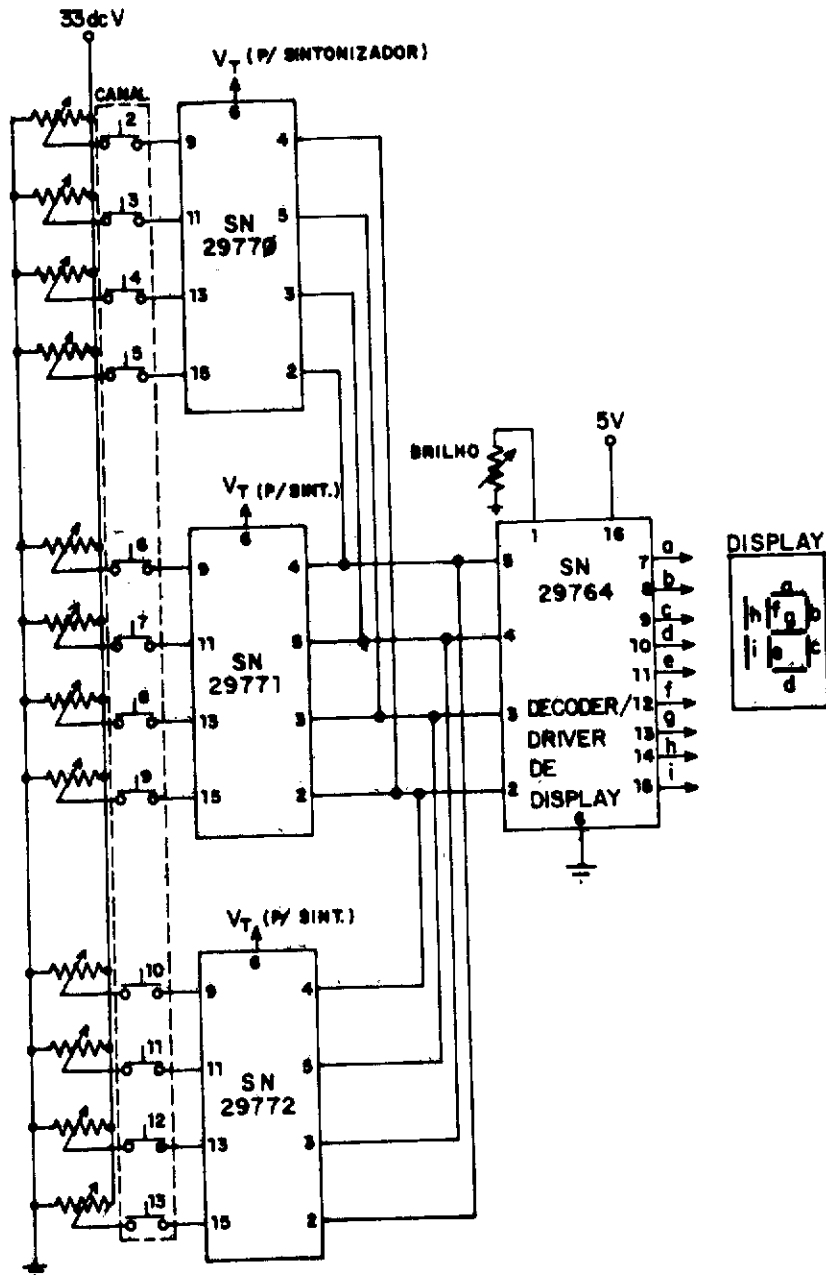


figura 11-6 - Seletor digital de canais de TV

citador, resultando um circuito ressonante à frequência do canal desejado.

Nos receptores modernos a bobina é mantida fixa e é variada a capacitância, através do uso de VARICAP (semicondutor polarizado inversamente cuja capacidade - FARAD - varia com a tensão reversa aplicada). Portanto, variando-se a tensão do varicap obtêm-se a frequência de ressonância correta para o canal.

Num circuito típico para esse fim (figura 11-6), a pressão da tecla de canal provoca alimentação de 3 a 30 Volts numa das quatro entradas do decodificador (SN 29770, SN 29771, SN 29772). Resulta uma tensão V_T para a varicap do sintonizador correspondente ao canal. A tensão na entrada e tensão V_T resultante são ajustáveis através do potenciômetro (a juste fino) da tecla. Até aqui temos Eletrônica Analógica; a seguir encontramos a parte digital propriamente dita.

O decodificador percebe a tensão ou não na entrada como bit 1 ou 0, respectivamente, e em consequência apresenta 4 bits na saída. É possível, assim, discriminar 16 canais:

B I T s	CANAL
0 0 0 1	1
0 0 1 0	2
0 0 1 1	3
⋮	⋮
1 1 1 1	15

Os quatro bits do decodificador de entrada são conectados a um decoder / driver de display (SN 29764), que aciona 7 - LEDs - display. Observe que este deve ser capaz de marcar a primeira dezena, além de 0 a 9 na segunda casa; neste caso ele recebe o nome "display de 1 dígito e 1/2".

CAPÍTULO 12 – PROJETOS PARA VOCÊ MONTAR

Apresentamos a seguir 3 dispositivos para o leitor montar e usar. São relativamente simples, baratos e com componentes de fácil acesso.

Mesmo para aqueles que não pretendem executar tais montagens recomendamos que acompanhem atentamente o texto, pois ele contém informações práticas e teóricas de interesse.

(1) PONTA DE PROVA LÓGICA

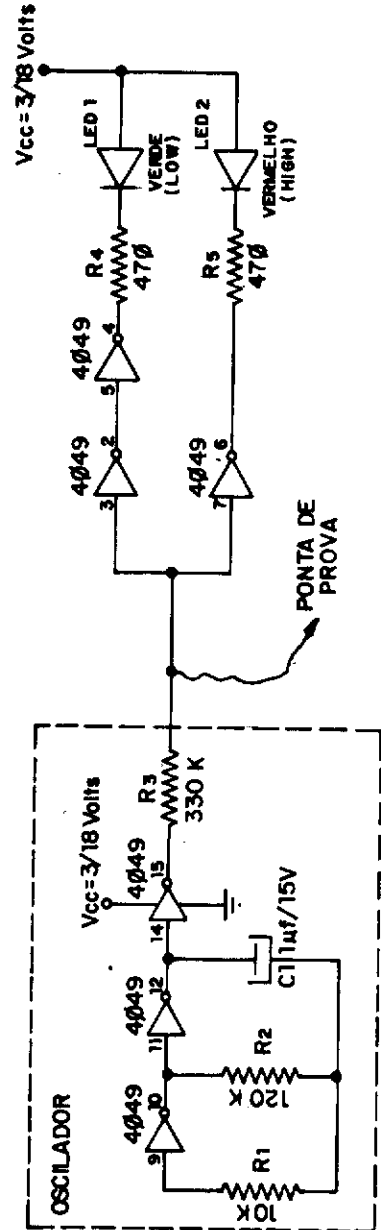
A PONTA DE PROVA LÓGICA (OU MEDIDOR DE NÍVEIS LÓGICOS) já estudada no capítulo 10 é apresentada aqui como sugestão para montagem, devido à sua enorme utilidade e fácil execução.

O circuito é visto na figura 12-1a, destacando-se em 12-b o CI 4049. Como foi analisado no estudo de CMOS é recomendável o uso do 4049B (com proteção) e MOTOROLA 14049.

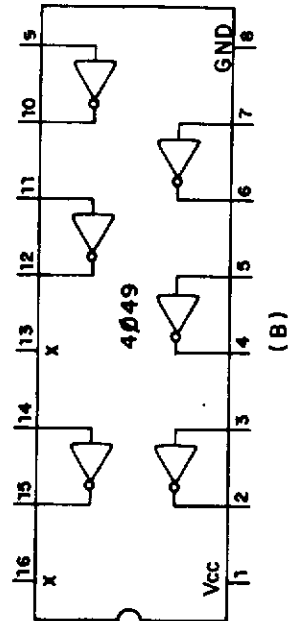
O lay-out da placa de circuito impresso consta na figura 12-2, em tamanho natural - lado dos componentes, podendo ser manipulada como uma caneta. Na verdade, com um pouco de habilidade e cuidado é possível montar todo o dispositivo diretamente no CI, sem usar circuito impresso (solde os componentes nos pinos do CI, tomando cuidado com contactos indesejáveis).

(2) CRONÔMETRO DIGITAL

A montagem de um cronômetro digital com al-



(A)



(B)

figura 12-1 - Ponta de prova l6gica (a) circuito (b) CI 4049

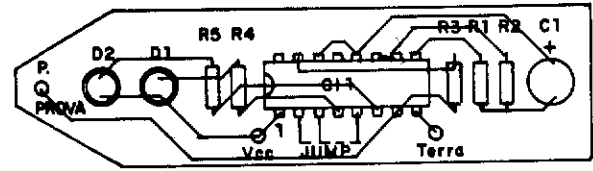


figura 12-2 - Lay-out da placa do circuito impresso da ponta de prova l6gica

cance de 0 a 99 segundos 6 apresentada a seguir , sendo de baixo custo e f6cil execu6o. Uma troca de componentes pode mudar sua escala de segundos para minutos, h6ras ou qualquer outra unidade.

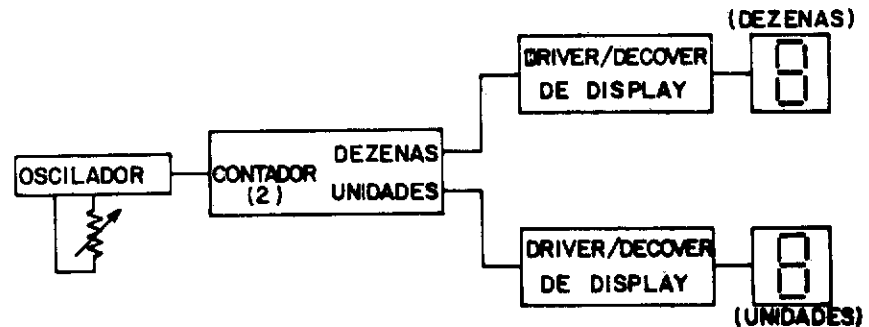


figura 12-3 - Esquema do cron6metro digital

Um circuito oscilador (como estudado no ca pítulo 6), fornece pulsos para todo o sistema, tendo sua frequ6ncia controlada por um potenci6metro. Tais pulsos funcionam como clock para o est6gio contador,

(2 contadores decimais), que a cada um deles incrementa uma unidade a contagem. Esta é dividida em duas - unidades e dezenas - que dão entrada nos respectivos drivers - decoders, sendo ali decodificadas e apresentadas aos displays. Uma chave no oscilador permite parar a contagem, enquanto que outra no contador é destinada ao reset (reinício).

O esquema elétrico do cronômetro digital é visto na figura 12-4, destacando-se os estágios.

O oscilador, baseado no CI 4011, tem sua base de tempo ajustável pelo trimpot de 1 M Ohms. Como a frequência de oscilação corresponde à constante RC ($2\text{ M} + \text{trimpot} \times 220\text{ nF}$) basta trocar esses componentes para se obter outra escala (minutos, horas).

O estágio contador é composto por um só CI-4518 (figura 12-5), que contém 2 contadores de décadas (contam de 0 a 9, colocando a contagem como 4 bits na saída, e no 10 resetam automaticamente). Quando o pino 1-ENA ("enable A" habilita A) está em nível 0, a cada pulso positivo do clock resultará em nível 1 nas duas entradas da NAND, incrementando 1 unidade a contagem. No contador A o pulso de clock é recebido diretamente do oscilador. Para o contador B o clock é originado no MSB do contador A (Q8). Este é levado a nível 0 no reset após estar em 1 para as contagens de $8 = 1000\text{B}$ e $9 = 1001\text{B}$, o que caracteriza 1 pulso a cada dezena. A chave do reset, quando acionada, alimenta com 9 Volts os pinos 15-RESA (reset de A) e 7- RESB (reset de B), provocando o reset dos dois contadores (0000).

Os drivers/decoders de display (figura 12-6) recebem a contagem do estágio anterior na forma de BCD (Q8 - Q4 - Q2 - Q1), decodificando-os para o 7-LEDs - display. O pino 3 deste CI (LT - "light") é usado para teste, ascendendo todos os 7 LEDs quando levado a nível 1. No mesmo caso enquadra-se o pino 4-BI ("bright interrupt", interrupção de brilho) que apaga os 7 LEDs quando colocado em nível 1.

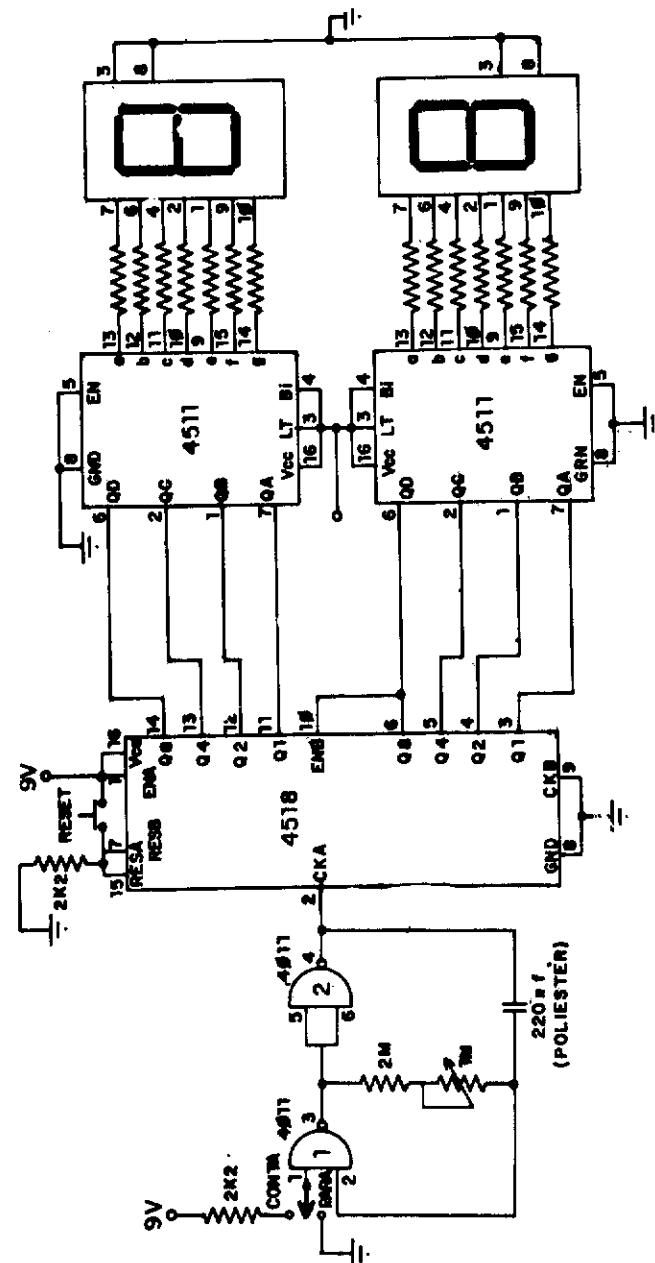


figura 12-4 - Circuito do cronômetro digital

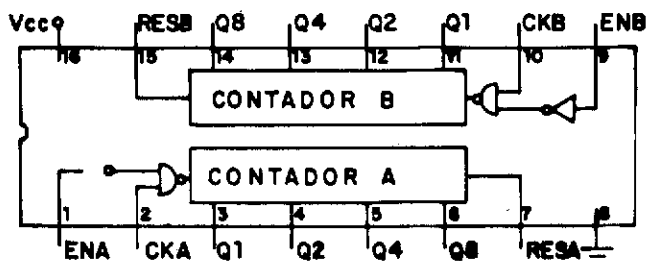


figura 12-5 - CI 4518 - 2 contadores

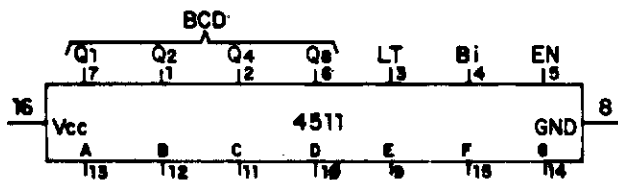


figura 12-6 - CI 4511 - driver/decoder de display

O 7-LEDs-display FND segue o mesmo esquema estudado no capítulo 2. Na figura 12-7 temos a vista traseira deste dispositivo, mostrando-se a numeração dos pinos.

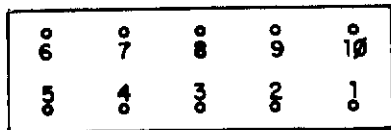


figura 12-7 - FND 500 - 7 LEDs DISPLAY - vista traseira

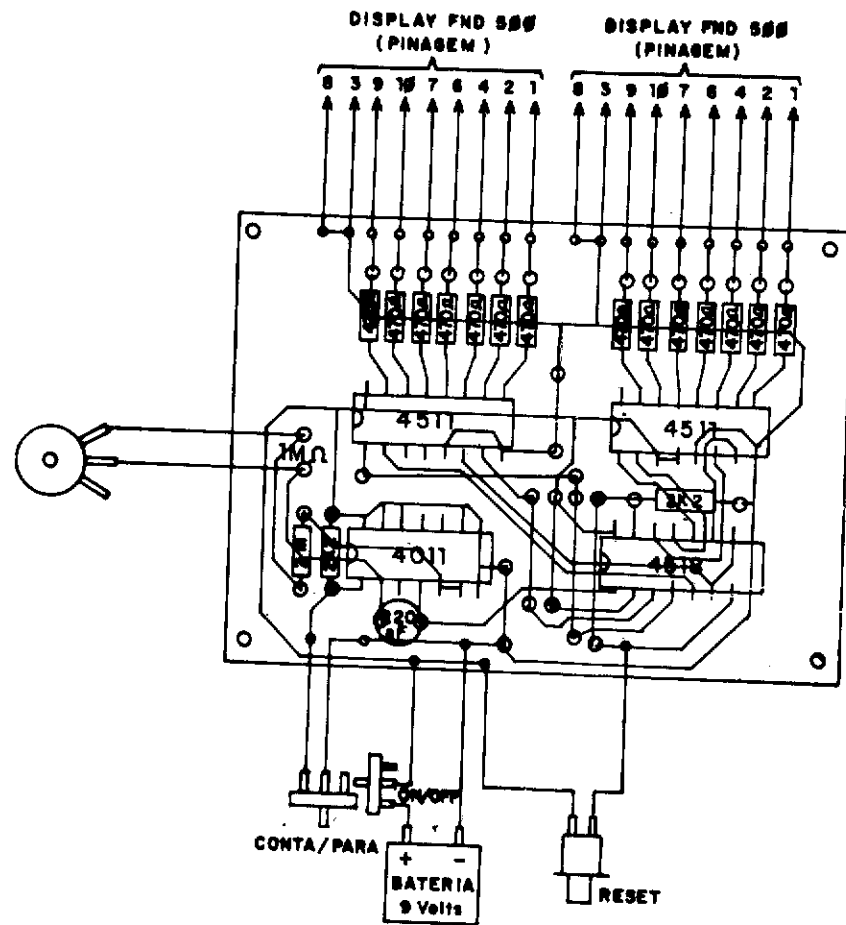


figura 12-8 - Lay-out do circuito impresso do cronômetro digital

O lay-out da placa de circuito impresso do cronômetro digital é apresentado na figura 12-8, com indicações para conexão de chaves e alimentação.

(3) GERADOR DE TONS

O GERADOR DE TONS é um dispositivo de grande interesse prático e teórico. Fornece um som programado do tipo encontrado em ambientes públicos e empresas, geralmente para introduzir um aviso ou informação.

No seu esquema geral (figura 12-9) um oscilador gera sinais de clock para o contador. Este tem quatro bits na saída, formando um número binário que é incrementado a cada pulso de clock. O contador dá entrada num decodificador que ativa a saída correspondente ao binário que está recebendo. Como o contador incrementa a contagem a cada clock, resulta que o decodificador ativará uma a uma suas saídas, sequencialmente. Estas estão ligadas ao estágio final oscilador

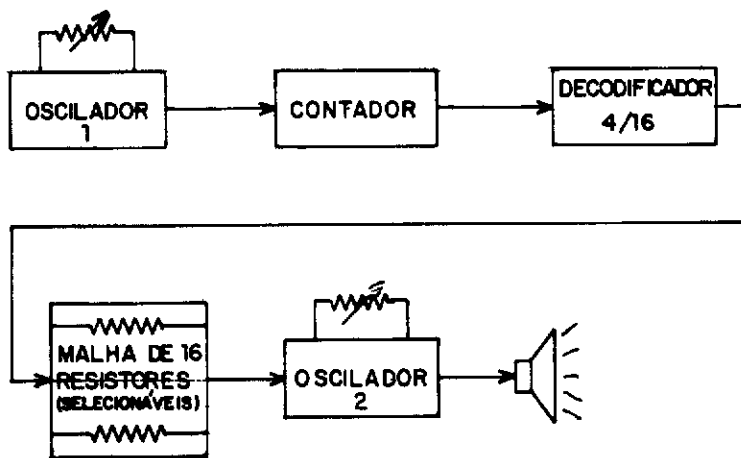


figura 12-9 - Esquema do gerador de tons

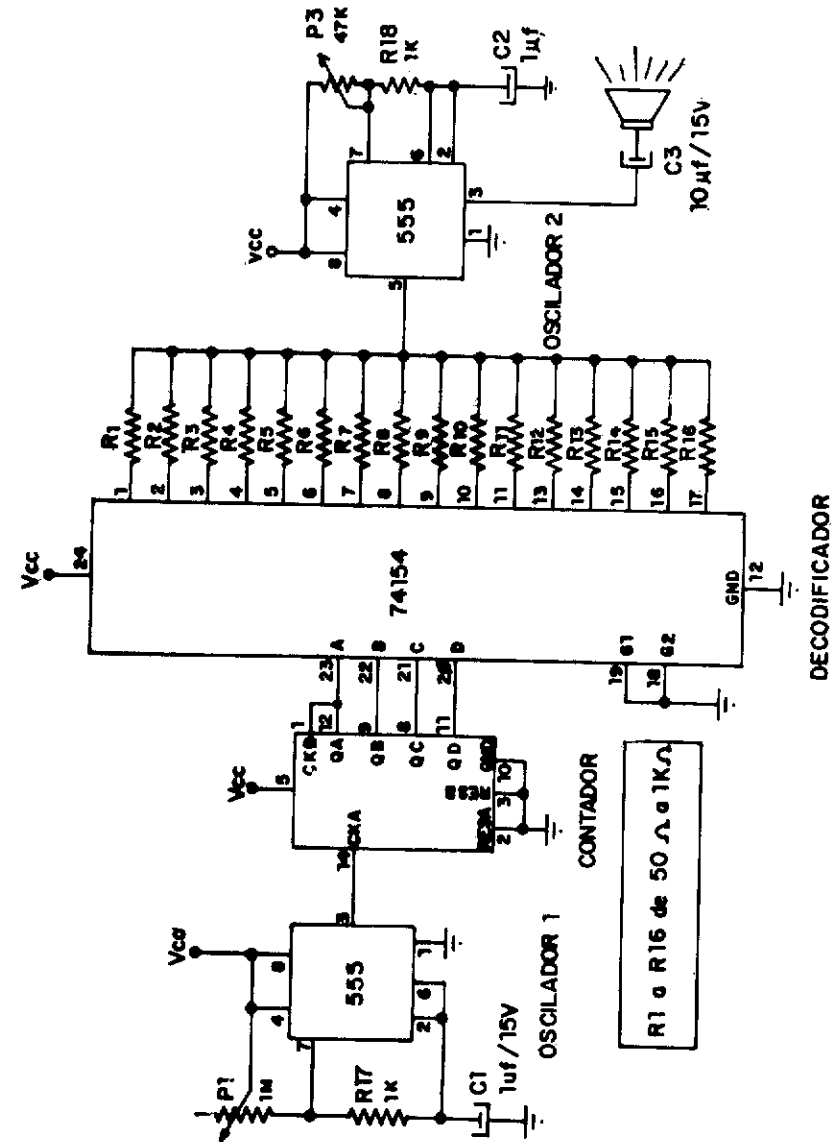


figura 12-10 - Circuito do gerador de tons

/alto falante, cuja frequência (e tom audível) depende do resistor de ligação. Selecionando-se valores adequados para os resistores obter-se-á a escala sonora desejada.

O circuito elétrico, destacando os estágios, é visto na figura 12-10.

O primeiro oscilador é baseado no CI 555 (figura 12-11), um dos mais conhecidos e usados dispositivos da Eletrônica Digital. Ele pode funcionar como multivibrador estável (como neste projeto) ou monoestável. O capacitor C 1 é carregado através do P 1 e R 1 até a voltagem de "threshold" (limiar) TH do pino 6, quando então é descarregado (para dentro do CI) no pino 7 - DISC ("discharge", descarga). A descarga provoca um pulso negativo no pino 2 - TRIG ("trigger", gatilho), que leva à emissão de um pulso (oscilação) no pino 3 - OUT. Note que P 1, R 1 e C 1 podem ser selecionados e ajustados para a frequência desejada. O pino 4 - RESET, quando em nível 0 reseta o CI (em operações de monoestável) ou o desliga (quando estável).

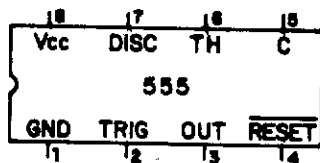


figura 12-11 - CI 555

O estágio gerador de sequência recebe os pulsos do 555, transformando-os em clock para o CI 7493 - (figura 12-12), composta de dois contadores de 2 bits cada (contador A: Q A - Q B; contador B: Q C - Q D). Conectando a saída do primeiro ao clock do segundo obtemos um só contador de 4 bits. Assim, teremos a sequência:

PULSO DO 555	QA	QB	QC	QD
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16	0	0	0	0

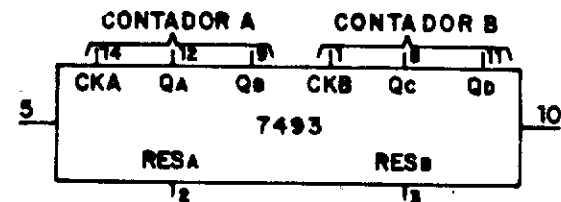


figura 12-12 - CI 7493 - Contador

Partindo de 0000 há incremento de 1 (em binário) a cada pulso do 555, até o 16º, quando então obtem-se novamente 0000 e a sequência recomeça. RESA (pino 2) e RESB (pino 3) são reset para os contadores, sendo mantidos inativos.

O decodificador 74154 (figura 12-13) possui 16 saídas (pinos 1 a 17, exceto o 12). Aquela que corresponder ao número de pulso sendo recebido na entrada (A, B, C, D, pinos 23, 22, 21, 20) é mantida em nível 0, as outras em 1. Por exemplo, no 9º e no 12º pulso do 555 teremos:

PULSO DO 555	QA	QB	QC	QD
9	1	0	0	1
12	1	1	0	0

G1 e G2 são controles de habilitação (ativos nível 0), sendo mantidos permanentemente aterrados.

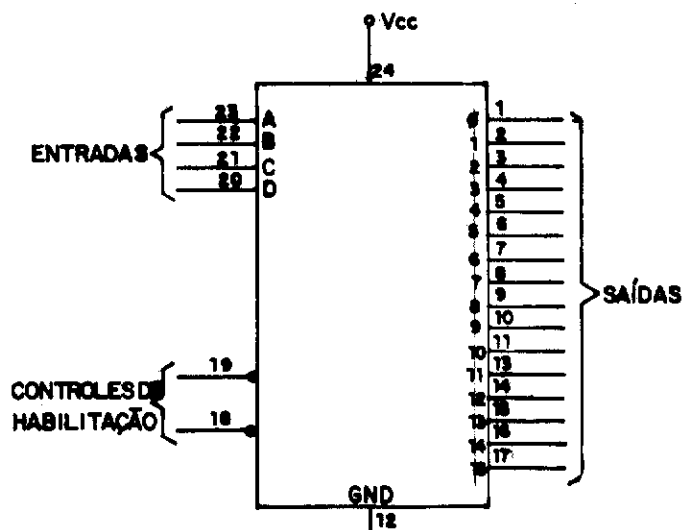


figura 12-13 - CI 74 154 - DECODIFICADOR 4/16

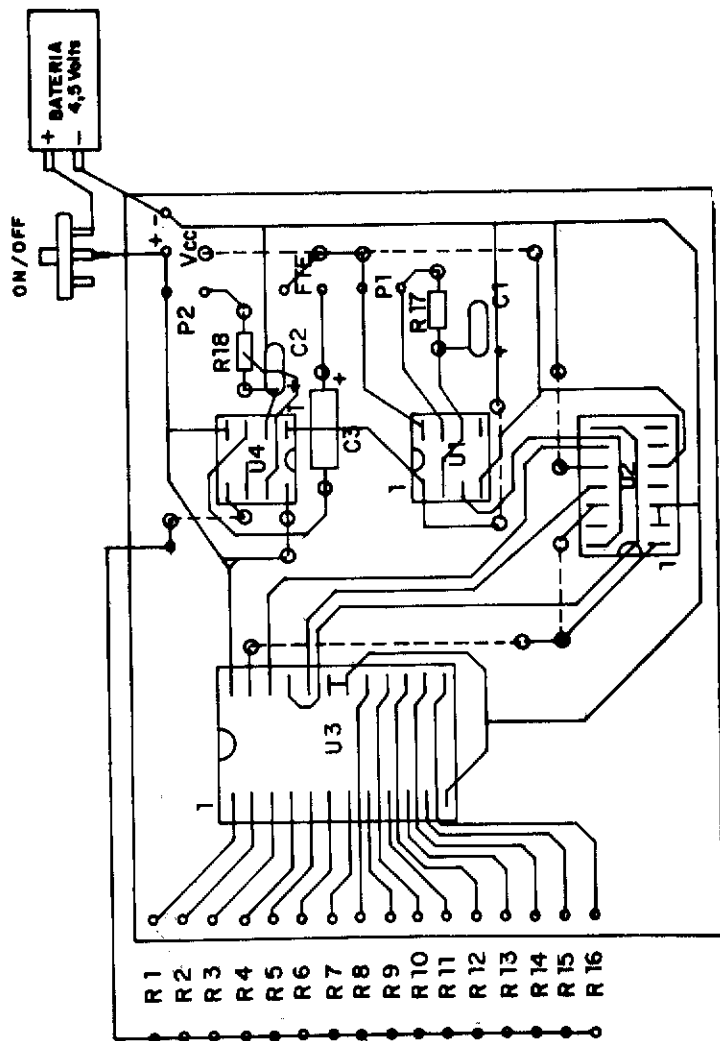
No estágio final encontramos novamente um 555, cujas oscilações se transformam em sinal de áudio para o auto-falante. Aqui apenas se acrescentou o pino 5 (no primeiro oscilador ele não é usado), C-CONTROL, cuja tensão controla a frequência das oscilações. Ligando este terminal à saída do decodificador temos sons (frequências no alto-falante) cujos tons dependem dos valores dos resistores usados para a ligação.

Os 16 resistores devem estar na casa de 50 a 1 K Ohms. Uma sugestão é fazer o primeiro 100 Ohms e incrementando de 50 Ohms os seguintes. O montador, entretanto, tem plena liberdade de escolha, desde bem que somente valores harmônicos produzem uma escala sonora adequada.

O ajuste final do tom pode ser feito pelo potenciômetro P 3 (tal como no primeiro oscilador). O alto-falante mais indicado é de 4 Ohms (tipo rádio transistor).

Para alimentação (V_{CC}) use 3 ou 4 pilhas comuns ou uma fonte regulada entre 5 e 9 Volts dc.

O lay-out do circuito impresso do gerador de tons é apresentado na figura 12-14.



CÓDIGO ASCII (EM HEXADECIMAL)

CARACTER	ASCII	CARACTER	ASCII
NUL	00	SP	20
SOH	01	!	21
STX	02	"	22
ETX	03	#	23
EOT	04	\$	24
ENQ	05	%	25
ACK	06	&	26
BEL	07	/	27
BS	08	(28
HT	09)	29
LF	0A	*	2A
VT	0B	+	2B
FF	0C	,	2C
CR	0D	-	2D
SO	0E	.	2E
SI	0F	/	2F
DLE	10	0	30
DC1	11	1	31
DC2	12	2	32
DC3	13	3	33
DC4	14	4	34
NAK	15	5	35
SYN	16	6	36
ETB	17	7	37
CAN	18	8	38
EM	19	9	39
SUB	1A	:	3A
ESC	1B	;	3B
FS	1C	=	3C
GS	1D	>	3D
RS	1E	V	3E
US	1F	?	3F

figura 12-14 - Lay-out do circuito impresso do gerador de tons.

CARACTER	ASCII	CARACTER	ASCII
@	40	'	60
A	41	a	61
B	42	b	62
C	43	c	63
D	44	d	64
E	45	e	65
F	46	f	66
G	47	g	67
H	48	h	68
I	49	i	69
J	4A	j	6A
K	4B	k	6B
L	4C	l	6C
M	4D	m	6D
N	4E	n	6E
O	4F	o	6F
P	50	p	70
Q	51	q	71
R	52	r	72
S	53	s	73
T	54	t	74
U	55	u	75
V	56	v	76
W	57	w	77
X	58	x	78
Y	59	y	79
Z	5A	z	7A
[5B	{	7B
\	5C		7C
]	5D	}	7D
^	5E	~	7E
_	5F	DEL	7F

AMBIENTE CASEIRO

A casa não é apenas um refúgio de madeira ou alvenaria, é o lar onde a união e o companheirismo se desenvolvem.

*

A paisagem social da Terra se transformaria imediatamente para melhor se todos nós, quando na condição de espíritos encarnados, nos tratássemos, dentro de casa, pelo menos com a cortesia que dispensamos aos nossos amigos.

*

Respeite a higiene, mas não transfigure a limpeza em assunto de obsessão.

*

Enfeite o seu lar com os recursos da gentileza e do bom-humor.

*

Colabore no trabalho caseiro, tanto quanto possível.

*

Sem organização de horário e previsão de tarefas é impossível conservar a ordem e a tranquilidade dentro de casa.

*

Recorde que você precisa tanto de seus parentes quanto seus parentes precisam de você.

*

Os pequeninos sacrifícios em família formam a base da felicidade no lar.

ANDRÉ LUIZ

(Página recebida pelo médium FRANCISCO CÂNDIDO XAVIER)

RÁDIO – TEORIA E TÉCNICAS DE CONSERTOS

A teoria de rádio é desenvolvida de uma maneira fácil e acessível, dos princípios elementares até as aplicações mais complexas. Paralelamente à descrição dos estágios do receptor são apresentadas as técnicas de diagnóstico do efeito e como consertá-lo. A abordagem inclui sistemas de transmissão AM, FM, HI-FI, estéreo e os respectivos circuitos no receptor. O livro é recomendado para assistência técnica e aficionados no mundo do rádio.

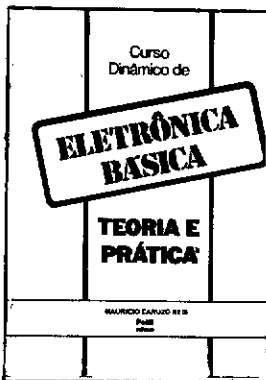


ELETRÔNICA DE VIDEOGAMES – TEORIA E MANUTENÇÃO

Este é um livro destinado a técnicos e hobbistas interessados em conhecer a fundo essas fascinantes máquinas. Desenvolve toda a teoria, desde Eletrônica Digital até a técnica de geração de imagens no receptor de televisão, apresentando também as bases para a programação e cópia de cartuchos.

Os videogames ATARI e ODISSEY são analisados detalhadamente como exemplos, registrando-se todos os seus circuitos e discutindo-se a teoria e operação dos estágios.

Um capítulo especial trata de instrumentos e técnicas de consertos, inclusive a conversão NTSC-PAL, possibilitando aos técnicos e oficinas entrar nesse rendoso e crescente mercado.



ELETRÔNICA BÁSICA TEORIA E PRÁTICA

Este é o caminho mais curto para se chegar ao reino da Eletrônica. Destacando o que há de realmente importante na área e mostrando os mínimos detalhes toda a técnica usada pelos veteranos de oficina e laboratórios, o livro habilita o leitor a atingir pontos mais avançados como rádio, TV e computadores.



CURSO DINÂMICO DE FOTOGRAFIA

Aprenda a fotografar, revelar filmes e fazer cópias em papéis sem utilizar qualquer equipamento especial, uma oportunidade rara para você conhecer os segredos da fotografia.

AUTOMÓVEIS – GUIA DE CONSERTOS

Finalmente chegou o braço direito do motorista. Com este guia o motorista já não terá tanta necessidade de levar o seu carro ao mecânico quando surgir algum problema, basta pegar o seu guia, abrir na página certa e lá estará a solução, de uma forma muito simples e até simpática, amenizando até aquele nervosinho que as vezes passamos quando não conseguimos identificar a solução para o problema que está ocorrendo.

E sem falar na economia que o motorista terá com mecânicos, pois muitos problemas parecem grandes, mas que às vezes basta um pequeno massete para resolver, economizando no dinheiro e no tempo. Os defeitos são apresentados em ordem alfabética para uma melhor consulta, e as soluções são fáceis de assimilar, para qualquer marca de carro, a álcool ou a gasolina. Ideal para dar de presente à uma pessoa que gostamos, ideal para ter no porta luvas, para os momentos inesperados.



TELEVISÃO – Teoria e Consertos - Cores/PB

Televisão era um segredo desvendado apenas por meia dúzia de cérebros ilustres; estranhos esquemas e fórmulas que era aprendidas misteriosamente por eles. Consertar a Televisão era pura mágica; "macetes" e "jeitinhos" que vinham do berço.

"TELEVISÃO - TEORIA E CONSERTOS" é uma obra para acabar com esses preconceitos, apresentando uma visão moderna e aberta sobre o assunto. O "segredo" é reduzido a uma teoria bem ordenada e fundamentada, conduzindo o leitor pela lógica do aparelho. "Macetes" e "Jeitinhos" são transformados em técnicas de aplicação universal, acessíveis a todos que tenham dedicação, esforço e disciplina.

Na primeira parte do livro é desenvolvida a teoria de transmissão e recepção de TV, a cores e preto e branco, analisando-se os circuitos de cada estágio do receptor, inclusive controle remoto. Na segunda parte são apresentados instrumentos para consertos e ajustes, roteiros de ajustes e técnicas de consertos, finalizando com um extenso guia de defeitos.

O esquema do livro é válido para estudantes que desejam dominar essa área indispensável, para interessados em entrar no negócio de assistência técnica e Técnicos experientes precisando aprofundar seus conhecimentos.



CONSTRUA SEU COMPUTADOR POR MEIO SALÁRIO-MINIMO

O livro é direcionado a engenheiros, técnicos e estudantes interessados num aparelho de baixo custo para prática de Eletrônica Digital, microprocessadores e programação assembler/código de máquina.

O computador proposto no livro — MICRO-GALENA — pode ser construído no fundo do quintal, sem circuito impresso e utilizando ferramentas caseiras comuns. Todos seus componentes são facilmente encontrados no comércio nacional, inclusive pelo reembolso postal, pelo custo de meio salário-mínimo.

E nas horas vagas o MICRO-GALENA estará disponível para um joguinho.



MANUTENÇÃO DE MICROCOMPUTADOR

Quase 500.000 microcomputadores estão implantados no Brasil. O que fazer para que continuem funcionando? A resposta é desenvolver urgente e rapidamente um contingente de mão-de-obra especializada em manutenção, capaz de fazer frente ao desafio da vertiginosa produção de micros.

A PETIT EDITORA presta sua colaboração e lança a 3ª edição de "MANUTENÇÃO DE MICROCOMPUTADORES", com detalhada descrição de técnicas, teorias e instrumentos envolvidos na manutenção de microcomputadores.

Os primeiros capítulos apresentam as bases teóricas — Eletrônica Digital, Microprocessadores e Arquitetura de Microcomputadores — incluindo um estudo sobre os microprocessadores Z-80, 6502, 68.000 e guia dos micros TK, CP e APPLE.

A redação e organização do livro é adequada para técnicos e engenheiros interessados em questões práticas de manutenção de microcomputadores, além de fonte de referência didática e consulta para estudantes que se iniciam na área.



CURSO DINÂMICO DE SILK-SCREEN

O que é Silk-Screen? Podemos chamar de um processo gráfico, através do qual é possível realizar estampas sobre as mais variadas superfícies e nos mais diversos materiais; madeira, tecido, papel, etc. Mas se você pensa que só pode ser feito com máquinas, está muito enganado, muitas pessoas já estão se saindo muito bem e ganhando muito com esta profissão, trabalhando apenas no processo manual. De que maneira? Bem, isso você verá quando tiver o curso em suas mãos, estudar, praticar e por mãos a obra. Aí você poderá fazer estampa em camisetas, circuitos impressos, adesivos, etc. E poderá fazer tudo isso, somente nas suas horas livres, ganhando duas vezes, ou se preferir, dedicar-se exclusivamente à nova profissão.



COLA

LIVROS EM PROMOÇÃO

A PETIT EDITORA sempre está lançando novos livros, e a primeira tiragem é feita em forma de promoção junto aos nossos clientes cadastrados, através de mala direta com descontos especiais. Caso você não tenha seu nome em nosso cadastro mas gostaria de receber nossos folhetos com promoções, envie-nos o cupom abaixo preenchido, hoje mesmo! Pode estar saindo um livro novinho.

DOBRA

DESEJO TER MEU NOME NO CADASTRO DA PETIT EDITORA, PARA RECEBER OS FOLHETOS PROMOCIONAIS;

Nome _____

End. _____ N.º _____

Apto. _____ Bairro _____

Cep: _____ Cidade: _____

Est.: _____

Profissão: _____ Fone: _____

Idade: _____

Assinatura

SELO

PETIT

EDITORA E MARKETING DIRETO LTDA.
CAIXA POSTAL 8414 - AG. CENTRAL
01051 - SÃO PAULO - SP

REMETENTE

Nome: _____
Endereço: _____
Bairro: _____
Cidade: _____ Est.: _____
Cep: _____