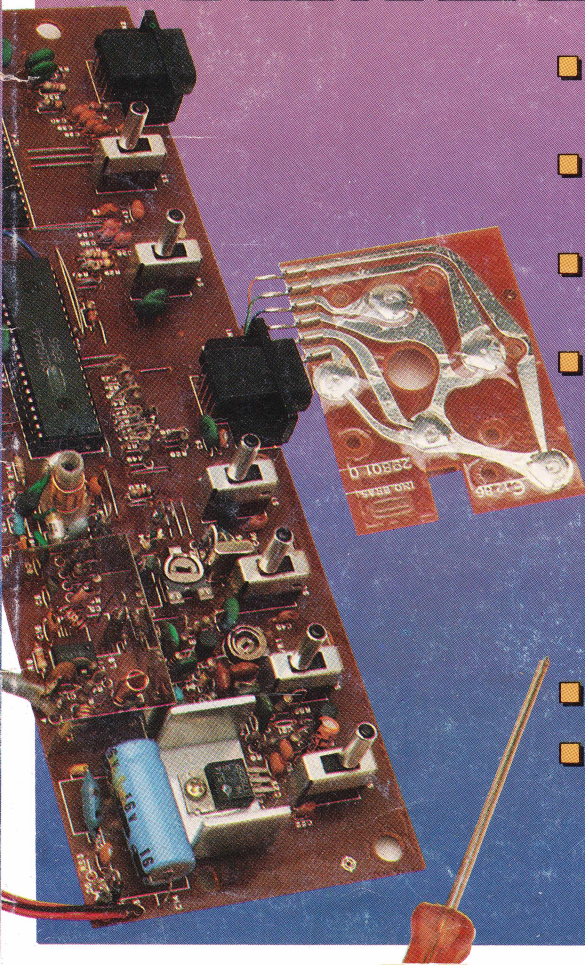


ENTENDA  
&  
CONSERTE

# VÍDEO GAMES



- INTRODUÇÃO AO SOFTWARE
- INTRODUÇÃO AO HARDWARE
- DESCRIÇÃO DO FUNCIONAMENTO
- DESCRIÇÕES DOS SEGUINTE VIDEOGAMES:  
ATARI  
ODISSEY  
MEGADRIVE (SEGA)  
NINTENDO  
MEGA-CD-ROM
- REPARAÇÃO
- TRANSCODIFICAÇÃO



*SÉRGIO R. ANTUNES*

# VÍDEO GAMES

editora  
**FITIPALDI**



**Direitos reservados da presente edição.  
Reprodução total ou parcial é expressamente  
proibida, sem a prévia autorização por escrito.**

**Printed in Brazil  
Impresso no Brasil**

**Impressão e acabamento  
W. ROTH S.A. — (011) 960-2988**

**COPYRIGHT BY  
EDITORA FITTIPALDI LTDA  
© 1994**

# APRESENTAÇÃO

A intenção e o propósito que me impulsionou a realizar esta obra foi o desejo de contribuir de alguma maneira na divulgação da tecnologia dos video-game.

Os video-games estão cada vez mais presentes nos lares brasileiros. E estão cada vez mais sofisticados. Apesar de seus avanços tecnológicos, uma reparação em video-game não é complexa, desde que conheçamos seu funcionamento e desde que adotemos uma metodologia de consertos eficaz.

Estes dois argumentos serviram de base para a elaboração deste livro:

Apresentar a teoria de funcionamento e orientar nas técnicas de reparos.

Esperamos que ao término desta leitura, possa estar apto para reparar ou transcodificar um video-game.

**Sérgio R. Antunes**

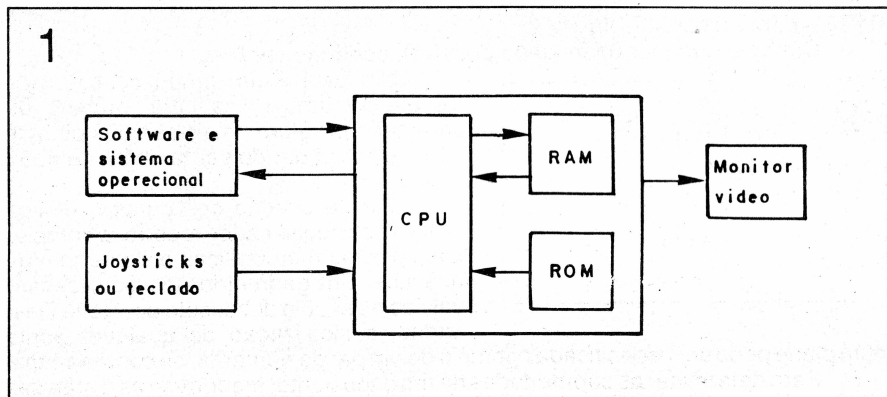
# CAPÍTULO 1

## INTRODUÇÃO AO SOFTWARE DO VIDEO-GAME

O segredo de todo video-game e de toda informática está no software. O software é um conjunto de programas, de códigos, de instruções, que fornecem ao equipamento suas múltiplas personalidades.

Para que o leitor possa compreender claramente todo processo da informática em um vídeo game, vamos simular um processo de dados.

Observando a ilustração da **figura 1**, observamos as diversas fases envolvidas no processamento de um software (dados).



Através do teclado ou joysticks o usuário se comunica com a máquina. O software identifica as combinações do teclado ou joysticks como instruções a serem processadas. Tanto os joysticks como outros dispositivos periféricos de entrada (tipo mouse, pistolas, etc), controlados manualmente, permitem que dados e programas sejam fornecidos para as memórias do microcomputador.

A memória RAM, uma memória volátil e temporária, mantém os dados e o programa memorizados durante o processamento.

A CPU - Unidade Central de Processamento - gerencia todos os estágios do video-game ou do microcomputador. A CPU reconhece o conjunto de instruções, os códigos, os programas. Para isso, ela lê e escreve nas memórias RAM e lê o conteúdo da ROM.

A ROM é a memória que contém permanentemente memorizado os programas internos da máquina. A ROM também pode conter uma linguagem, como o BASIC, para programação ou ampliação de funções.

Um monitor de vídeo ou a tela de um TV mostra o resultado visual de todo o processamento.

## A PRODUÇÃO DE UM VIDEO-GAME

Um video-game é um software sofisticado. Dizemos assim pelo fato de ele conter gráficos coloridos, efeitos visuais e efeitos sonoros, além das instruções normais de qualquer software.

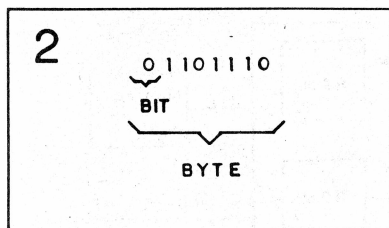
Todo video-game é um jogo de computador. O programador ou criador do software é um artista, que além de produzir todos códigos necessários à execução de um programa, também cria ilusões e estímulos em alta velocidade. Deste modo, o jogador que pretende um bom desempenho no video-game deve ter reflexos rápidos e excelente coordenação motora, enquanto que o criador do software, precisa encontrar a combinação exata das variáveis para torná-lo ao mesmo tempo divertido e emocionante.

Para se criar a ilusão de movimento de um carro, é obtido pela mudança das imagens na tela várias vezes por segundo. O criador do software deve compactar na memória todas as informações necessárias que possibilitem essas imagens instantâneas.

## CRIAÇÃO DE UMA FIGURA

Vamos explanar aqui como é criado uma figura ou imagem pelo software. Como ponto de partida, devemos saber que todo video-game e todo computador usam a lógica binária, de dois níveis, para processar as informações em BITS e BYTES como mostra a (figura 2).

Um bit é a menor quantidade possível, podendo ser 0 ou 1.



Um byte é um grupo de bits, que representa uma única letra, número ou caracter. Os valores mais comuns de byte são 8 e 16, podendo ser também de 4, 32 e 64.

Para a criação de figuras e efeitos visuais, o programador recorre a complexos cálculos matemáticos e de geometria analítica. Em geometria analítica, desenvolvida no século dezessete por René Descartes, a localização de qualquer ponto

num plano pode ser especificada por meio de um par de números, ou coordenadas.

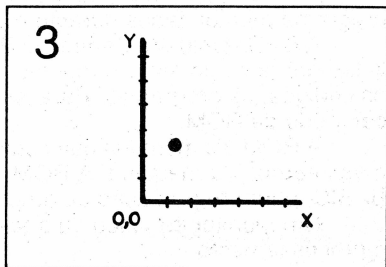
Para determinar as coordenadas de um dado ponto, medem-se as distâncias que o separam de duas linhas de referência que se interceptam em ângulos retos. A linha horizontal denomina-se eixo dos X e a linha vertical de eixo dos Y; o ponto de intersecção (origem) é zero.

Usando coordenadas cartesianas, qualquer imagem bidimensional pode ser reduzida a uma lista de coordenadas dos pontos que formam a imagem. Efetuando-se a operação inversa, estabelecidas as linhas de referências e uma escala, uma lista de coordenadas pode ser convertida numa imagem.

Por exemplo: a equação  $X - Y = 4$  gera uma reta; a equação  $X$  elevado ao quadrado -  $4Y = 0$  gera uma parábola.

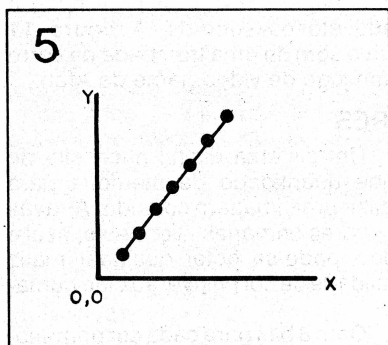
Nas figuras 3 a 7 ilustramos várias seqüências da geração de imagem pelo método descrito acima.

Na figura 3, marcamos um ponto, com uma distância definida a partir do ponto de intersecção do eixo dos X com o eixo dos Y.



Na **figura 4** move-se um ponto somando-se ou subtraindo-se números do valor de uma ou das duas coordenadas. Nesta figura, o ponto move-se duas unidades para a direita somando-se 2 na coordenada X.

Na **figura 5** cria-se a ilusão de uma linha contínua, inserindo-se vários pontos igualmente espaçados entre dois pontos (1,1) e (8,8). As coordenadas dos pontos intermediários são obtidas somando-se um valor constante às coordenadas de cada ponto sucessivo.



Finalmente, ilustramos na **figura 7** um programa que desenha os triângulos, utilizando as coordenadas dos 3 vértices do triângulo externo. O ponto médio de cada triângulo torna-se um vértice do triângulo sucessor menor. A coordenada X de um ponto médio é a metade da soma das coordenadas X dos pontos extremos desse lado. A coordenada Y é a metade da soma das coordenadas Y dos pontos extremos.

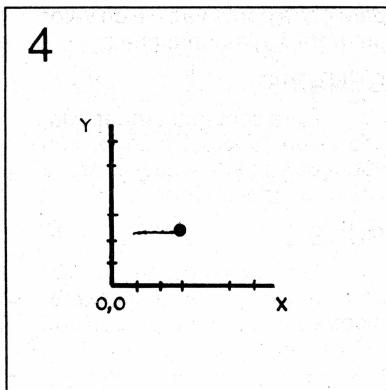
## PASSO A PASSO

Vamos ilustrar o software de uma figura passo a passo.

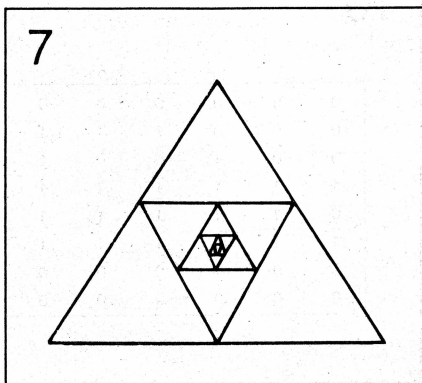
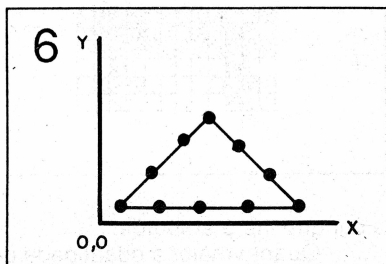
Cada figura que aparece no monitor ou na tela do TV, gerada no videogame, começa como um simples desenho, como o carro da **figura 8**.

Em seguida, sobrepõe-se um quadriculado ao desenho, adaptando-se o formato do carro, de tal forma que este passa ser codificado posteriormente como mostra a (**figura 9**).

A codificação como mostra a (**figura 10**) envolve a atribuição do valor zero



Na **figura 6**, foram marcados 3 pontos que assinalam os vértices. Em seguida, marcavam-se os pontos dos segmentos que os unem.





para cada ponto vazio e do valor um para cada ponto cheio.

## CENÁRIO

Para construir um cenário, cria-se um fundo constante, com variações ligeiras que darão a ilusão de movimento.

## SONS

Um video-game produz sons pelo envio de sinais eletrônicos ao alto falante. Através de

8



uma linguagem (BASIC, por exemplo), é gerado efeitos sonoros. A figura 11 ilustra o som de uma trombada de carro em um jogo de video-game do Atari.

## CORES

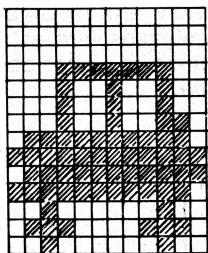
Um sistema digital necessita de grande quantidade de memória para produzir uma imagem colorida. Através das 3 cores primárias - vermelho, azul e verde - pode-se obter qualquer matiz (tonalidade de cor) visível ao olho humano.

Com 8 bits para cada cor primária, pode-se obter os pixels. Multiplicando por 3 teremos 24 bits de memória (8 para cada tonalidade primária) para registrar

a cor que lhe é atribuída.

Quanto maior a quantidade de cores que se deseja obter, e quanto maior a resolução da imagem desejada, maior quantidade de pixels deverá ter. Assim, uma tela de 24 milhões de pixels necessitará de uma memória de 576 milhões de bits

9



10

0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	0	0	0
0	0	1	0	0	1	0	0	1	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	1	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

10 Rem Explosion  
 20 For X  
 30 Sound 0, 200, 4, 10 -- ABS (X)  
 40 Sound 1, 255, 4 10 -- ABS (X)  
 50 Sound 2, 255, 4 10 -- ABS (X)  
 60 Sound 3, 150, 4 10 -- ABS (X)  
 70 Next X  
 80 Return

(24 x 3 cores primárias x 8 bits cada).

Um software de computação gráfica para video-game, combina boa resolução cromática com pixels transformados em bits e armazenados nas memórias do video-game.

## MAPEAMENTO DA MEMÓRIA

Pois bem, falamos sobre a grande capacidade de memória necessária para o armazenamento e processamento das imagens. Para tornar possível o acesso, programas e dados são armazenados como seqüências de bits (0 e 1) em células de memórias RAM. Cada célula contém um único bit, que pode ser zero ou um, e todas elas estão agrupadas em unidades maiores, denominadas bytes. AROM, que não pode ser apagada, tem uma configuração similar.

A **figura 12** ilustra a explicação acima.

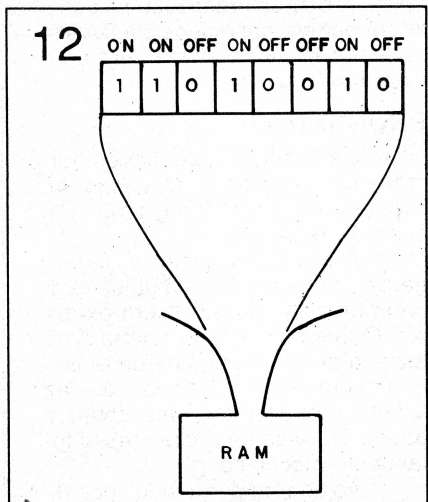
Cada byte de memória apresenta um único endereço. A CPU lê o conteúdo de um byte e efetua seu processamento.

A memória pode ser sub-dividida, conforme ilustra a **figura 13**. Isto é necessário porque vários bytes contêm dados que formarão a imagem. Outros bytes contêm instruções para gerenciar o video-game - este é o sistema operacional da máquina. Outros bytes contêm software de aplicações e normalmente há espaço livre para eventuais utilizações pela CPU.

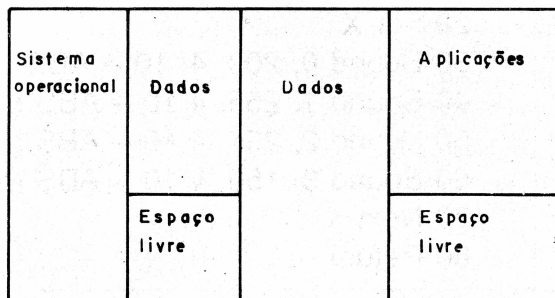
Agora que você já conhece os princípios básicos do software de um video-game, vamos passar para o estudo do seu hardware.

## INTRODUÇÃO AO HARDWARE DO VIDEO-GAME

No capítulo anterior, o leitor to-



## MAPA DA RAM



mou conhecimento do software, elemento que faz operar um sistema de computação, no caso do nosso tratado, um video-game.

Ao contrário da linguagem de alto nível e inteligente dos programas de software, o hardware opera com linguagem de baixo nível, ou seja, uma linguagem binária, que só possui dois algarismos: 0 e 1.

Mas, afinal, o que é hardware?

Hardware é o equipamento formado por circuitos integrados, interfaces, memórias, CPU, etc. Uma placa de circuito impresso com os seus componentes compõe um hardware.

Hardware e software estão interligados. Assim sendo, um video-game construído com um hardware da SEGA, só poderá rodar jogos cujo software seja compatível.

## DADOS

Todos os processos físicos podem ser caracterizados através de sinais de informações conhecidos por DADOS. Os dados são apresentados na forma de bits, podendo assumir o valor zero ou o valor um.

## PROGRAMA

Um conjunto de dados formam um programa. Um jogo de video-game é um programa, um software especial.

A **figura 14** ilustra a linguagem de máquina. Para melhor compreendermos, daremos um exemplo. Quando você tecla a letra A no teclado de uma máquina de datilografia, um martelo bate na fita e faz a letra aparecer em tinta sobre o papel. Trata-se de um processo totalmente mecânico.

Ao contrário, quando você di-

```

1000 10000011
01001101110001
10001100101001
01001101110011
01001100001011
11001110010001
10001101011011

```

gita a tecla A em um computador, sua ação gerará um conjunto de zeros e uns, que fazem a letra aparecer, de forma luminosa sob a tela. O processo é totalmente baseado na eletrônica digital. Estes zeros e uns utilizados para codificar uma letra, um número ou qualquer outro caractere, são padronizados.

Por exemplo, existe um padrão denominado ASCII (American Standard Code For Information Interchange) - Código padrão Norte-Americano para intercâmbio de informações. Este código (com alguns exemplos na **figura 15**) é bastante completo, pois ele fornece informações sobre se é uma letra maiúscula, letra minúscula, tipo de caractere, etc.

Para que o leitor possa dominar todos os conceitos teóricos sobre os videogames, faremos uma revisão dos conceitos básicos de eletrônica digital, baseado na álgebra booleana.

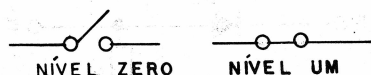
15

A - 01000001
a - 01100001
B - 01000010
F - 01000110
K - 01001011

## FUNÇÕES E PORTAS LÓGICAS

A eletrônica digital utiliza um sistema numérico de base dois (sistema binário). A **figura 16** ilustra as duas situações do sistema binário.

16



As informações binárias são representadas por níveis de tensão que podem ser de qualquer valor, desde que os sistemas a serem interligados sejam compatíveis.

Esse nível de tensão é chamado de nível lógico. Ou seja, o nível zero é representado por um certo valor de tensão, não necessa-

riamente zero volts.

O nível 1 possui outro valor bem distinto de tensão.

Na prática, é muito comum usarmos zero volts como nível lógico zero, e 5 volts (C.C.) como nível lógico 1. Veja a (**figura 17**).

Em casos particulares, os valores de tensão podem assumir outros valores.

Em capítulos anteriores observamos uma álgebra toda especial para eletrônica digital. Agora, veremos como é possível executar essas operações eletronicamente, através de circuitos que envolvem chaves, diodos transistores, e outros semicondutores.

## PORTA LÓGICA

Porta ou bloco lógico é o circuito que realiza uma operação ou função lógica.

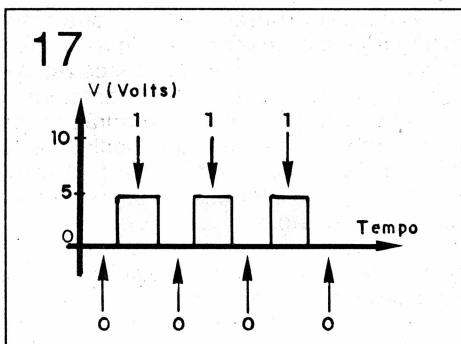
## FUNÇÃO E

A função E é igual a 1 somente quando todas as variáveis forem iguais a 1.

Veja figura 18 e 19.

## FUNÇÃO OU

A função ou é igual a 1 somente quando uma ou mais variáveis forem iguais a 1.



18

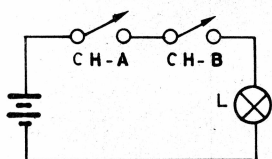


TABELA DE REPRESENTAÇÃO  
(TABELA VERDADE)

VARIÁVEIS		SAÍDA
Ch-B aberta	Ch-A aberta	L apagada
Ch-B aberta	Ch-A fechada	L apagada
Ch-B fechada	Ch-A aberta	L apagada
Ch-B fechada	Ch-A fechada	L acesa

Veja as figuras 20 e 21.

## FUNÇÃO NÃO

NÃO, NOT ou inversor são portas que invertem a condição existente na sua entrada.

A figura 22 mostra o circuito elétrico equivalente e a figura 23 o seu símbolo.

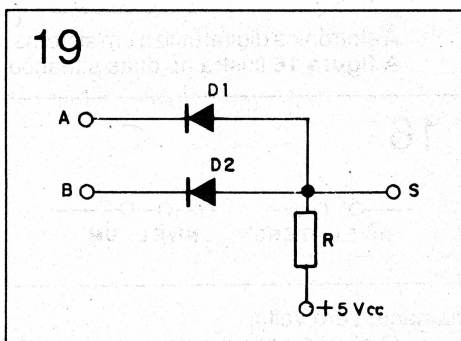
## FUNÇÃO NAND

Esta função (ou porta) é uma composição da função E com a função NÃO.

A figura 24 ilustra a tabela verdade e as figuras 25 e 26 os circuitos elétricos equivalentes.

## FUNÇÃO NOR

A porta NOR é uma composição da porta OU com a porta NÃO.



20

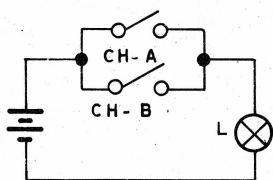
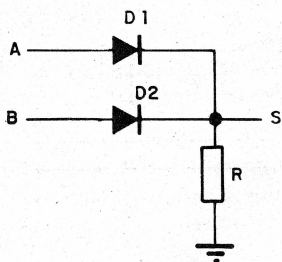
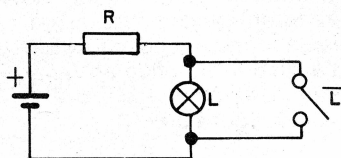


TABELA REPRESENTAÇÃO (TABELA VERDADE)		
VARIÁVEIS		SAÍDA
Ch-B aberta	Ch-A aberta	L apagada
Ch-B aberta	Ch-A fechada	L acesa
Ch-b fechada	Ch-A aberta	L acesa
Ch-B fechada	Ch-A fechada	L acesa

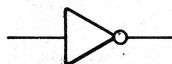
21



22



23



Veja figuras 27 e 28.

### PORTA OU EXCLUSIVA

A porta OU Exclusiva é implementada a partir das funções básicas E, OU, e NÃO. A figura 29 ilustra este teorema.

24

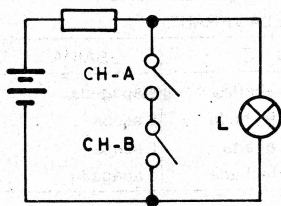
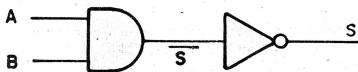
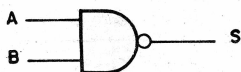


TABELA DE REPRESENTAÇÃO (TABELA VERDADE)		
VARIÁVEIS		SAÍDA
Ch-B aberta	Ch-A aberta	L acesa
Ch-B aberta	Ch-A fechada	L acesa
Ch-B fechada	Ch-A aberta	L acesa
Ch-B fechada	Ch-A fechada	L apagada

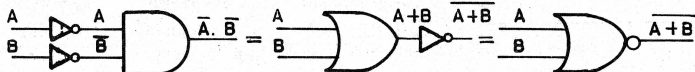
25



26



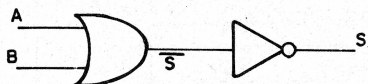
27



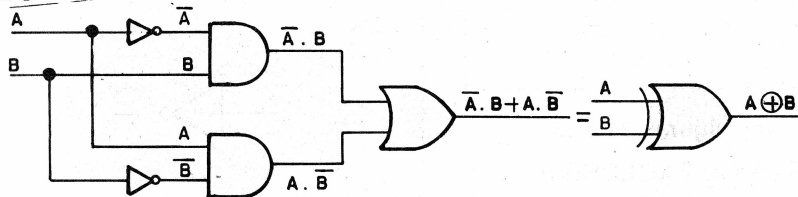
A função OU Exclusiva é igual a zero somente quando todas as variáveis forem iguais a zero ou um. Da mesma forma a função OU Exclusiva é igual a 1 quando as variáveis forem diferentes.

Veja a figura 30.

28



29



30

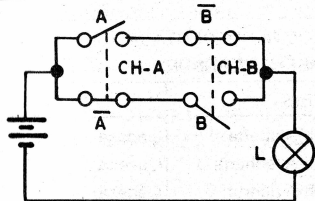
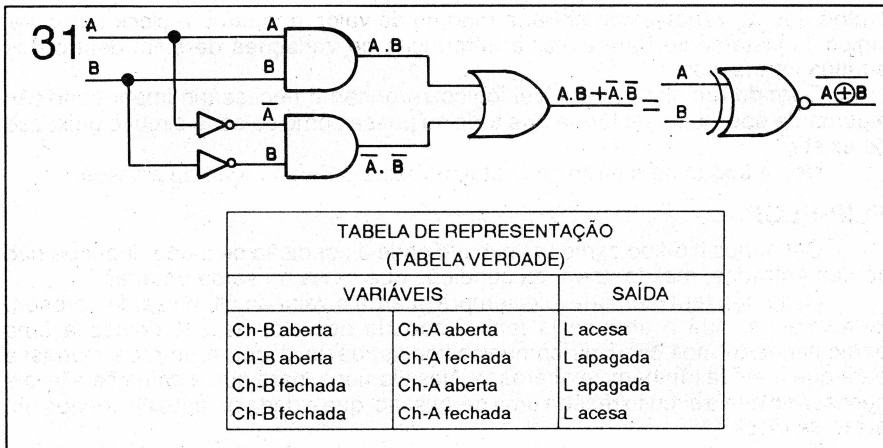


TABELA DE REPRESENTAÇÃO  
(TABELA VERDADE)

VARIÁVEIS		SAÍDA
Ch-B aberta	Ch-A aberta	L apagada
Ch-B aberta	Ch-A fechada	L acesa
Ch-B fechada	Ch-A aberta	L acesa
Ch-B fechada	Ch-A fechada	L apagada

## PORTA E COINCIDÊNCIA

Esta porta é o inverso da porta OU Exclusiva. Observe as **figuras 31 e 32**.



## CIRCUITOS DIGITAIS

Uma função lógica pode ser realizada por vários tipos de circuitos e tecnologias diferentes. Quando combinamos vários destes circuitos formamos os chamados circuitos seqüenciais.

## CIRCUITOS SEQÜENCIAIS

Circuitos seqüenciais são sistemas cujo comportamento, depende não só do valor atual das variáveis de entrada, mas também da forma de como estas variáveis se comportaram anteriormente.

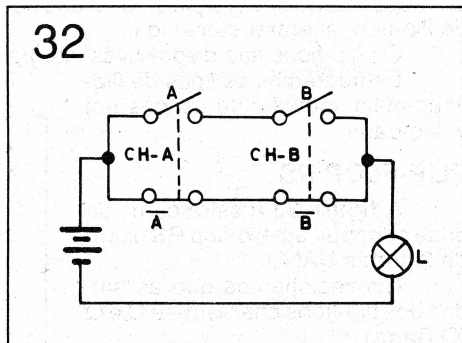
Consideremos um sub sistema seqüencial real. Determinado estado interno desse subsistema é caracterizado pelo valor de um conjunto de variáveis, denominadas variáveis internas de estado. Da mesma forma que as variáveis de entrada determinam o estado da entrada, uma combinação de valores das variáveis de estado, determina o estado interno.

Na prática existem duas grandes categorias de circuitos seqüenciais, os circuitos seqüenciais assíncronos e os síncronos.

Nos circuitos seqüenciais assíncronos o comportamento do circuito depende unicamente da propagação dos sinais elétricos, nos diversos atrasos extrínsecos, cujas saídas fornecem os valores das variáveis internas de estado.

Além dos atrasos extrínsecos (não pertencentes ao dispositivo), temos também atrasos intrínsecos (pertencentes ao dispositivo).

A presença de atrasos intrínsecos introduzem variáveis de atrasos espúrias que alteram o comportamento do circuito seqüencial.





Os circuitos seqüenciais síncronos, são circuitos seqüenciais cuja operação é comandada por um clock (relógio) externo. Em outras palavras, o sinal de relógio ou clock comanda a mudança de um estado interno a outro.

Este circuito síncrono é mais lento. Nos circuitos síncronos normalmente se proíbe que as variáveis de entrada mudem de valor, enquanto o clock tem nível lógico 1, justamente para evitar a introdução de variações de sinal dentro dos atrasos intrínsecos.

Quando um clock tiver nível lógico zero, não é necessário impor condição alguma de operação, de forma que tudo se passa como se esse circuito deixasse de existir.

Nesta seção de eletrônica digital, trabalha-se em função de atrasos.

## FLIP-FLOP

Definimos flip-flop como um circuito onde a condição de saída depende não só das entradas, mas também da condição que havia na saída anterior.

O flip-flop tem memória, ele sempre guarda o valor da última saída apresentada, mesmo que a entrada já tenha mudado de valor. Isso é devido a uma particularidade: nos circuitos comunicacionais, bastava que a entrada mudasse para que a saída também se alterasse. Nos flip-flops, para que a saída se altere é necessário que o circuito receba uma permissão, que é dada quando ele recebe um pulso de clock.

Esse pulso libera o flip-flop para que este possa verificar o nível lógico das entradas e depois, segundo uma função de transferência característica de cada tipo de flip-flop, alterar a sua saída.

Os flip-flops são dispositivos integrados.

Estudaremos os tipos de flip-flops mais comumente usados em video-game.

## FLIP-FLOP RS

A figura 33 mostra como se pode construir um flip-flop RS usando 2 portas NAND.

Convecionamos que as saídas dos flip-flops chamam-se Q e Q (Q Barra).

Notamos que a realimentação faz com que as saídas sejam injetadas juntamente com as variáveis de entrada, ficando claro que o estado que as saídas irão assumir depende das variáveis de entrada.

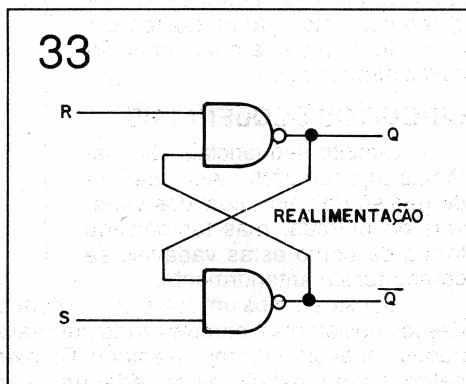


TABELA DA VERDADE

R	S	Q	$\bar{Q}$	
0	0	*	*	Não é permitido
0	1	1	0	
1	0	0	1	
1	1	1	1	Não muda

O flip-flop RS (Reset/Clear) são entradas assíncronas posto que a saída muda imediatamente após a mudança de qualquer um deles).

Na **figura 34** vemos o símbolo do flip-flop.

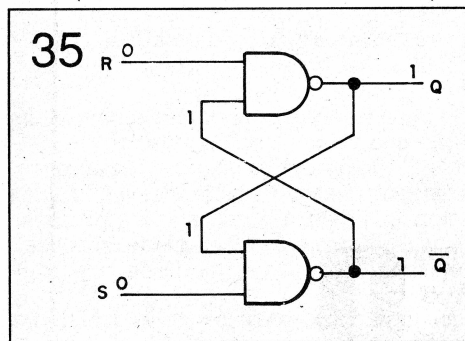
Uma particularidade é que a saída  $Q$  é sempre o complemento (inverso) de  $\bar{Q}$ .

Estudo do comportamento de RS.

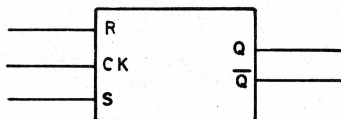
1º caso:  $R = S = 0$  veja a

(**figura 35**)

(lembrete: tabela da verdade da porta NAND)



**34**



A	B	S
0	0	1
0	1	1
1	0	1
1	1	0

Notamos que esse é um estado instável, pois  $Q$  forçosamente irá assumir valor 1. Este estado não é admissível em flip-flop. É um estado não permitido.

2º caso:  $R = 0$  e  $S = 1$  veja a (**figura 36**)

3º caso:  $R = 1$  e  $S = 0$  veja a (**figura 37**)

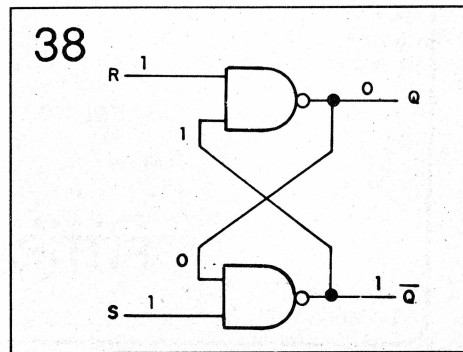
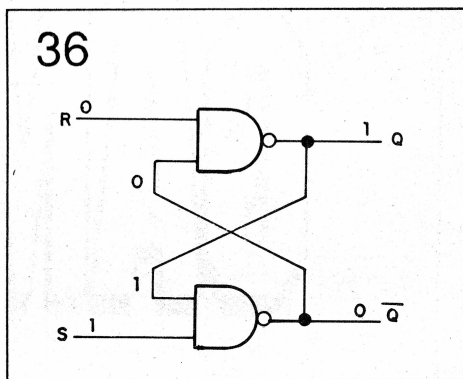
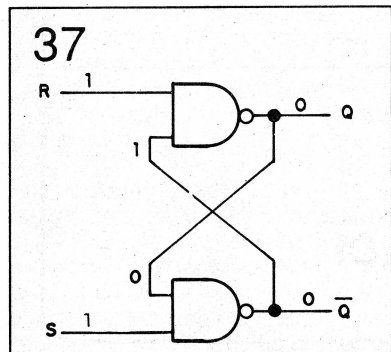
4º caso:  $R = S = 1$  veja (**figura 38**)

Podemos notar que este estado é estável, logo o valor que a saída  $Q$  irá assumir será igual ao seu valor atual. Portanto, não mudou, não se alterou.

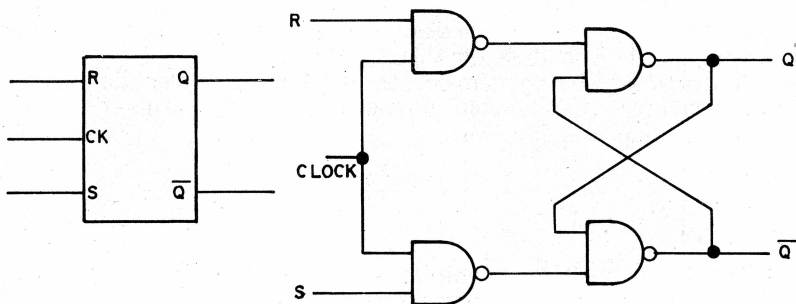
## FLIP-FLOP RS COM CLOCK

Na **figura 39** vemos este circuito RS comandado por clock.

Neste circuito iremos notar que quando a entrada do clock for igual a zero, o flip-flop irá permanecer no seu estado, mesmo que variem as



# 39



entradas R e S. Isso pode ser confirmado analisando-se o circuito, onde concluímos que para o clock igual a zero, as saídas serão sempre iguais a um.

Esse circuito irá mudar de estado apenas quando o clock for igual a 1.

## FLIP-FLOP JK

Na figura 40 podemos ver como construir um flip-flop JK utilizando 2 portas NAND de 3 entradas e 2 NAND de 2 entradas. Esse flip-flop veio solucionar o problema do RS onde existia a situação que  $R = S = 1$  e produzia um estado irrelevante.

O flip-flop se comporta como um RS com exceção ao caso em que  $JK = 1$ . Neste caso, a saída é complementada.

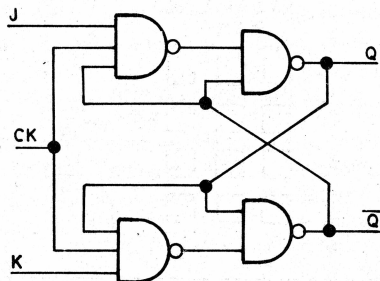
## FLIP-FLOP COM CLOCK, PRESET E CLEAR

Na figura 41 vemos um JK com entradas Preset e Clear, além do clock. A figura 42 ilustra a tabela verdade.

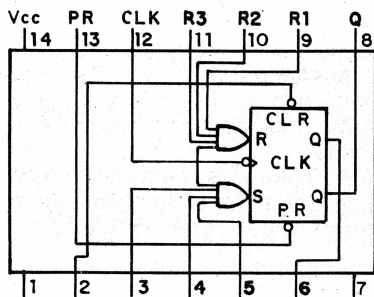
## FLIP-FLOP TIPO D





Esse é um JK com as entradas invertidas. O termo D (DELAY) se refere ao atraso que o sinal possui entre entrada e saída. Este FF simplesmente transfere o valor da

# 40



# 41



ENTRADAS					SAÍDAS	
PR	CLR	CLK	J	K	Q	$\bar{Q}$
0	1	X	1	1	1	0
1	0	X	1	1	0	1
0	0	X	1	1	*	*
1	1		0	0	Q0	$\bar{Q}0$
1	1		1	0	1	0
1	1		0	1	0	1
1	1		1	1	0	1

entrada para a saída sempre que houver um pulso de clock. Veja **figura 43**.

## REGISTRADORES DE DESLIGAMENTO

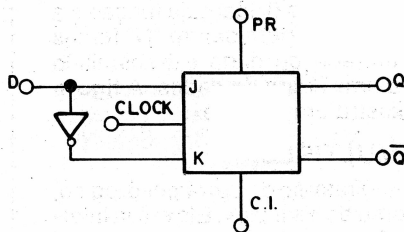
Vimos que os flip-flops armazenam 1 bit de informação. Nos sistemas digitais, porém, uma informação é formada por vários bits (4, 8, 16, etc).

Para possibilitar o armazenamento de 4 bits, foi desenvolvido o registrador de deslocamento. O circuito é formado por 4 flip-flops ligados em cascata. Considerando que cada flip-flop fornece 1 bit, teremos 4 bits que se deslocarão a cada novo pulso de clock.

As entradas do clock devem ser interligadas, e a entrada de dados do segundo fica ligada a saída do primeiro. E assim, sucessivamente.

A **figura 44** ilustre um registrador de deslocamento.

43



## CODIFICADORES

São circuitos lógicos que geram com código binário em outras informações.

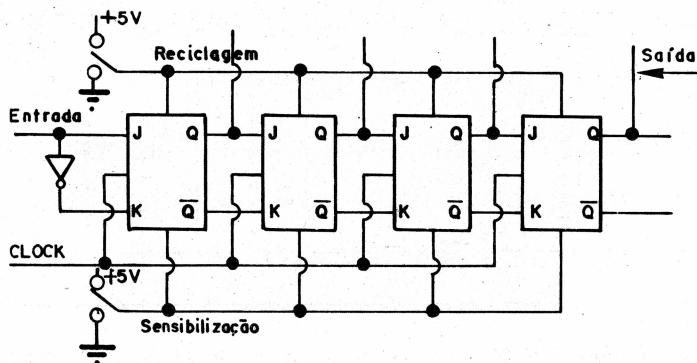
## DECODIFICADORES

São circuitos, lógicos que recuperam uma informação - binária.

## CONTADORES

Contadores são circuitos lógicos seqüenciais, cuja saída depende dos vários estados de entrada aplicados ao circuito. São implementados com flip-flops e registradores de deslocamento.

São amplamente utilizados para contagens, geração de códigos binários, geradores de forma de onda, conversão A/D, etc.



## MULTIPLEX

Circuito digital cuja função é a de selecionar um dentre "N" fontes de entrada de dado e transmiti-lo para uma saída de dados. A figura 45 ilustra um multiplex.

## DEMULTIPLEX

Trata-se de um circuito lógico, inverso do multiplex. Ele envia informações contidas em uma única entrada para diversos canais de saída de dados.

A figura 46 ilustra o bloco de um circuito DEMULTIPLEX.

## CONVERSOR D/A

Trata-se de um dispositivo que converte pulsos digitais em sinal analógico (tensão ou corrente).

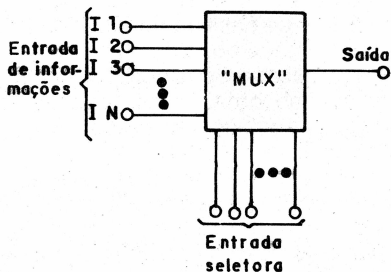
A figura 47 apresenta um diagrama em blocos de um D/A.

## CONVERSOR A/D

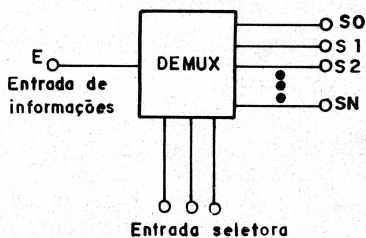
São circuitos que examinarão uma entrada analógica (tensão ou corrente) para então convertê-la em uma palavra binária equivalente.

O princípio básico do conversor A/D é a integral realizada com

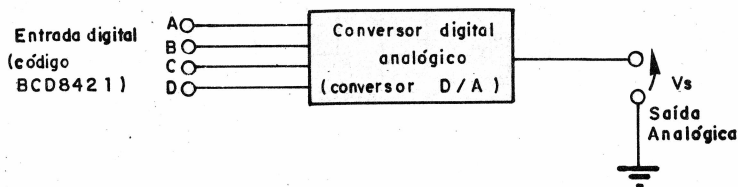
45



46

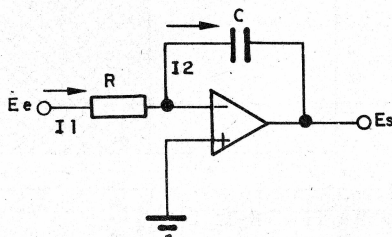


47



amplificador operacional veja a (figura 48).

48



## FUNCIONAMENTO DO VIDEO-GAME (DIAGRAMAS EM BLOCOS)

Cada um dos estágios de um **VIDEO-GAME** executa uma determinada tarefa de maneira específica. De todas as funções executadas por um **VIDEO-GAME**, a geração das imagens em formato digital é a mais importante. O **VIDEO-GAME** deve gerar um sinal de vídeo composto (incluindo áudio) de tal forma que se equipare a uma emissora de TV.

### GERAÇÃO DA IMAGEM

Para acompanhar passo a passo este assunto, tomaremos por base um receptor de TV PAL-M com frequência vertical de 60 Hz e horizontal de 15.750 Hz. Vamos adotar a seguinte convenção:

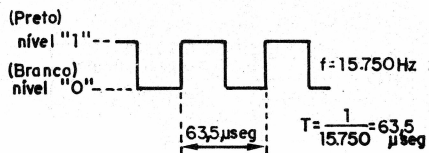
- nível de preto: nível alto (1)
- nível de branco: nível baixo (0)

Na **figura 49** vemos o sinal de uma linha horizontal, cujo período é de 63,5 microssegundos. Observe agora pela **figura 50** que na tela corresponde à metade do traçado em preto e a outra metade em branco.

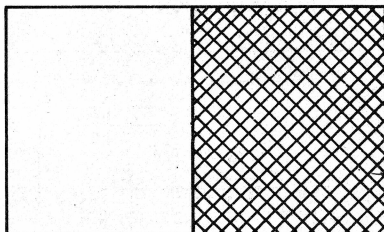
Se dobrarmos a frequência horizontal (31.500), dobraremos também o número de barras pretas e brancas na tela. Veja **figura 51**.

Utilizando o mesmo raciocínio, vejamos o que ocorre com a varredura vertical. Veja **figura 52**.

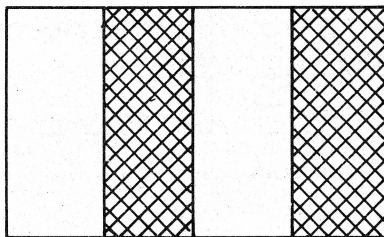
49



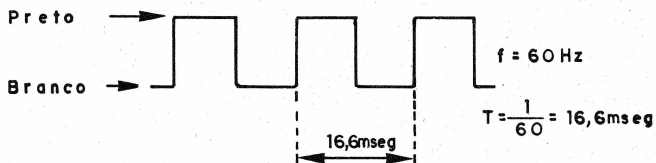
50



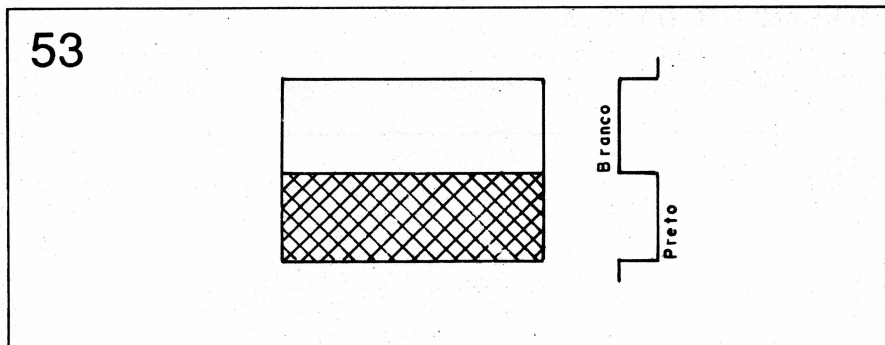
51



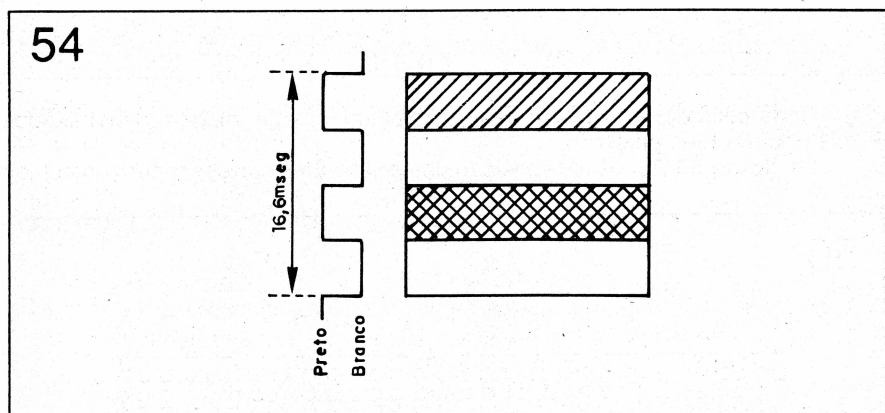
52



Observe que esta frequência vertical de 60 Hz equivale a um período de 16,6 milissegundos. Quando um sinal do formato de uma onda quadrada de 60 Hz é injetado no amplificador de vídeo, durante a primeira metade do ciclo (ou seja, 8,3 mS) ele estará em nível 1, reproduzindo na tela o preto até a metade da varredura vertical, e durante a segunda metade do ciclo (os 8,3 mS restantes), ele estará em nível 0, produzindo na tela o branco. Observe a **figura 53**.



Quando dobramos essa frequência (120 Hz), o resultado será a formação de duas barras pretas e duas barras brancas na tela veja a **figura 54**.



A tabela abaixo contém os principais dados da relação vertical/horizontal.

FREQÜÊNCIA (Hz)	BARRAS PRETAS	BARRAS BRANCAS	SENTIDO
60	1	1	H
120	2	2	H
240	4	4	H
480	8	8	H
960	16	16	H
15.750	1	1	V
31.500	2	2	V
63.000	4	4	V
126.000	8	8	V
252.000	16	16	V
504.000	32	32	V
1.008.000	64	64	V

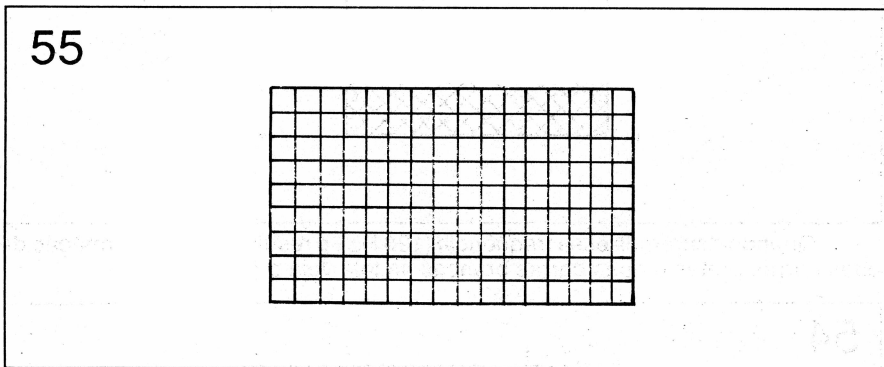


Agora, podemos misturar as frequências vertical e horizontal e produzir barras horizontais e verticais simultaneamente. Para isto, torna-se necessário que estes sinais derivem de um mesmo oscilador, denominado clock, e através de sucessivas divisões, selecionar os sinais que nos interessem.

A escala de cinzas do padrão de vídeo é formada pela soma dos sinais H e V, mas antes eles são transformados de tal forma que um deles tenha a metade da amplitude do seguinte (isto é feito através de resistores).

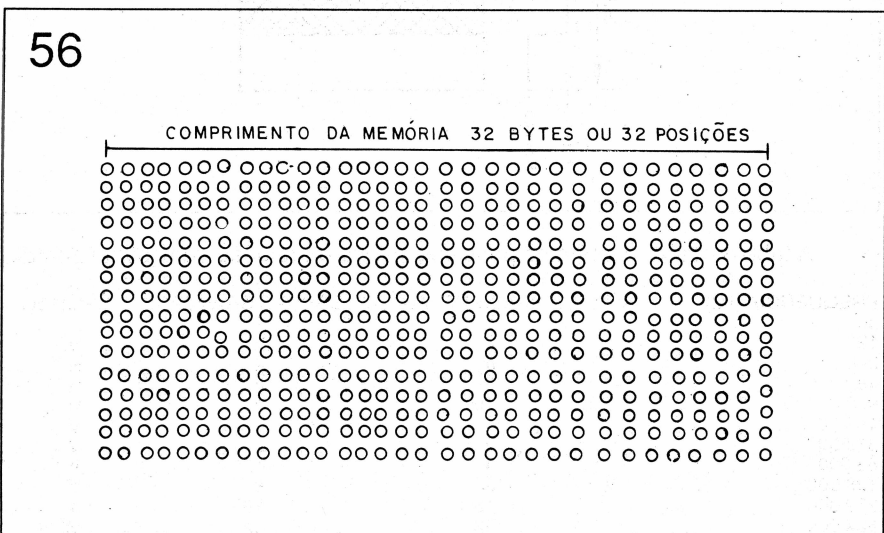
### MAPEAMENTO DA TELA

O **VIDEO-GAME** trata a tela do TV como uma matriz de colunas e linhas, formando quadrículas denominadas "células", como ilustra a **figura 55**.



Cada célula é dividida em pequenos elementos de imagem denominados Pixels (Picture Elements).

Na **figura 56** vemos a disposição dos pixels. Eles correspondem ao estado



1/0 ou colorido/apagado. Quanto maior for o número de pixels, melhor será a resolução, o que significa uma imagem com mais detalhes.

Todas as células e todos os pixels são gravados numa ROM especial chamada "gerador gráfico". O conteúdo da ROM é o conjunto de pixels da imagem colorida.

## DIAGRAMA EM BLOCOS

Na **figura 57** vemos um diagrama em blocos do estágio gerador de imagens.

São três as condições de se inicializar este circuito: clock, cartucho e operações do joystick.

Inicialmente o microprocessador assume o controle e escreve na RAM os endereços das células e pixels.

O microprocessador junto com um multiplex de endereçamento realizará uma contagem seqüencial de pulsos para fornecer o endereço do vídeo para a memória RAM.

Os sinais necessários são gerados a partir de um clock e obtidos através de divisores lógicos, a saber:

- divisor para contador de fileiras
- divisor para contador de linhas de varredura
- divisor para contador de linha de segmentos
- divisor para contador de linha de célula
- divisor para contagem de quadros (sincronismo vertical)

A memória RAM de vídeo utiliza o princípio de multiplexação de endereços para diminuir o tamanho físico. Por exemplo, seriam necessários os 14 bits ou pinos de endereços para acessar 16 K de células. Com a multiplexação, necessitamos apenas de 7 pinos de endereço.

Os sinais de sincronismo vertical e horizontal é obtido a partir de contadores que são acionados pelos divisores de fileira de pixels, de linhas de varredura e de linhas de segmentos. O próprio microprocessador recebe o sinal do sincronismo vertical de tal forma que ele já gera os sinais para a RAM de vídeo com o padrão de varredura definido.

A **figura 58** ilustra o diagrama de sinais e tempos do gerador de sincronismo.

Quem inicia a geração de sinais de vídeo é o microprocessador. Posteriormente, o controle passa ao bloco gerador de sincronismo, que aciona seu contador de 64 endereços que apresenta a RAM de vídeo com o endereço inicial zero, onde o microprocessador escreveu o pixel na parte superior esquerda da tela.

A RAM de vídeo envia à ROM (gerador gráfico) o conteúdo da posição endereçada.

Esta ROM carrega o registrador de deslocamento que produzirá um clock de Start (partida) para o mixer de vídeo.

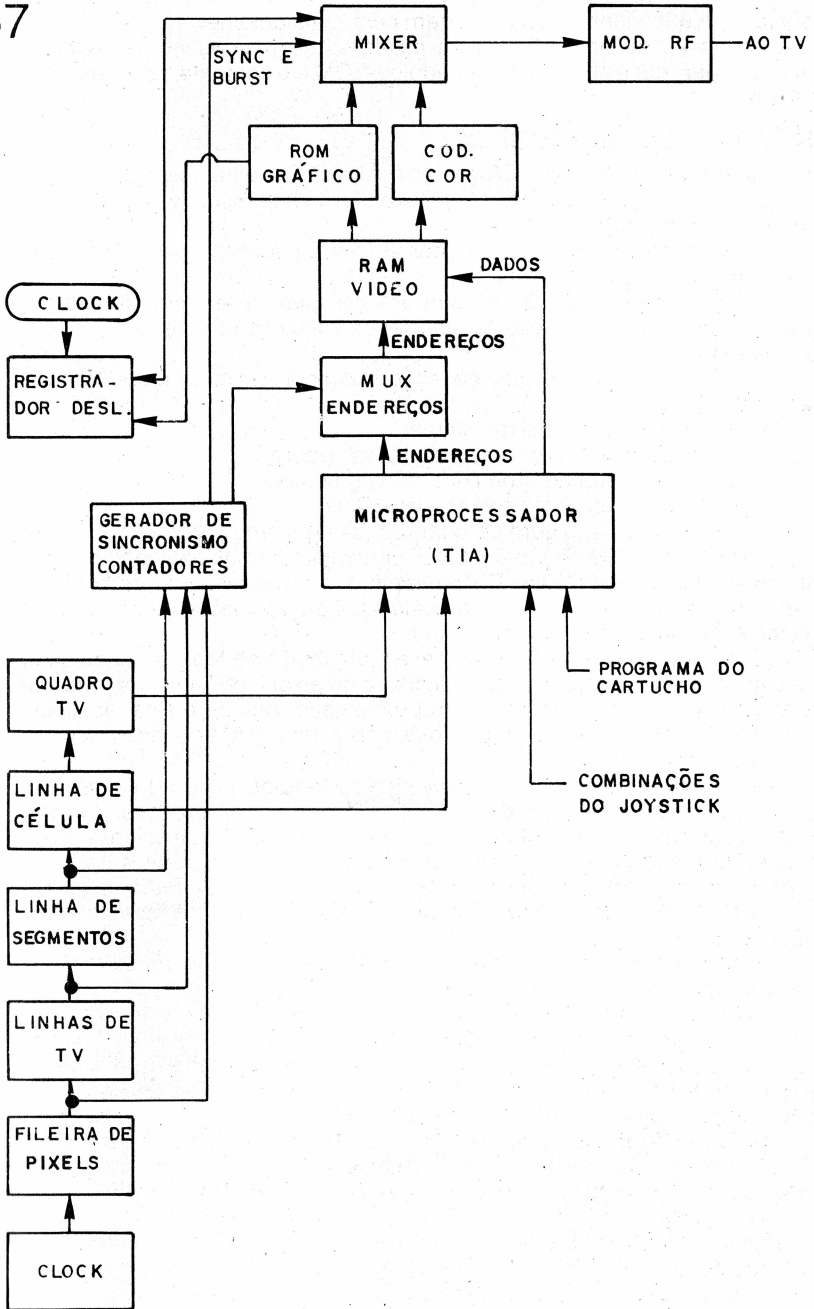
O mixer de vídeo recebe este bit do registrador de deslocamento e mistura-o com os sinais de sincronismo, sinais do codificador de cor e os sinais de vídeo formados pela ROM, formando o sinal de vídeo composto que seguirá até o modulador de RF, onde modulará uma portadora (chaveada para canal 2 ou canal 3, permitindo injetar na entrada de antena do TV).

A operação acima se repete 64 vezes, até completar a apresentação na tela.

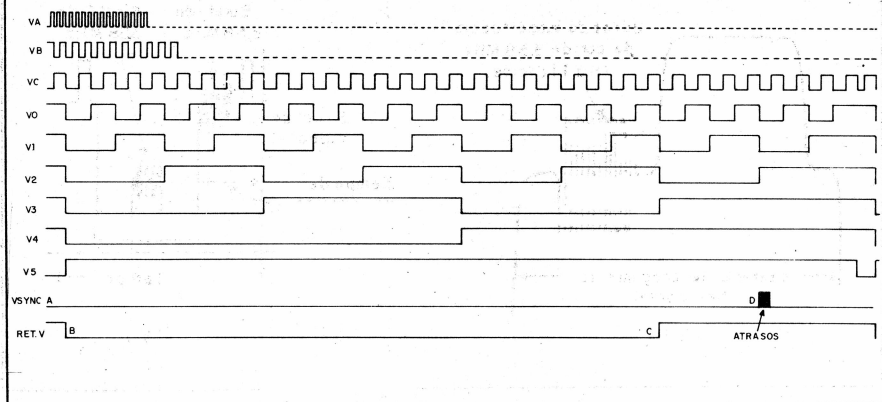
Cabe ainda ao gerador de sincronismo resetar o contador de endereços da RAM, fazendo-a começar pelo endereço zero.

O microprocessador lê o programa do cartucho, lê as combinações de bits fornecidas pelo joystick e lê também os bits de sincronismo fornecidos pelos divisores de quadro (vertical) e linhas (horizontal).

O quadro completo sinalizado por completo tem um mapeamento de 1024 células (64 x 16). Ao ligar o **VIDEO-GAME** com o cartucho instalado, antes de



# 58



apertar o botão "início", temos uma cena estática, pois não estão havendo combinações no microprocessador e o clock do programa não foi acionado.

Todos estes blocos da figura 57 formam o estágio TIA - interface adaptadora de televisão.

## CODIFICAÇÃO DE COR

Em TVC, a informação das cores é denominada crominância, composta dos sinais:

- R - Y
- B - Y
- Burst
- Sub-portadora suprimida de 3,58 MHz

O sinal R - Y contém o matiz vermelho.

O sinal B - Y contém o matiz azul. A combinação R - Y com B - Y resulta no matiz verde.

Todas as cores do espectro visível são obtidas a partir de valores das misturas dos sinais R - Y e B - Y.

**BURST - A figura 59** mostra os detalhes do burst (a) e da sub-portadora de 3,58 MHz (b).

O burst sincroniza a fase do oscilador de cor de 3,58 MHz. O sinal de 3,58 MHz gerado pelo cristal no TV é comparado com a fase do burst.

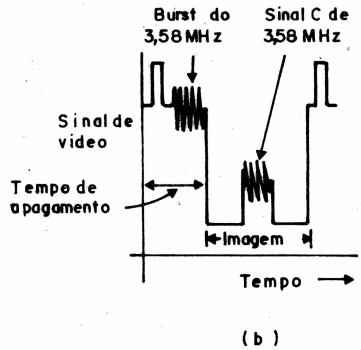
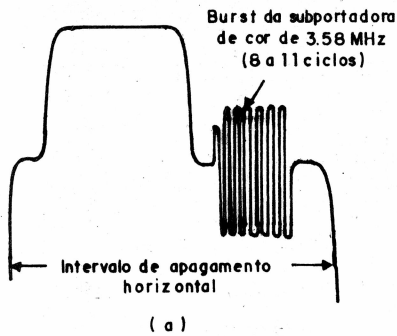
A fase de tensão do oscilador de 3,58 MHz determina os matizes. Portanto, o sincronismo de cor é necessário para estabelecer os matizes corretos para os demoduladores.

O burst tem 8 a 10 ciclos da sub-portadora de 3,58 MHz, transmitidos no pórtyco posterior de cada pulso de apagamento horizontal.

O burst e o sinal da sub-portadora são de 3,58 MHz. Contudo, o burst está presente no tempo de apagamento quando não há informação de imagem. Já o sinal da sub-portadora está presente durante a informação de imagem.

A ausência do burst implica em uma imagem em preto e branco.

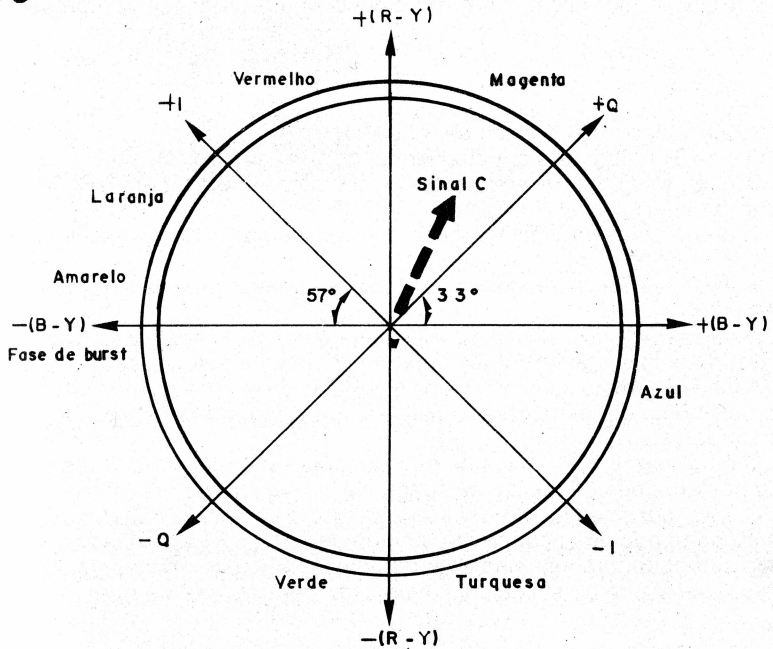
59



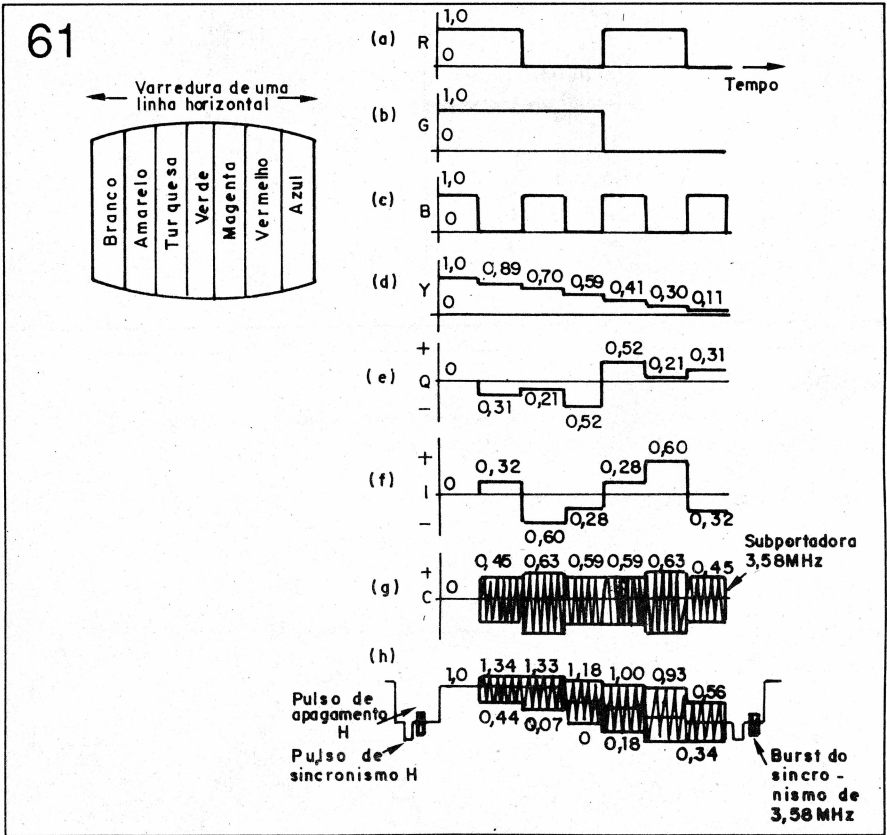
### ÂNGULOS DE FASE

A informação de cor é modulada em amplitude em duas fases diferentes, produzindo os sinais I (B - Y) e Q (R - Y).

60



A figura 60 ilustra o círculo de cores com seus matizes de diversos ângulos. O burst é a referência deste eixo e, portanto, possui o valor de  $180^\circ$  em NTSC (os ângulos são contados no sentido inverso aos ponteiros do relógio). Veja na figura 61 a construção do sinal composto de vídeo com os sinais R, G, B, Y, Q, I, C.



A cromaticidade é representada pela simbologia da figura 62.

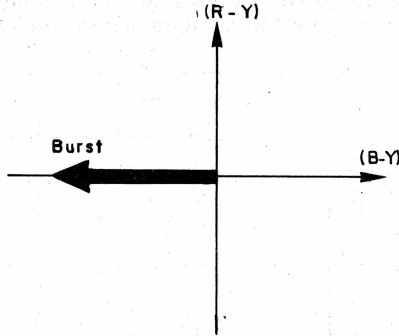
## SISTEMA PAL-M

O sistema PAL-M foi desenvolvido para corrigir as deficiências do NTSC. A diferença substancial reside no burst.

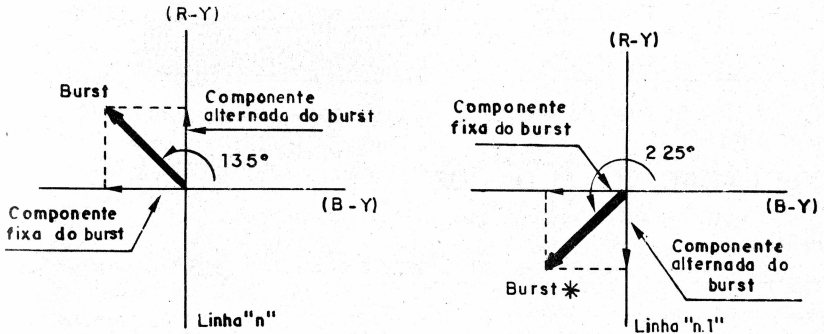
O burst PAL-M é emitido em um ângulo de  $135^\circ$  numa linha e  $225^\circ$  na linha seguinte, alternadamente com uma defasagem de  $90^\circ$ . Veja figura 63.

Devido às inversões linha a linha da componente R - Y, a transmissão do sinal burst no sistema PAL possui uma função a mais em relação ao sistema NTSC, ou seja, além de sincronizar o oscilador a cristal de 3,58 MHz, o burst tem a incumbência de sincronizar o demodulador PAL, informando qual linha foi invertida e qual não foi.

62



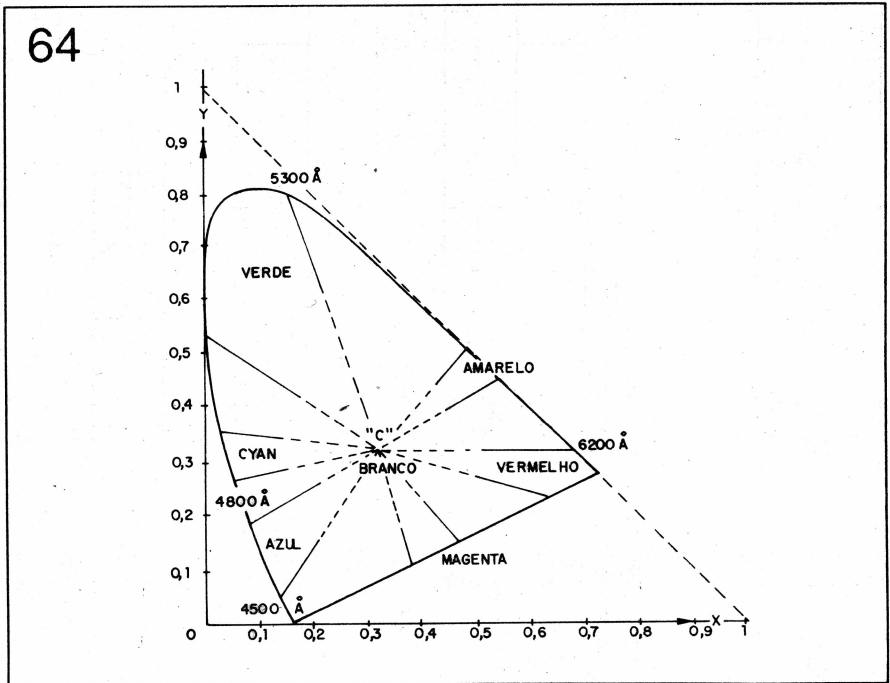
63



Isso caracteriza o sistema PAL por inverter linha a linha a sub-portadora R-Y acompanhada pela inversão linha a linha do burst, ora  $130^\circ$  ( $180/90$ ), ora  $225^\circ$  ( $180/270$ ).

Em outras palavras, em PAL-M ocorre uma defasagem de  $45^\circ$ , ora negativo, ora positivo, do Burst original.

A título de ilustração, apresentamos na **figura 64** o diagrama de cromaticidade, onde observamos as cores principais em relação a seus matizes.



A **figura 65** ilustra o sinal de barras de cores com os respectivos sinais de sincronismo e burst.

## CODIFICAÇÃO DIGITAL DA COR

Conforme acabamos de ver, a cor é codificada como defasagem do sinal de 3,58 MHz.

Uma maneira de se fazer isto é fazer a portadora de 3,58 MHz atravessar portas lógicas, onde ele sofrerá atrasos de 40 nS em cada gate. Selecionando-se o número de gates que o sinal deve atravessar, obtém-se a defasagem desejada. Veja **figura 66**.

Temos na **figura 66** a codificação para NTSC. O burst foi retirado do segundo gate com um atraso de 80 nanosegundos. O vermelho foi retirado do quarto gate com 160 nanosegundos de atraso em relação aos 3,58 MHz e 80 nS em relação ao burst.

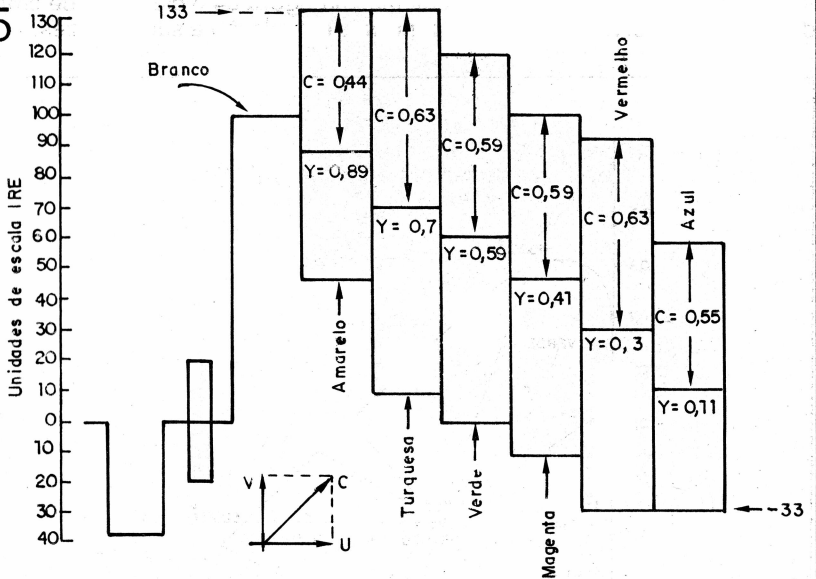
O azul é retirado do sétimo gate com 280 nS da sub-portadora e 200 nS em relação ao burst.

O verde é retirado do primeiro gate com um atraso de 40 nS em relação a sub-portadora de 3,58 MHz e menos 40 nS em relação ao burst.

Pelo fato de realizarem todas estas operações em velocidade relâmpago, pode parecer ao leitor que a compreensão destes conceitos seja complicada. Todavia, sua estrutura é simples.

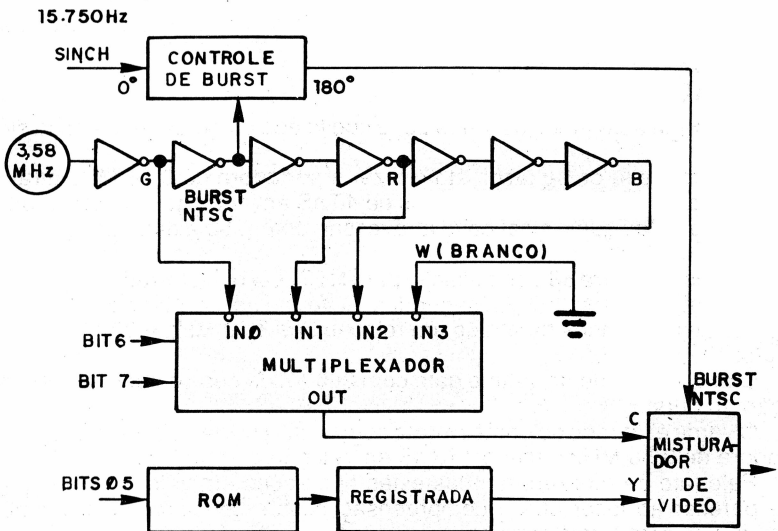


65



Todas as diversas partes de um **VIDEO-GAME** seguem um modo de operação chamado processamento serial. Ele executa todas as funções lógicas

66

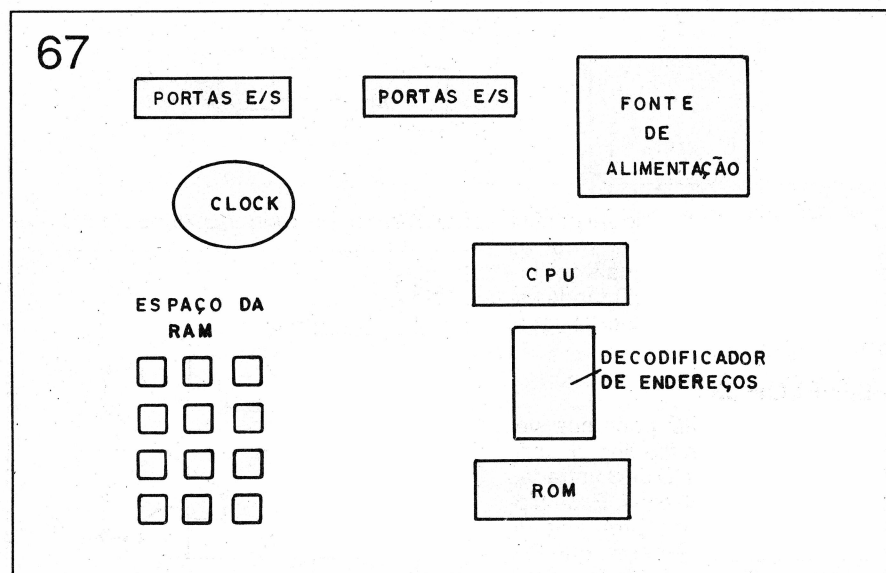


passo a passo, examinando cada instrução por vez e agindo sobre ela, antes de passar à instrução seguinte.

## UMA VISÃO ABRANGENTE

Apresentaremos uma série de diagramas em blocos na intenção de ilustrar em detalhes o funcionamento de um VIDEO-GAME.

Na **figura 67** vemos o primeiro desses diagramas. Esta placa contém a CPU - Unidade Central de Processamento, um microprocessador que gerencia todas as operações lógicas do VIDEO-GAME.



A CPU depende do clock (relógio). É um gerador de pulsos sincronizado por um cristal de quartzo que coordena o ritmo de trabalho dos circuitos lógicos.

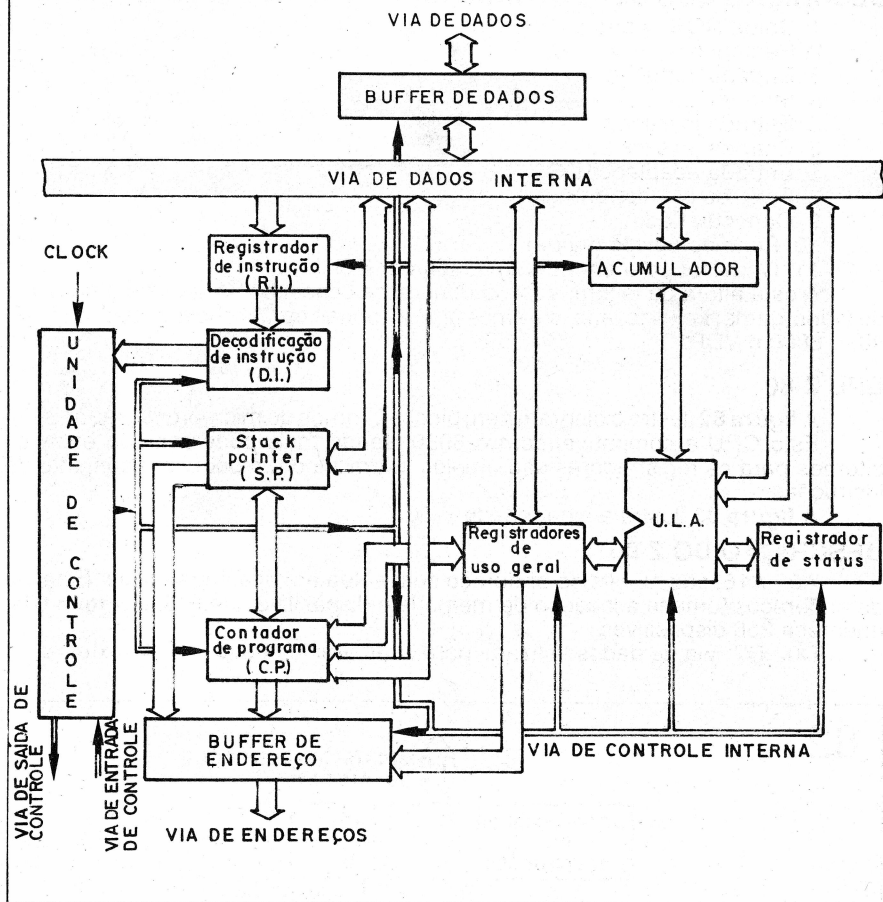
A placa do sistema possui partes E/S (entrada/saída) para interligar todos os circuitos (CPU, RAM, ROM, etc).

A ROM (Read only memory) é a memória somente de leitura que no VIDEO-GAME corresponde ao cartucho. De fato, cada cartucho é uma memória ROM, cujo conteúdo é o software de um jogo.

A RAM (random access memory) é a memória de acesso aleatório que é a memória disponível para leitura e gravação de dados durante a execução da CPU. A RAM armazena programas e dados apenas enquanto o VIDEO-GAME estiver operando.

As instruções saem da CPU como uma série de pulsos elétricos codificados, para encontrar um determinado endereço. Posteriormente, a informação retorna sob forma de pulsos decodificados, para que a CPU possa realizar o processamento.

A **figura 68** mostra outro diagrama em blocos com uma configuração completa de um sistema de dados em um VIDEO-GAME (trata-se de um diagrama didático).

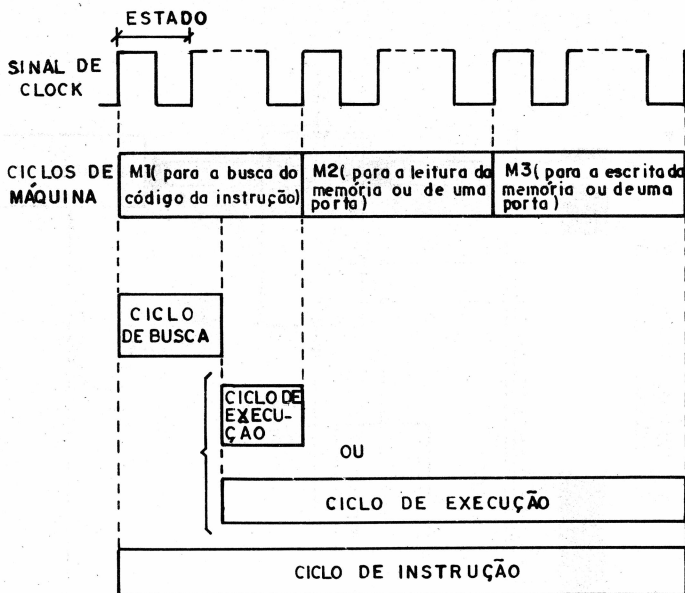


Quando é acionado o sistema, o clock começa a produzir sinais de frequências para a CPU. Isto fará com que a CPU comece a "ler" seu programa, gravado em seu registrador de instruções.

Se houver um cartucho colocado no compartimento da ROM, a CPU passa a ler e processar estas informações. Não havendo nenhum cartucho neste compartimento, a CPU passa a executar um programa inicial, que está armazenado em seu acumulador interno, denominado bootstrapping (abreviado por Boot).

Este programa é um conjunto de instruções que contém dois ciclos básicos, que são: ciclo de busca e ciclo de execução. A figura 69 ilustra estes ciclos. Este diagrama da figura 69 é conhecido por timing do microprocessador.

As instruções que manipulam informações utilizam-se de um ciclo de máquina. Cada ciclo de máquina possui certa quantidade de estados (Status) ou períodos de clocks. O estado é a menor quantidade de tempo de uma CPU.



A medida que o clock emite pulsos, os circuitos dentro da ROM localizam o endereço no CI seleccionado. Os bits são fornecidos em uma cadeia serial que pode ser de 4,8 ou 16 bits.

A CPU lê as informações da ROM mas deve esperar por um sinal especial de controle, que é fornecido pelo próprio cartucho (ROM).

A cada novo ciclo, a CPU transmite o sinal de leitura para o chip da ROM que, instantaneamente, transfere o conteúdo para a via de dados.

Da via de dados, os bits seguem até as portas E/S onde serão interfaceadas fora da CPU.

A interface E/S é o dispositivo que cuida da comunicação entre a CPU e os periféricos (os componentes exteriores à CPU). As portas E/S tornam acessíveis à CPU as informações contidas nos periféricos e vice-versa.

A comunicação pode ser de dois tipos:

Paralela e Serial.

## PARALELA

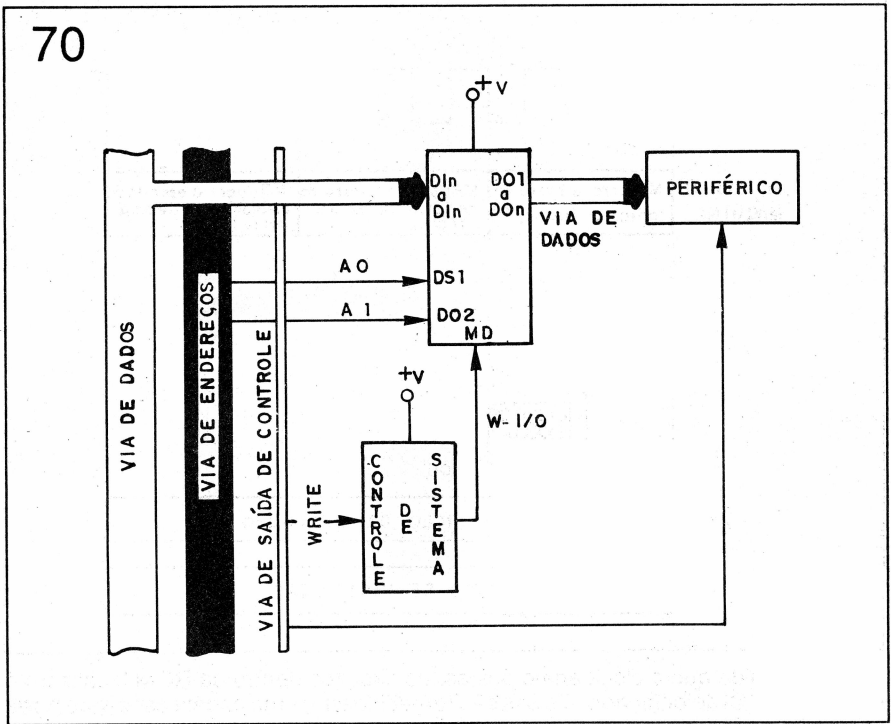
Na comunicação de dados paralela os dados são transmitidos ou recebidos de uma só vez pela CPU.

Uma das portas E/S paralela mais conhecida é a PPA (Adaptador Periférico Paralelo). A figura 70 ilustra este PPA.

Uma vez que o dispositivo foi seleccionado, dá-se o clock aos latches, levando-se os dados contidos na entrada para a saída destes.

Existem ainda outros tipos de portas mais versáteis, que são controladas por programas, dentre as quais podemos citar a PIA (Adaptador Periférico de Interface), a PIO (Paralela de Entrada e Saída) e a PPI (Interface Periférica Programável).

A comunicação destes periféricos com a CPU é realizada através da via de dados.



## SERIAL

Outro tipo de comunicação (interface) é o sistema serial programado. Neste caso, a transferência de dados se processa em série, através de uma única linha de dados.

Podemos dividi-los em dois blocos. Os de comunicação Assíncrona e os de comunicação Síncrona.

A comunicação Assíncrona é aquela onde não são transmitidos sinais de clock, porém existe um protocolo determinando o início dos dados.

A comunicação Síncrona utiliza um clock para sincronizar a transmissão dos dados.

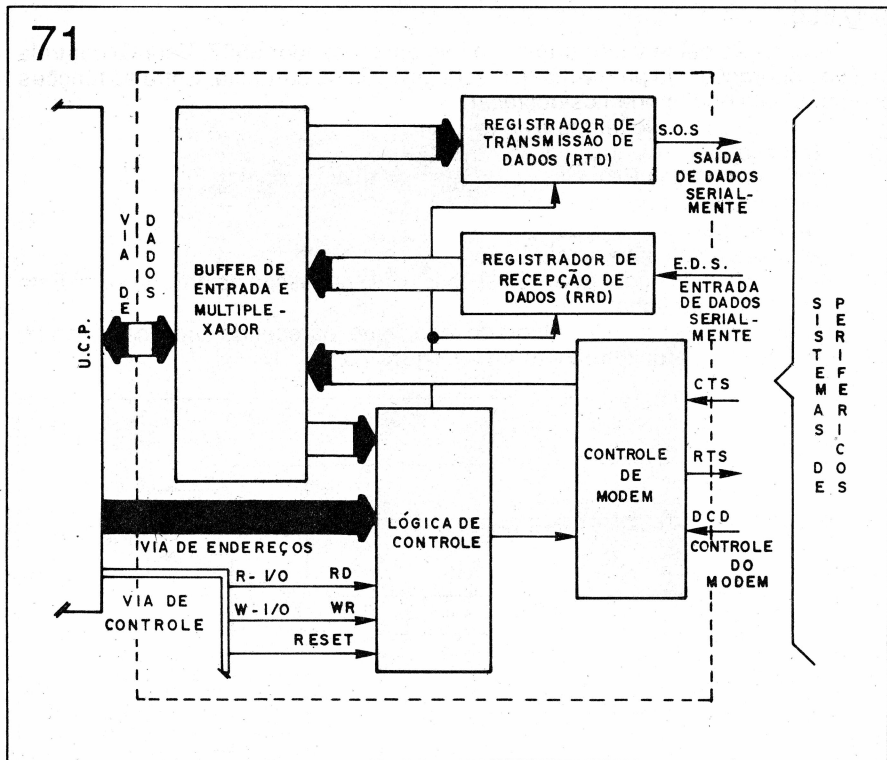
A **figura 71** ilustra o diagrama em blocos de um sistema de transmissão Serial.

Agora que já conhecemos os fundamentos da arquitetura de um vídeo-game, vamos estudar detalhadamente os modelos mais famosos do mercado.

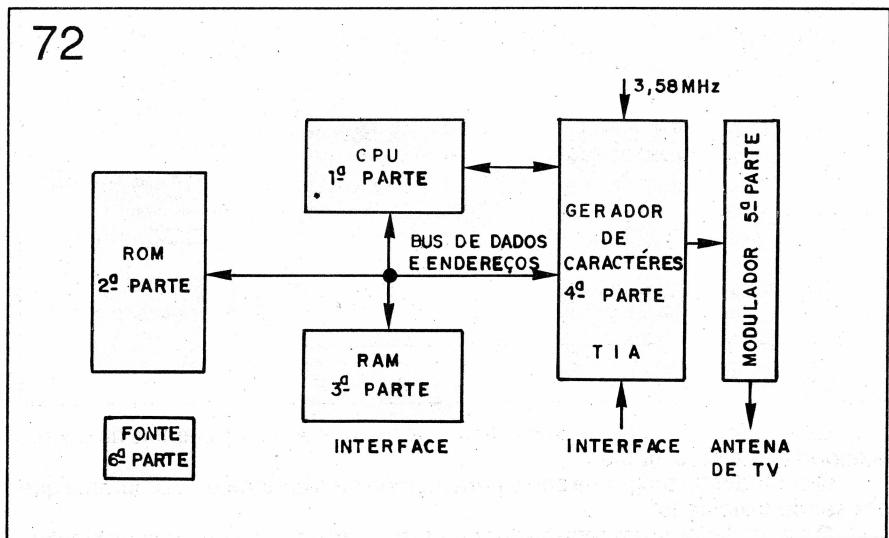
## DESCRIÇÃO DO ATARI

O vídeo-game Atari (comercializado no Brasil por várias marcas), utiliza a configuração da **figura 72**. Como o leitor pode observar, utiliza uma pequena quantidade de circuitos integrados. O entendimento deste modelo permitirá ao leitor compreender outros modelos mais complexos.

71



72



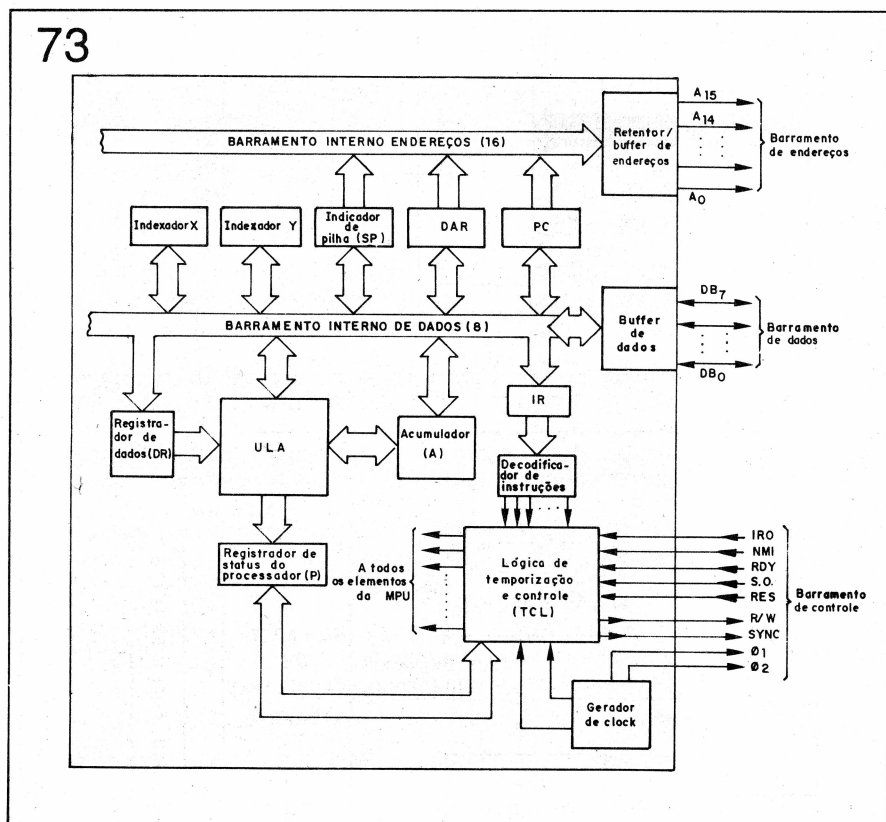
## CPU 6507

O coração deste vídeo-game é o microprocessador 6507. Este CI controla todas as operações do programa que está gravado no cartucho. Entre as funções exercidas pelo 6507 podemos destacar:

- fornece sinais de temporização
- fornece todos os sinais de controle
- transfere dados E/S
- decodifica instruções
- efetua operação lógicas
- executa o software do cartucho

Apesar de todo poder que a CPU possui, ela necessita de um software. Este software está no cartucho.

Para que este circuito integrado não fique parecendo uma caixa preta, apresentamos sua arquitetura interna na **figura 73**.



Existem vários registradores internos acoplados aos barramentos de dados, de endereços e de controle.

O contador de programa serve para memorizar o endereço do programa que está sendo executado.

O acumulador é um registrador que toma parte na maioria das operações

executadas pela ULA (Unidade Lógica e Aritmética), sendo que o acumulador se comunica com a seção de entrada e saída de dados.

## **PINAGEM DO 6507**

1. Reset - Reseta o programa, zerando o conteúdo interno.
2. Terra
3. RDY (Ready, pronto). Neste pino é recebido um sinal que informa que está apto para receber novas instruções.
4. VCC - pino de alimentação (5 V).
- 5 a 17. Via de endereço. Através destes pinos, o micro endereça as memórias, a PIA e a TIA, onde serão lidos ou escritos os dados de operação.
- 18 a 25. Via de dados - Conduzem todas as informações para os integrados da TIA, PIA e cartucho.

Para se ter uma idéia melhor sobre as funções das vias de dados e de endereços, oriente-se pela figura 73. A via de dados é bidirecional. Ela tanto pode levar, quanto trazer informações externamente. Estas informações são processadas entre a CPU e os periféricos.

O tamanho da via de dados é o tamanho da palavra do microprocessador. Este possui 8 bits. Já a via de dados é unidirecional, ou seja, o endereçamento sai direto do microprocessador para o sistema. Estes endereços são processados para buscar ou enviar informações externas.

Os sinais de controle comandam todas as operações do microprocessador, monitorando as informações.

26. R/W - Seleciona se lê ou escreve (em nível alto ele lê).

27. Entrada de clock - Internamente este micro gera o clock serial.

28. Saída de clock que permite sincronizar os demais circuitos integrados.

O 6507 possui uma arquitetura interna bastante simplificada, formada por acumulador, registradores, contador de programa e a ULA - Unidade Lógica Aritmética.

A ULA realiza todas as operações lógicas (E, OU, NAND, etc) e aritmética (soma, subtração, etc). No entanto, a ULA não faz tudo sozinha. Ela utiliza os registradores internos para auxiliá-la.

Estes registradores atuam como sinalizadores (Flags) que junto com a unidade de controle, selecionam os registros necessários para cada nova situação do circuito.

Os resultados de cada operação são armazenados num registrador especial denominado acumulador.

A unidade de controle é o coração do sistema: ela controla todas as funções. No entanto, as decisões são fornecidas pela memória, através das instruções recebidas pela via de dados. Ocorre que estas instruções vem codificadas e é preciso, portanto, decodificá-las para que possam ser executadas. Isto é feito pelo decodificador de instruções e executado pelo CPU.

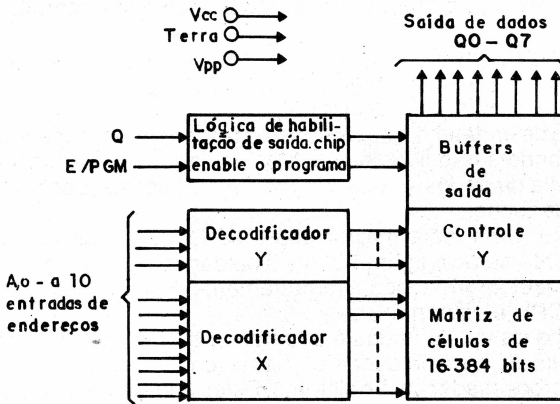
## **ROM (CARTUCHO)**

O cartucho consiste em uma memória ROM do tipo 2716. A figura 74 ilustra o diagrama em blocos de um cartucho. Normalmente, utiliza-se uma organização de 2048 palavras de 8 bits cada. Estas memórias que contém o software dos jogos apresentam 4 modos de operação:

- leitura
- stand by
- programação
- verificação

O modo de leitura precisa ser ativado pela CPU do vídeo-game, que habilitará





através do pino ENABLE (controle de alimentação).

O modo STAND BY é a espera, quando o vídeo-game é ligado, mas não é dado o START (partida). As saídas apresentam alta impedância.

O modo programação é quando o cartucho é gravado. Apresenta o estado interno 1. Portanto, a gravação é feita por selecionar quais os bits que são nível zero.

O modo verificação é para se determinar se a programação está correta, para isto, aplica-se 25V no pino VPP. Em condições normais a tensão deve ser igual a 5 V.

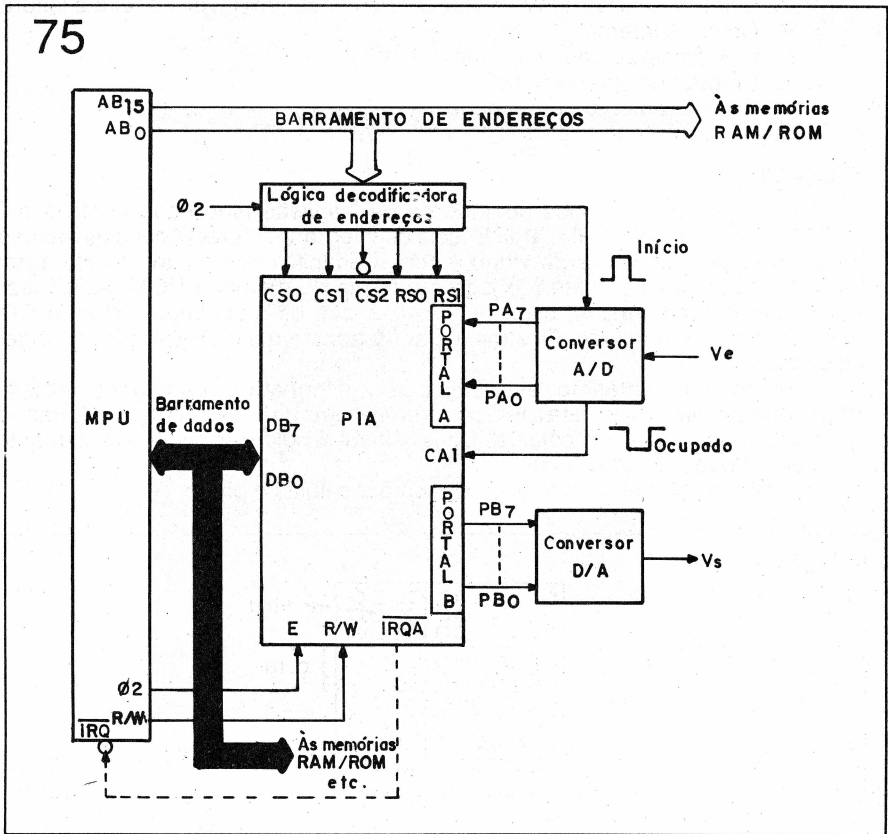
## PIA (6532)

A PIA (Interface Adaptadora de Periféricos) é formada pelo CI 6532 que tem por função fazer o interfaceamento entre CPU e os joysticks, além das chaves de controle.

A **figura 75** ilustra o diagrama em blocos internos da PIA. Nesse diagrama podemos ver que o dispositivo se comunica com a CPU através de um barramento bidirecional de 8 bits que fica acoplado ao mesmo barramento de dados da CPU. Existem duas portas bidirecionais de 8 bits, independentes, para serem conectadas aos joysticks.

Há vários registradores internos com a finalidade de permitir ao programador a escolha certa de recursos de interrupção, além de informações de Status referentes às interrupções (as interrupções de programa são verdadeiros triunfos em alguns jogos mais complexos).

Observe também neste diagrama em blocos que a PIA possui conversores A/D (Análogica Digital) e D/A (Digital Análogica), justamente para possibilitar o intercâmbio com sinais analógicos das chaves com a CPU e vice-versa.



Possui seis registros internos que armazenam os dados provenientes do joystick. Descreveremos sua pinagem,

1. Terra
- 2 a 7. Via de endereço - Estas saídas são utilizadas para endereçar as memórias e também os circuitos periféricos.
- 8 a 15. São as entradas dos joysticks.
16. Conexão com a chave S3 (dificuldade para o jogador direito).
17. Conexão com a chave S4 (dificuldade para o jogador esquerdo).
- 18 e 19. Não conectados (NC).
20. VCC
21. Entrada chave S5 (com cor/sem cor)
22. NC
23. Chave S6 (seleção de jogo)
24. Chave S7 (reset) - Faz com que o conteúdo do programa gerado seja resetado e que este recomece a partir da posição zero da memória.
25. Sinal de controle.
- 26 a 33. Via de dados. Forma uma via de comunicação bi-direcional entre este chip e o microprocessador.
34. Reset.

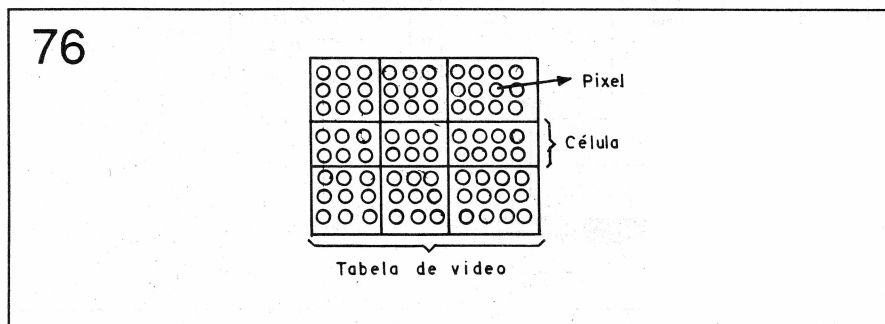
- 35. Sinal de controle R/W - Seleção lê/escreve (em nível 1 é feita a leitura).
  - 36. Controle interno.
  - 37 a 38. Endereçamento de memórias.
  - 39. Entrada do sinal de clock.
  - 40. Endereçamento da memória.
- Este CI é controlado pelo clock do 6507.

## TIA (6592)

Este CI faz a interface adaptadora de televisão (seu equivalente é o CI 10.750). Este CI do tipo LSI (Larga Escala de Integrado) oferece todos os recursos para a geração dos sinais de vídeo e som. Recapitulando o exposto até agora, temos que o cartucho contém o jogo em forma de memória ROM, a PIA faz a adaptação dos periféricos, onde os joysticks são os mais importantes; a CPU administra todo o percurso. Finalmente, a TIA transforma tudo isso em linguagem de vídeo.

Em capítulos anteriores estudamos como é formado uma imagem em digital. Vimos que o vídeo-game trata a tela como uma matriz de linhas e colunas, formando quadrículos denominados "células". Cada célula é dividida em pequenos elementos de imagem denominados "pixel".

A **figura 76** ilustra uma tela formada por células e pixels.

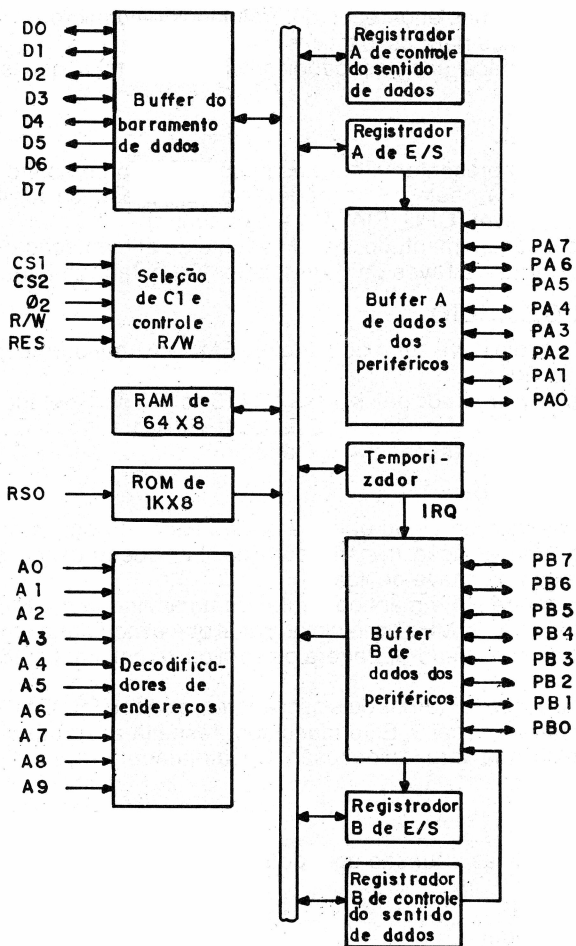


Ora, a TIA deve gerar todos os sinais de dados e endereços para formar cada pixel, cada célula e cada tela de imagem, com compatibilidade com o TV. Para executar esta árdua tarefa, a TIA possui uma estrutura interna bem completa, como ilustra o diagrama em blocos da **figura 77**.

Como se vê neste diagrama, esse componente contém uma memória ROM, uma memória RAM, além de buffer, registradores e decodificadores de endereços. (Queira reportar-se à **figura 57**, onde temos o diagrama em blocos didático da TIA, enquanto que este diagrama da **figura 77** é um diagrama mais técnico; a ROM e RAM contém os níveis de endereçamento para a geração de vídeo). Descreveremos sua pinagem:

### PINAGEM:

1. Terra
2. Sinal de sincronismo horizontal
3. Sinal de controle
4. Sinal de clock
5. Sinal de luminância
6. NG



- 7 e 8. Sinais de luminância
- 9. Saída do sinal de croma
- 10. Linha de atraso PAL
- 11. Sinal de clock PAL-M (3,575611 MHz)
- 12 e 13. Saídas de áudio
- 14 a 19. Via de dados
- 20. Alimentação VCC
- 21 a 24. Via de endereços
- 25. Seleção lê/escreve (lê em nível alto)
- 26. Entrada de clock proveniente do microprocessador

27 a 32. Via de endereços

33 e 34. Nestes pinos entram todos os dados ou informações a serem armazenados ou lidos nos endereços determinados pelo micro.

35 e 36. Interface para os joysticks.

37 a 40. Interface para os paddles (são os botões que substituem os joysticks).

## ESQUEMA GERAL

Na **figura 78** apresentamos um esquema elétrico completo e real de um vídeo-game versão Atari: nele, o leitor pode observar a presença dos 4 módulos que acabamos de descrever (CPU, PIA, TIA e o cartucho).

O circuito é complementado com uma fonte de alimentação que recebe 9 V do adaptador externo e através do CI regulador IC2 estabiliza a tensão para 5 V.

## OSCILADOR DE ÁUDIO

O sinal de áudio deve ser modulado em FM com uma portadora de 4,5 MHz da frequência de vídeo.

O sinal de áudio retirado pelos pinos 12 e 13 do IC 5 são injetados no transistor em Q 2 e deste circuito segue até o modulador de RF. O transistor Q 2 possui como polarização um circuito ressonante formado por L 5, L 6, C 15 e C 18.

## MODULADOR DE RF

O último estágio do vídeo-game é o modulador ou conversor de RF, que funciona como uma estação transmissora de TV, operando no canal 2 ou 3, conforme a posição da chave de ligação.

O modulador de RF é mantido dentro de uma blindagem. Ele deve gerar a portadora de vídeo a 1,25 MHz do início do canal que é modulado em AM pelo sinal de vídeo e mixada com a sub-portadora de som que o coloca a 4,5 MHz acima de seu valor.

No modulador de RF temos dois osciladores, a saber: 55,25 MHz para o canal 2 e 61,25 MHz para o canal 3. Este modulador respeita a largura do canal, que é de 6 MHz, especificada para radiodifusão. Um atenuador reduz o nível do sinal na saída do modulador.

## MC 1373

O IC MC 1373 faz a função do modulador de RF explanado acima.

Sua pinagem:

1. Oscilador RF
2. Circuito tanque
3. Terra
4. Entrada do sinal de vídeo composto
5. Entrada dos sinais de som e croma.
6. VCC
7. Saída do RF
8. NC

Este CI recebe sinais provenientes da TIA pelos pinos 4 e 5.

O transistor Q 3 amplifica o sinal de vídeo.

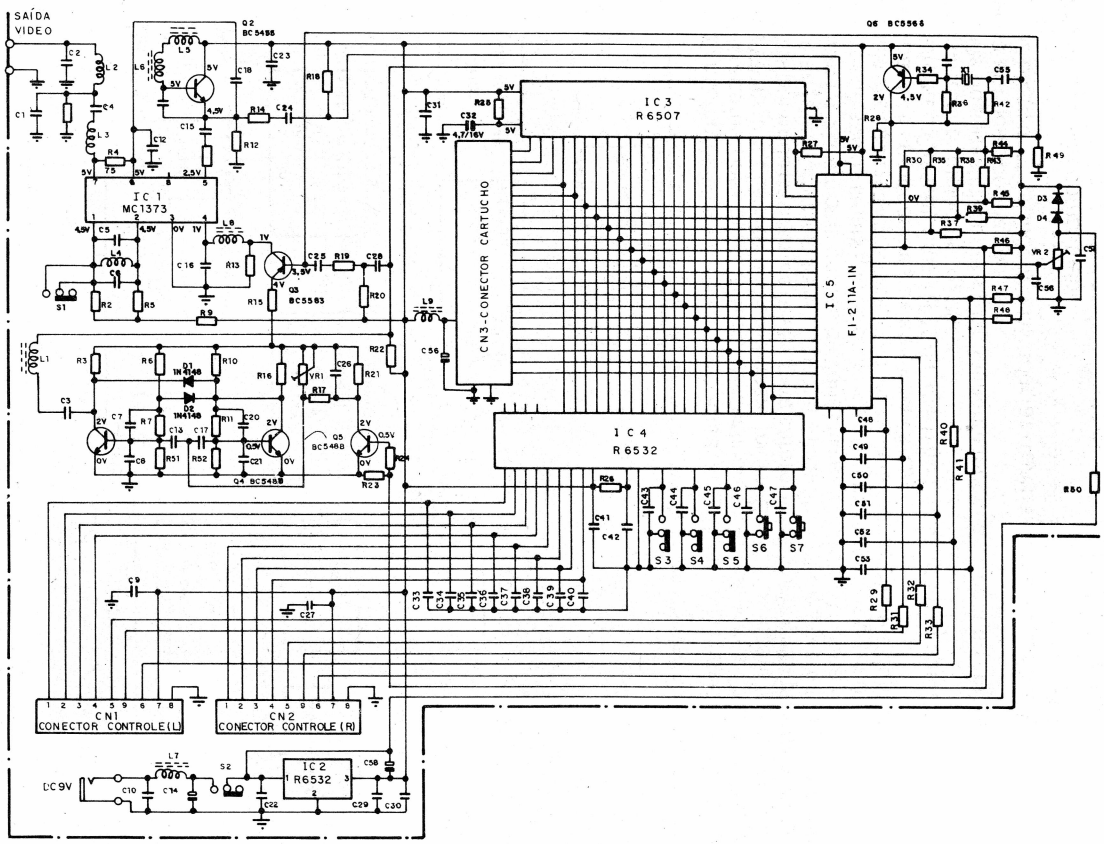
Q 4 e Q 1 amplificam os sinais de crominância.

Q 6 amplifica o sinal da sub-portadora de 3,58 MHz.

Q 5 amplifica os pulsos de sincronismo.

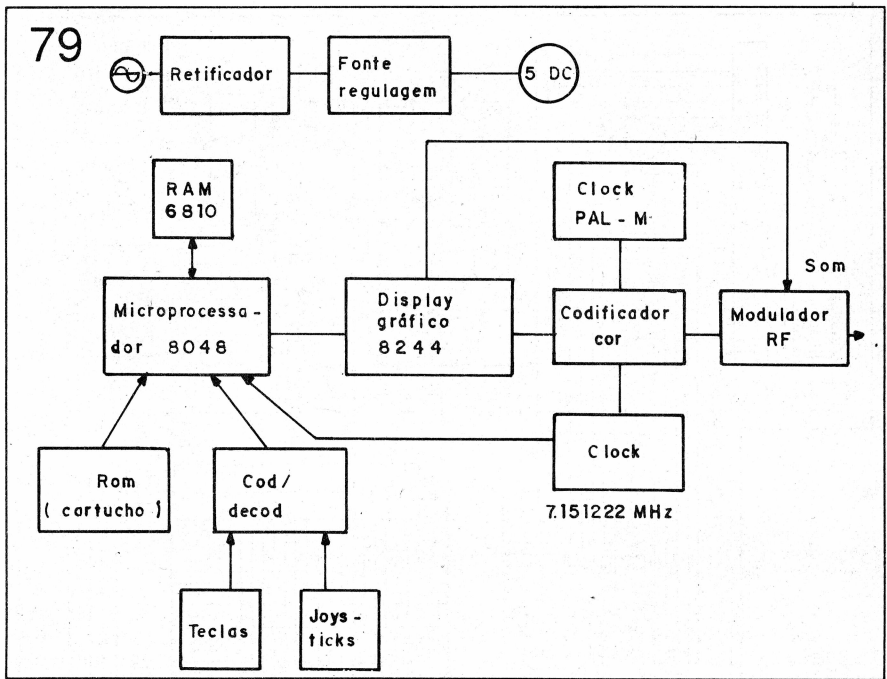
Q 2 é o amplificador de áudio.

No pino 7 retira-se o sinal de RF que segue até o conector de RF via L3, C4 e L2. O atenuador e filtro é formado por R1, C1 e C2.



## DESCRIÇÃO DO ODISSEY

O vídeo-game Odissey se assemelha muito a um microcomputador. A figura 79 ilustra o diagrama em blocos de forma didática.



O microprocessador utilizado é o 8048 da Intel, que incorpora acumuladores e registradores para substituírem, respectivamente, memórias RAM e ROM. A figura 80 ilustra a estrutura interna deste micro. Sua CPU possui um barramento de dados por onde passam os dados para serem usados pela unidade lógica (ALU).

A seção de controle fornece as variáveis de controle para o funcionamento do sistema. Existem dois grupos de variáveis de controle: As variáveis de controle internas que são os sinais que acionam os latches, buffers, registradores, multiplex e demais circuitos internos da CPU e as variáveis de controle externas que são os sinais para os componentes externos da CPU.

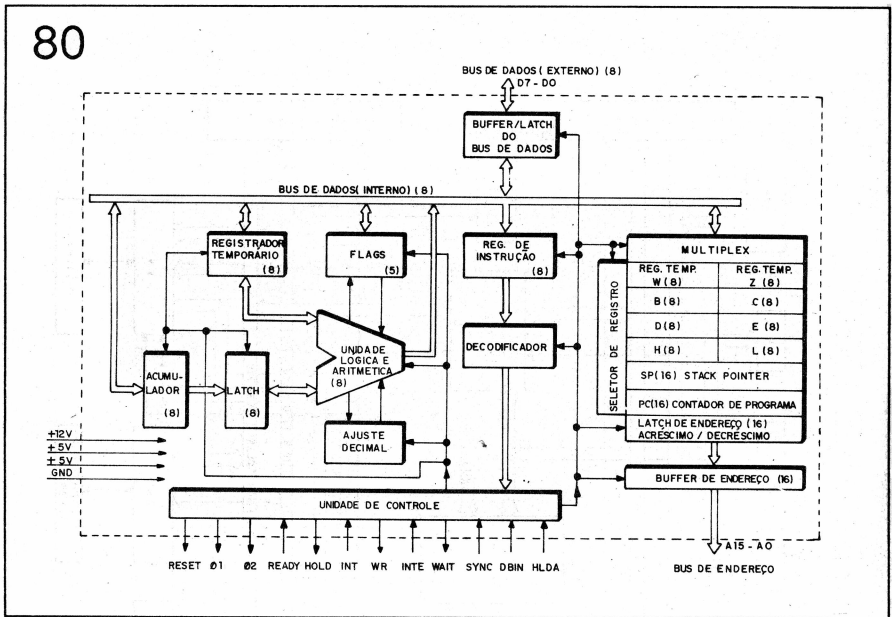
Utiliza o Odissey uma RAM de 128 bytes, utilizada nas operações da CPU com o cartucho, teclado e joysticks.

Os dois joysticks operam digitalmente fechando o contato correspondente à direção em que são manipulados.

O bloco decodificador/codificador faz o processamento dos sinais provenientes do teclado e dos joysticks. O teclado é lido pelo mesmo sistema de varredura utilizado em microcomputador.

O CI 8244 é o Display gráfico. Trata-se de uma CI dedicada, tipo uma ROM cujo software inclui: funções de controle, função de posições, gerador de sinais de vídeo composto (sincronismo, luminância e crominância) e sinais de som.

Baseado no modelo Atari, estudado no capítulo anterior, o 8244 seria o ROM gerador gráfico.



## DESCRIÇÃO DO MEGA DRIVE (SEGA)

O circuito do vídeo-game Mega Drive (procedência da Sega do Japão) é um dos mais sofisticados vídeo-games da atualidade. Utiliza um microprocessador de 16 bits, o que permite realizar 65.536 combinações binárias diferentes (lembrando que os modelos anteriores analisados são de 8 bits — 256 combinações).

Entre algumas de suas características podemos citar:

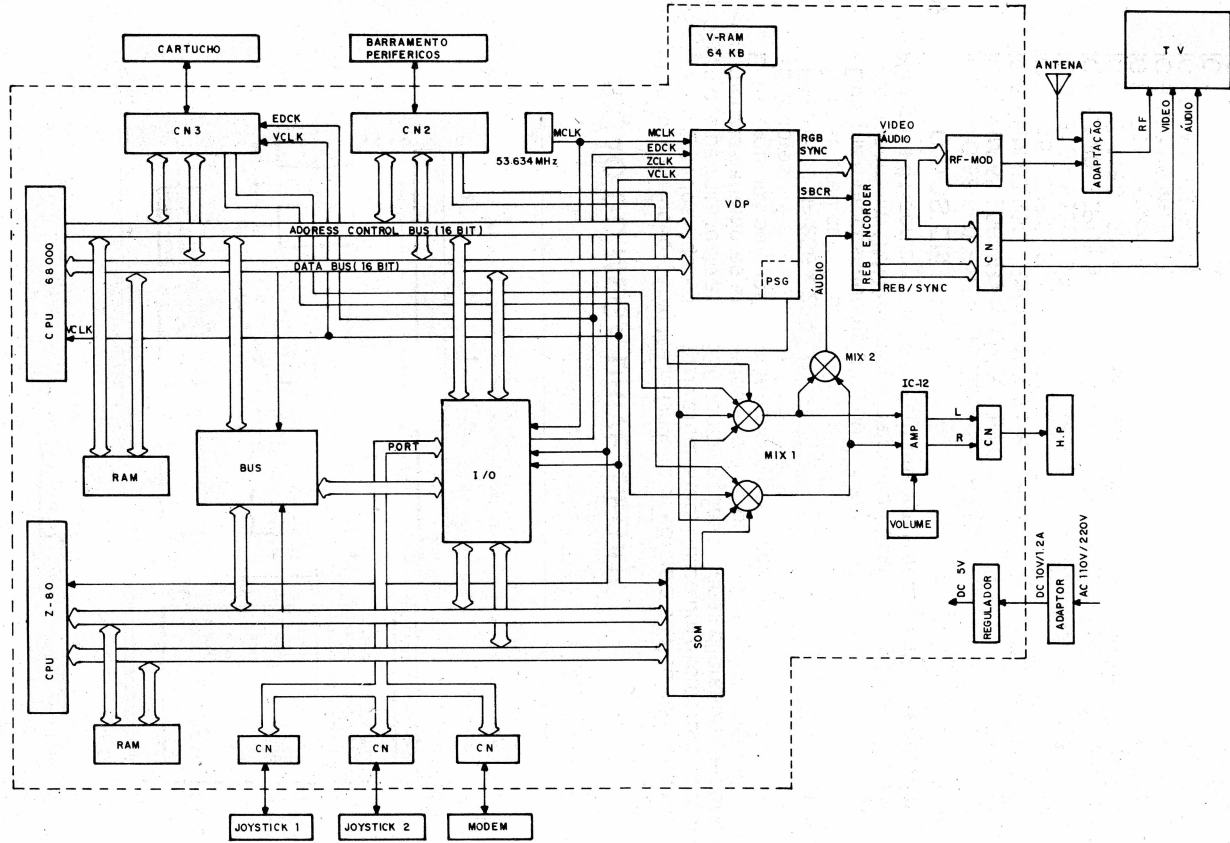
- 512 cores
- Resolução de 256 x 224
- Som estéreo com 6 tons
- Dois microprocessadores (68.000 e Z-80)
- montagem em SMD (montagem em superfície - dupla face)

Na **figura 81** apresentamos o diagrama em blocos deste vídeo-game.

## DESCRIÇÃO DOS ICs

- IC 1 - MC 68.000 - microprocessador 16 bits
- IC 2 - HM 65.256 - Ram 32K X 8 bits
- IC 3 - HM 62.256 - Ram 32K X 8 bits
- IC4 - 315 - 5364 - Bus
- IC 5 - 315 - 5309 - I/O
- IC 6 - Z-80 - CPU para áudio
- IC 7 - D 4168 - Ram 8K X 8 bits
- IC 8 - 315 - Processador de vídeo (VDP)
- IC 9 - HM 53461 - Ram 32K X 8 bits
- IC 10 - HM 53461 - Ram 32K X 8 bits
- IC 11 - YM 2612 - Conversor D/A para áudio
- IC 12 - CXA 1034 P - Pré estéreo de áudio
- IC 13 - CXA 1145 P - Codificador RGB e Buffer de áudio





IC 14 - LM 358 - Amplificador operacional  
IC 15 e IC 16 - LM 7805 - Regulador 5 V.

## DESCRIÇÃO DOS CONECTORES (CN)

1. Saída RGB (vídeo composto)
2. Periférico
3. Entrada cartucho
4. Modem
5. Entrada joystick 1
6. Entrada joystick 2
7. Entrada adaptador DC
8. Fone de ouvido (entrada)
9. Conector Led
10. Fone de ouvido (saída)
11. Conector para entrada do adaptador DC

A esta altura da leitura, você já domina os conceitos básicos de uma placa de vídeo-game, por isso, analisaremos agora apenas os CIs chaves, que são: o Z-80, o 6800 e VDP.

## CPU Z-80

A figura 82 ilustra o diagrama em blocos internos do microprocessador Z-80.

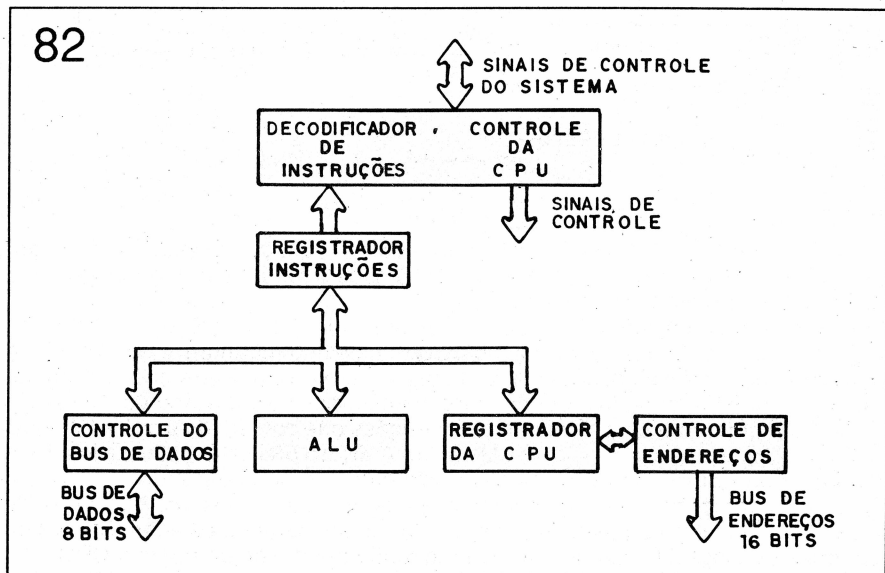
Esta CPU é compatível com o 8080, sendo mais poderosa. Os espaços internos para os registradores são amplos, permitindo um poderoso conjunto de instruções.

A figura 83 ilustra a pinagem do Z-80.

## DESCRIÇÃO DO Z-80

A0 - A16: são 16 bits de endereço com saída em alta impedância. Esta via de endereços fornece a locação de memória e dispositivos de E/S. Por ter 8 bits, endereça 256 dispositivos.

D0 - D7: Via de dados utilizada para intercâmbio com o circuito externo.



M1: saída que indica que a CPU está realizando um ciclo de busca de instruções. Veja diagrama do ciclo de busca de instruções na figura 48.

MREQ: Requisição de memória - Indica que a via de endereços possui um endereço para ler ou gravar na memória.

RFSH: Refresh - Pino de saída que possibilita o uso de memórias RAM dinâmicas.

WR: Pino que indica que a via de dados contém um dado para ser armazenado em dispositivos de I/O ou em memória.

RD: Pino de saída que indica que a CPU fará leitura de dados na memória ou em periféricos.

IORQ: Requisição para entrada ou saída de dados.

WAIT: Sinal de entrada que indica para a CPU que a memória ou periférico endereçado não está pronto para transferência de dados. Por isso, a CPU fica em estado de espera.

INT: Sinal de entrada proveniente de um periférico que está solicitando uma interrupção.

HALT: Pino de saída que indica que a CPU está executando uma parada por instrução de software aguardando uma interrupção.

NMI: (Non Maskable Interrupt) - Pino de entrada que controla o contador de programa interno.

RESET: Inicializa a CPU (entrada, ativa em zero).

BUSRQ: Pino de entrada que requisita as vias de dados sem a intervenção da CPU. É usado no DMA - acesso direto a memória.

BUSAQ: Pino de saída que indica que o sinal BUSRQ foi reconhecido.

CLOCK 2 MHz - No Z-80 a oscilação tem que ser feita por meio de um circuito externo, normalmente através de um cristal a quartzo.

## APLICAÇÃO NO MEGA DRIVE

Podemos dividir a atuação do Z-80 (IC 6) no vídeo-game Mega Drive em duas situações.

a) Cartucho Mega Drive

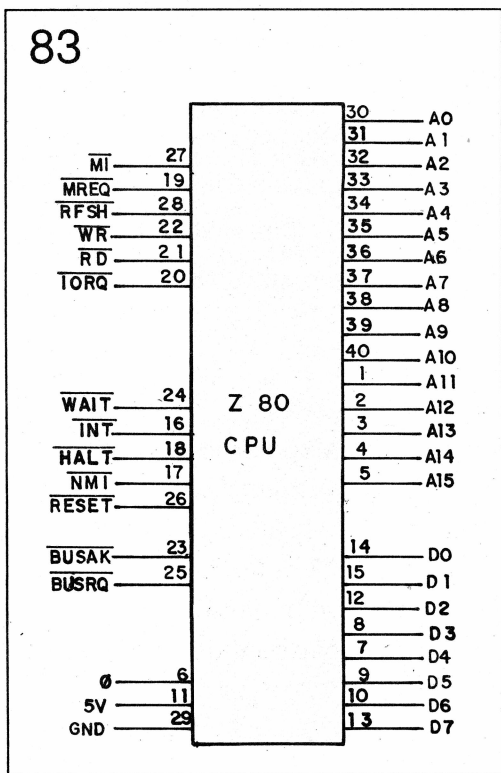
Ao utilizar o cartucho normal, esta CPU opera apenas na seção de áudio.

b) Com Adaptador Para Cartucho Master System

Ao utilizar o adaptador e um cartucho Master System, ele passa a operar como CPU principal e não mais processa o áudio (neste caso o áudio será processado pelo IC 8 - VDP).

## CPU 68.000

A CPU 68.000 da Motorola é um microprocessador de 16 bits com 17 registradores de dados e de endereços de 32 bits e possui um barramento de dados



de 16 bits e um barramento de endereço de 24 bits.

Nas **figuras 84, 85 e 86** vemos três tipos de encapsulamentos do 68.000 e suas pinagens e na **figura 87** temos o destaque dos principais sinais.

## DESCRIÇÃO DA PINAGEM

DO - D15: compõe a via bidirecional de dados

A1 - A23: é a via de endereços

BGACK: esta entrada indica que o periférico reconheceu sinal BG (autorização de via).

IPL0, IPL1 e IPL2: entradas que o periférico utiliza para requisitar interrupções no programa da CPU.

BERR: Entrada que indica se existe algum erro no ciclo corrente. Quando este sinal é ativado, a CPU inicia uma seqüência de operações para processar os periféricos.

RESET: inicialização

HALT: é tanto um sinal de entrada como de saída. É um sinal de controle.

ENABLE: sinal de habilitação

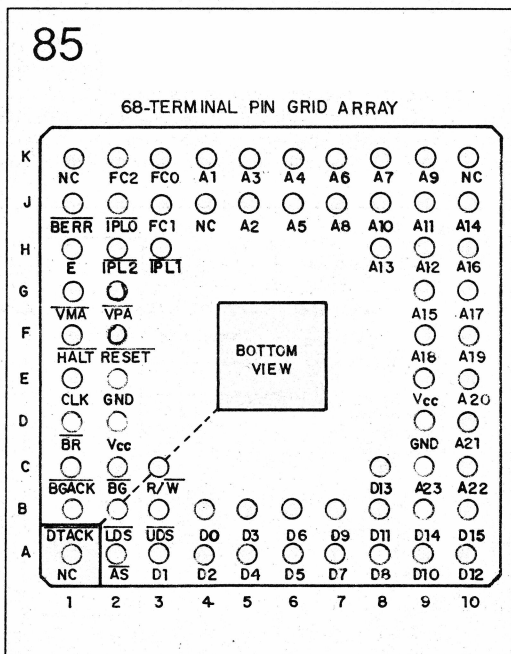
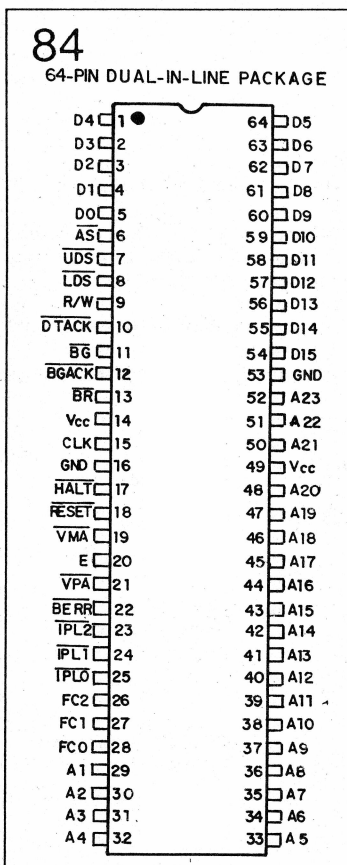
VPA: entrada que indica que o periférico está pronto para a transferência de dados, sincronizado com o sinal Enable.

VMA: saída que indica que o processador tem endereço de memória válido no barramento.

FC0, FC1, FC2: saídas de status da CPU. São códigos de função, conforme ilustra a tabela da figura 51.

AS: indica que o endereço no barramento de endereço é válido (Strobe)

R/W: Seleção lê/escreve



USD/LDS: A tabela da figura 52 define o significado de UDS e LDS em relação à via de dados, ou seja, qual é válida.

UDS - Válido de D8 a D15

LDS - Válido de D0 a D7

DTACK: Esta entrada indica que a transferência de dados foi completada.

BR: Solicitação de via, requisitando via do periférico.

BG: Saída que indica que a CPU reconheceu o pedido de via de periférico.

CLOCK: Sinal de temporização. É compatível com nível TTL.

### VDP

O VDP (Vídeo Display Programable) é o CI LSI que gera todos os sinais de vídeo (sincronismo, luminância e crominância). Está codificado no Mega Drive por 315-5313. Ele desempenha as mesmas funções da TIA, descrita na figura 57, porém, com uma resolução muito maior, com a diferença que no CI da TIA, os blocos de processamento eram lógicos-combinatórios (contadores, flip-flops, etc). No VD o processo é feito por lógica-programável por software.

## DESCRIÇÃO DO MEGA DRIVE

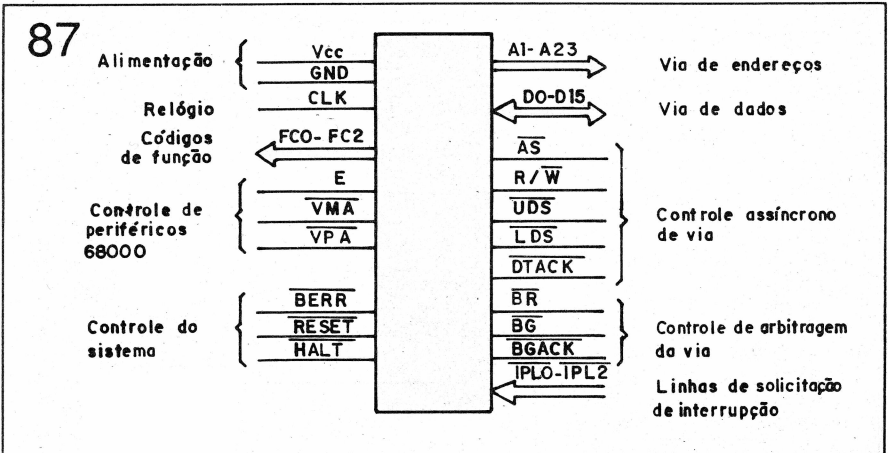
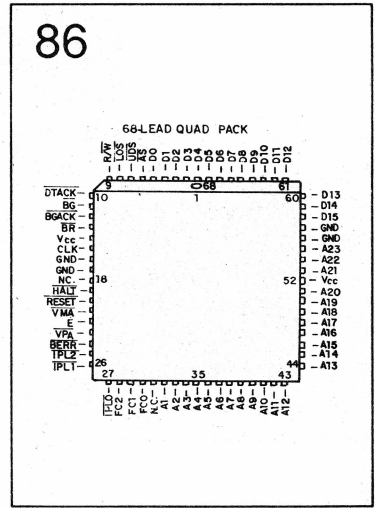
Oriente-se pelo diagrama em blocos da figura 83.

A CPU (IC 1) 68.000 controla todo o sistema. As memórias RAM estáticas (IC 2 e IC3) são as memórias disponíveis para a CPU.

Através do IC 4 (Bus Arbiter) é feita a interligação entre o Z-80 e o 68.000. É utilizado quando é colocado o adaptador e um cartucho normal, opera como gerenciador de dados de áudio com Z-80. Possui como memória RAM o IC 7.

O IC 5 (I/O) é o responsável pela leitura dos joysticks.

O IC 8 (VDP - Vídeo Display Programable) é o processador de vídeo que gera todos os sinais de vídeo (sincronismo horizontal, vertical, burst, luminância e crominância). Seu clock é de 53.634165 MHz que é dividido por 15, gerando o clock



de 3,575611 MHz.

Possui também, internamente, um gerador de som programável, utilizado durante os cartuchos Master System. Utiliza os IC 9 e IC 10 como memórias RAM de vídeo.

O IC 11 (Sound) é um conversor A/D (analogico/digital) destinado ao processamento de áudio. Ele é gerenciado pelo Z-80 durante o uso dos cartuchos Mega Drive.

O sinal de áudio, depois de convertido para analógico é amplificado pelo IC 12 e é injetado no conector do fone de ouvido.

Um mixer efetua a mistura dos sinais de áudio para injetá-lo junto com o sinal de vídeo no IC-13.

O IC 13 (Encoder RGB) codifica todos os sinais na forma de vídeo composto e ainda possui um buffer de áudio que seguirá até o conversor de RF e também uma saída de áudio mono.

## DESCRIÇÃO DO NINTENDO

A exemplo do Mega Drive, o vídeo-game Nintendo utiliza uma CPU de 16 bits, permitindo uma ampla gama de combinações binárias, sendo neste detalhe que reside o aumento da resolução da imagem e na quantidade de cores possíveis.

A arquitetura de um Nintendo assemelha-se muito a uma placa mãe de um microcomputador.

Neste tipo de configuração, o cartucho é como se fosse um disquete de computador contendo todo software, e a CPU e seus periféricos irão processar todos os sinais e transformá-los em sons e imagens.

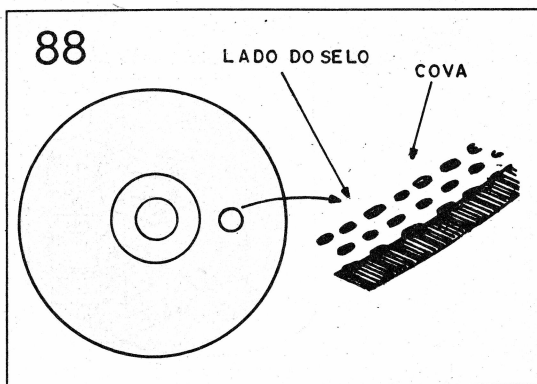
## DESCRIÇÃO DO MEGA-CD-ROM

Numa linguagem bastante simplificada poderemos dizer que o Mega CD-ROM é um vídeo-game que possui um leitor óptico, semelhante aos utilizados nos compact disc players. No lugar do cartucho temos um disco óptico, compacto, denominado CD.

A principal diferença existente entre um CD-ROM usado em vídeo-game e computadores e um CD de áudio, está na capacidade de correção de erros. Ambos os discos são fabricados com a mesma tecnologia. Imagine o leitor que nos mesmos 12 cm de diâmetro de um CD como mostra (figura 88) usados para informação de áudio (música), foi adicionado imagens e outros sinais de controle (software) em um CD-ROM. Como não aumentou o espaço físico e o sistema deve ser compatível (o Mega CD reproduz um CD de áudio, mas um CDP de áudio não reproduz um CD-ROM), foi adotado uma codificação, especial para a correção de erros.

(Nota ao leitor: Sugerimos ao leitor que adquira desta mesma Editora, o livro que trata sobre " Compact Disc", também de minha autoria, para que possa dominar os conceitos da tecnologia de leitura óptica a laser).

Como ilustração, apresentamos na figura 89 o esboço de como funciona um



CD-ROM: a unidade óptica emite um feixe de luz laser que irá ser refletida pelo disco, ou absorvida pelas microcavidades. O feixe de luz de retorno é decodificada pela unidade óptica e convertida em pulsos elétricos que posteriormente serão decifrados pelos circuitos integrados dedicados.

## CONSERTE SEU PRÓPRIO VÍDEO-GAME

Após ter adquirido os conhecimentos teóricos do vídeo-game, você está apto a realizar reparações em vídeo-games.

Passaremos algumas diretrizes para melhor eficiência nos seus serviços de reparação.

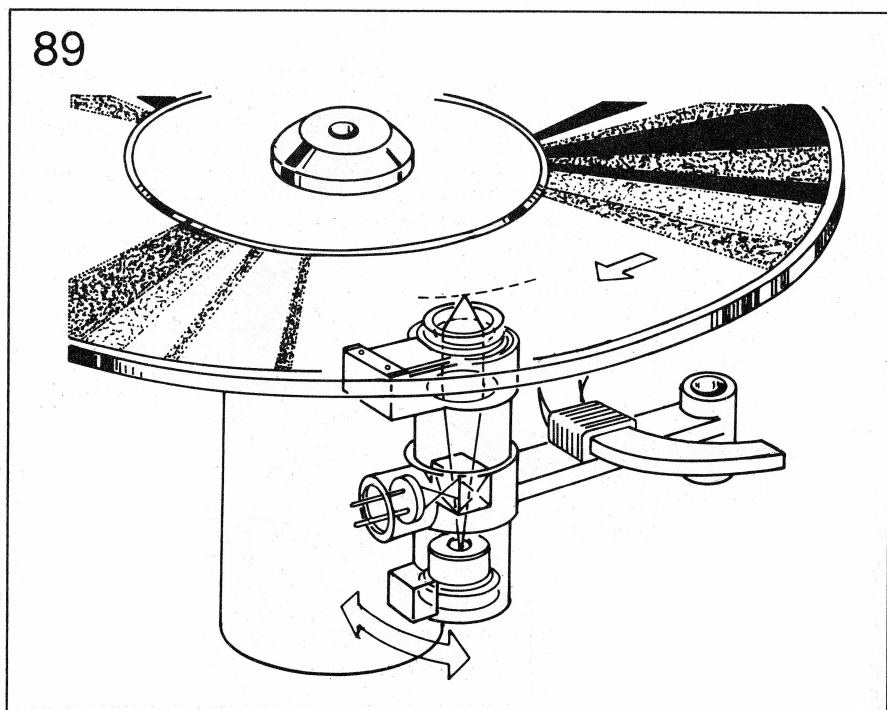
Podemos dividir as técnicas de reparação em vídeo-game em quatro partes, a saber:

1. Análise dos sintomas
2. Diagnóstico
3. Isolar estágio defeituoso
4. Isolar componente defeituoso

### 1. ANÁLISE DOS SINTOMAS

Primeiro, precisa manusear o aparelho, instalar ele no televisor (terminal de antena), colocar o cartucho e ligar. Para isto, você precisa saber reconhecer quando o funcionamento é normal e quando não está correto.

A etapa de análise dos sintomas não significa que você faça uma investida pelo vídeo-game adentro, com chave de fenda, instrumentos, etc. Significa que você deve fazer uma inspeção visual, operar os joysticks e teclas em busca de



maiores indícios. É recomendável ter cartuchos de diversas marcas (Atari, Odissey, Nintendo, Sega) para poder realizar os testes.

## **2. DIAGNÓSTICO**

O diagnóstico consiste em uma análise funcional, para saber, a partir do levantamento de sintomas, determinar qual estágio poderia provocar aquele defeito. Aqui fazemos uso dos nossos conhecimentos teóricos, nosso raciocínio técnico e nossas deduções teóricas.

## **3. ISOLAR ESTÁGIO DEFEITUOSO**

Depois do defeito ter sido localizado numa área funcional pelo diagnóstico, a etapa seguinte é isolar o estágio defeituoso.

Esta etapa envolve o uso de instrumentos de medida, multímetro, osciloscópio, frequencímetro, ponta de prova lógica. Pesquisa-se os sinais utilizando o esquema elétrico do vídeo-game. Utiliza-se na saída um sinal de que depende dos sinais injetados na entrada.

Note que o diagnóstico serviu para determinar qual estágio deve ser testado, para, então, determinar se realmente é aquele o estágio defeituoso.

## **4. ISOLAR COMPONENTE DEFEITUOSO**

Basicamente este estágio consiste em testar isoladamente o componente suspeito, determinado pelo terceiro passo.

É comum o uso de CI dedicado do tipo LSI (Larga Escala de Integração). Por isso, muitas vezes o defeito é um destes CIs, que deverá ser feito por outro de igual código e, se possível, da mesma marca.

Algumas vezes, isolar o componente defeituoso implica em retirar transistores, diodos ou capacitores para testá-los fora do circuito, utilizando para isto o ohmímetro (multitester).

## **INSTRUMENTAÇÃO**

Se analisarmos o vídeo-game como sendo um microcomputador, então, todo instrumental de microcomputador poderá ser utilizado na reparação de vídeo-games.

Evidentemente, um instrumental completo pode ser inviável se formos pensar em utilizá-lo só para reparações de vídeo-game. Contudo, este instrumental (ou parte dele) é aplicável em quase todas as modalidades de reparações.

Abaixo, relacionamos o instrumental completo para os serviços de reparação de vídeo-game e em seguida, comentaremos alguns itens que julgamos especiais.

## **LISTA DE INSTRUMENTOS**

- Multímetro
- Osciloscópio
- Frequencímetro
- TV para monitoração
- Cartucho teste
- Ponta de prova lógica
- Analisador de níveis lógicos
- Pesquisador de sinais de áudio

## **CARTUCHO TESTE**

O cartucho teste (ROM de Debugging) é uma memória ROM onde está gravado um programa que testa os estágios e indica na tela os prováveis estágios defeituosos. Este cartucho tem que ser compatível com o vídeo-game (exemplos: Atari, Odissey, Sega, etc). Observando a imagem na tela do TV, podemos verificar o funcionamento correto da memória, da CPU, da TIA, etc...

O cartucho teste poderá ser substituído por um cartucho comum.

## **PONTA DE PROVA LÓGICA**

A ponta de prova lógica é muito útil, pois indica com um simples conjunto de leds, se o nível é alto, baixo, tri-state ou trata-se de pulsos oscilantes.



## **ANALISADOR DE NÍVEIS LÓGICOS**

O analisador de níveis lógicos é um instrumento que permite analisar os níveis digitais de forma dinâmica, ou seja, executando um programa junto ao microprocessador.

## **PESQUISADOR DE SINAIS DE ÁUDIO**

Este instrumento terá aplicação apenas quando o defeito for relacionado ao áudio, permitindo pesquisar o trajeto do sinal de áudio a partir do conversor de RF. Pode ser substituído pelo multímetro ou, então, pelo osciloscópio.

## **MÉTODOS PARA REPARAÇÃO**

Seguindo a técnica dos quatro passos, recomendada anteriormente, podemos ainda sugerir alguns métodos de reparação, que variam em função do modelo de vídeo-game a ser reparado. Os métodos são:

### **a) SUBSTITUIÇÃO**

Este método consiste em substituir (trocar) o CI que supostamente esteja danificado (conclusão tirada pelo diagnóstico e pelo terceiro passo que é isolar o estágio defeituoso).

Naturalmente, este método só é válido para vídeo-games que tenham um número reduzido de integrados ou quando estes são soquetados.

Caso tenha um vídeo-game de reserva (um para cada modelo), poderá fazer as substituições do bom (seu reserva) para o defeituoso (do cliente). Faça sempre com um CI de cada vez.

Por exemplo, retire o microprocessador do aparelho defeituoso e substitua-o pelo microprocessador de reserva. Se este funcionar, então, o defeito era o microprocessador.

Deste modo, você comprará esta peça na certeza de que ela era necessária. Por outro lado, se o defeito prevalecer, então retorne o microprocessador para o seu aparelho de reserva, reinstale o microprocessador original e repita a operação para a memória, depois para a TIA, PIA, etc.. Mas lembre-se, sempre um de cada vez.

### **b) PESQUISA COM OSCILOSCÓPIO**

A pesquisa com osciloscópio (recomenda-se um acima de 20 MHz) deve ser feita a partir do microprocessador até chegar na saída de RF (conector da antena). A figura 53 ilustra algumas formas de ondas do videogame Atari. É sempre necessário colocar um cartucho e acionar o videogame.

Uma sugestão interessante é pegar um videogame bom, colocar um cartucho e verificar as formas de ondas dos principais CIs. Desta forma, você se familiarizará com os oscilogramas e poderá compará-las com o videogame defeituoso. Quanto maior for o valor de varredura do seu osciloscópio, melhores aparecerão estes oscilogramas.

## **TESTES POR COMPONENTES**

### **1 - MICROPROCESSADOR**

Verifique a alimentação, reset e o clock. Estando tudo normal, verifique os pulsos de varredura que saem do microprocessador. Teste os sinais de sincronismo e os sinais de controle.

### **2 - MEMÓRIAS**

Verifique a alimentação e os sinais de controle (no caso de RAM, lê/escreve). Na prática, a substituição é o melhor teste.

### **3 - CARTUCHO**

Para testá-lo, tire o circuito impresso com a ROM do cartucho e encaixe-o no

console, sem as suas armaduras. Verifique o funcionamento com o aparelho ligado. Então saberá se o defeito é proveniente do conector do cartucho ou da ROM.

#### **4 - JOYSTICK**

Para testar o joystick e o seu respectivo circuito (PIA) utilizamos a ponta de prova lógica ou até mesmo um voltímetro DC (nível 0 = 0 V. e nível 1 = acima de 2,5 V.). Veja os procedimentos:

- com os joysticks não acionados, devemos ter nível um (alto) nos pinos 8 a 15 da PIA.

Mova os nas quatro direções e verifique se para cada uma delas o respectivo pino da PIA se altera para nível zero.

Faça o teste dos botões - deve ter o pino 35 e pino 36 da PIA em zero ao ser acionado (um pino para cada botão).

Se houver irregularidade nos testes, então retire os joysticks e simule na PIA (através dos terminais do joystick) os níveis 0 (terra) e 1 (5 V. aplicado via resistor de 4k7 ou 6k8).

Se todas as entradas apresentarem saída normal, substitua o joystick. Se não, substitua o CI da PIA

No caso do Odissey, deve-se testar também as combinações do teclado, que vão para o decodificador 74LS156.

Recomendamos a aquisição destes esquemas (Odissey e Atari).

#### **DEFEITOS MAIS PROVÁVEIS E SUAS CAUSAS**

A seguir, apresentamos uma relação de defeitos composta por sintomas e prováveis causas. É válido para qualquer tipo de vídeo-game.

Salientamos que estes "macetes" só serão verdadeiros se antes de tudo forem verificados os seguintes dispositivos ou blocos funcionais:

- fonte de alimentação
- conector antena
- ballon
- inspeção visual da placa
- conexões mecânicas, incluindo o gabinete (encaixes, plugs, conectores, cabos, etc).

- conector do cartucho (este é provavelmente a parte que apresenta maior incidência de defeitos em virtude de ser o que fica mais em contato direto com o usuário, que na maioria das vezes é criança).

Uma vez checados os itens acima, podemos catalogar os defeitos mais comuns.

##### **1. Imagem ruim com uma única cor.**

- Possíveis causas:
- microprocessador
  - cristal
  - TIA
  - PIA
  - Modulador RF
  - Transistores conectados na TIA

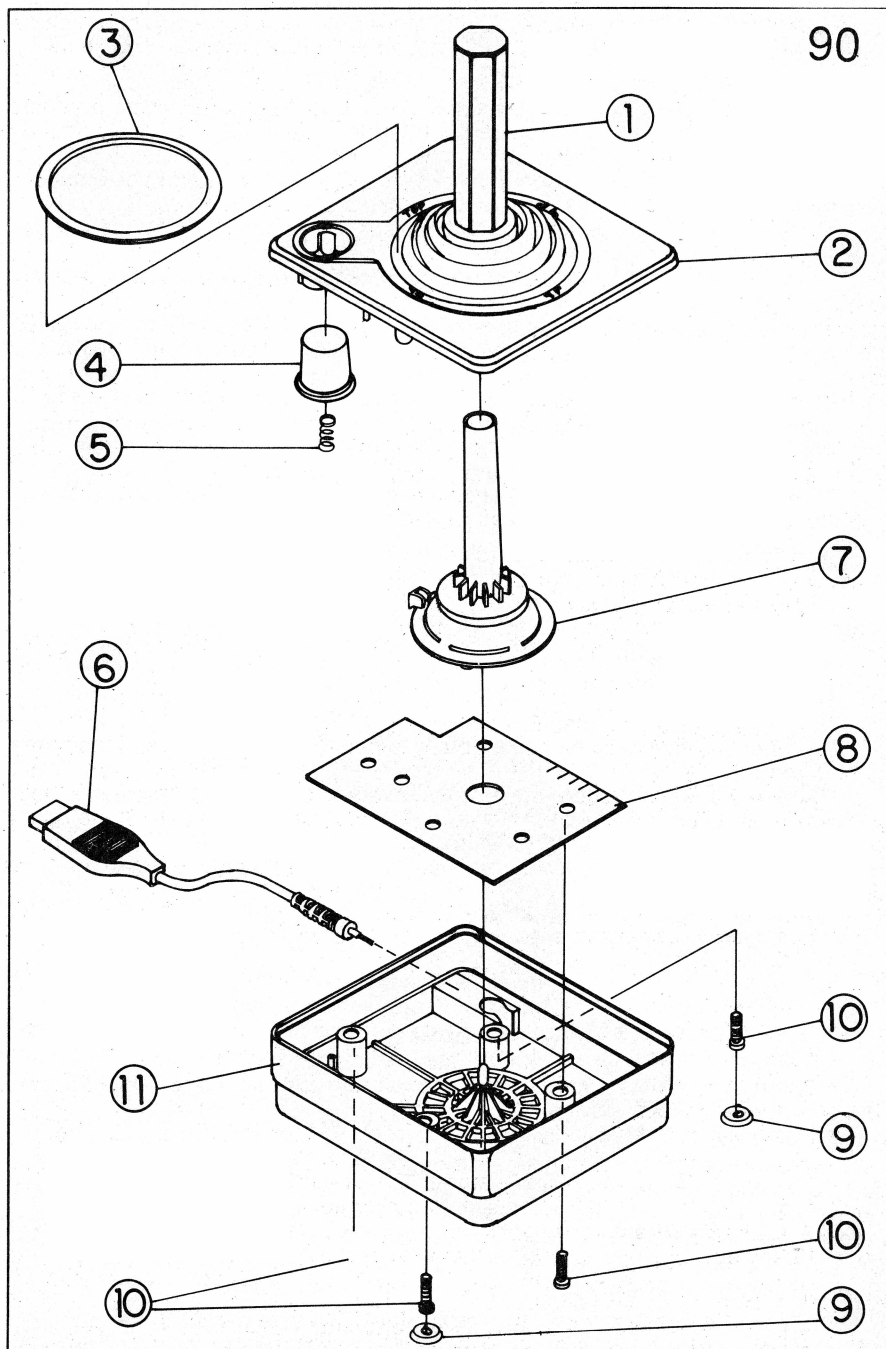
##### **2. Imagem com retraços que podem ser em formas diagonais, verticais, em N ou em X na tela.**

- Possíveis causas:
- TIA
  - PIA
  - Transistores conectados na Tia
  - Capacitores conectados na TIA ou PIA

##### **3. Linhas verticais com cores:**

- Possíveis causas:
- microprocessador
  - TIA
  - PIA
  - Conector do cartucho

##### **4. Vídeo com aparência de neve**



Possíveis causas: - Fonte regulador de 5 V.  
- Modulador de RF  
- Ballon

#### **5. Vídeo fraco, sem contraste**

Possíveis causa: - Modulador RF  
- Conector da antena

#### **6. Imagem boas mas sem cor**

Possíveis causas: - TIA  
- Cristal de 3,58 MHz  
- Circuito PAL-M  
- Capacitores do circuito de croma  
- Transistores do circuito de croma

#### **7. Cores fracas**

Possíveis causas: - Modulador RF  
- Conector e cabo da antena  
- Resistores e capacitores  
- TIA

#### **8. Não tem a mesma cor ou ainda aparecem barras de ruídos nas cores**

Possíveis causas: - TIA  
- Trimpot na seção de croma  
- Diodos conectados a TIA  
- Capacitores adjacentes a TIA

#### **9. Falta som (imagem boa)**

Possíveis causas: - Oscilador de 4,5 MHz  
- Modulador RF  
- Transistor de áudio que está ligado na TIA

#### **10. Sinal fraco ou com ruídos**

Possíveis causas: - TIA  
- Transistores e capacitores conectados a TIA e que tenham funções no áudio  
- Calibragem no modulador RF

#### **11. Botão de disparo de tiro não funciona**

Possíveis causas: - Conector do joystick  
- Joystick

#### **12. Controlador não funciona**

Possíveis causas: - PIA  
- Conectores  
- Resistores

#### **13. Chaves inoperantes**

Possíveis causas: - PIA  
- Capacitores conectados a PIA  
- Chaves com problema de meu contato

#### **14. Não faz todas as funções e ainda aparece no display "Overlap Bad".**

Possíveis causas: - Microprocessador  
- RAM  
- Circuitos adjacentes a RAM

### **ATENÇÃO NA COMPRA DE CIs**

Recomendamos muita cautela e atenção na compra de circuitos integrados para reposição em vídeo-games, pois os seus códigos costumam ser complicados. Ao substituí-los, certifique-se de que realmente sejam equivalentes. As vezes, a mudança de uma letra no código do CI pode torná-lo inadequado ao circuito original, sob pena do aparelho não funcionar.

### **TROCA DO JOYSTICK**

É muito comum a quebra do joystick devido a ação mecânica do operador para substituir o(s) elemento(s) danificados do joystick, oriente-se pela **figura 90**.

### Legenda

- 1 - Capa da manopla
- 2 - Tampa do controle
- 3 - Anel de retenção
- 4 - Botão de disparo
- 5 - Mola espiral
- 6 - Cabo
- 7 - Manopla
- 8 - PCI do controle
- 9 - Lacre (estando fora da garantia, pode desconsiderá-lo)
- 10 - Parafuso de fixação
- 11 - Base do controle

## ORIENTAÇÃO PARA DESSOLDAGEM EM SMD

É comum em alguns modelos de vídeo-games o uso de circuitos integrados montados em superfície (técnica denominada SMD- Surface Monted Devices).

Para dessoldar um SMD defeituoso, deverá utilizar uma malha de cobre de cabo coaxial umedecida com um mistura de álcool isopropílico com Breu. Coloque entre os terminais de CI e aqueça com o ferro de soldar.

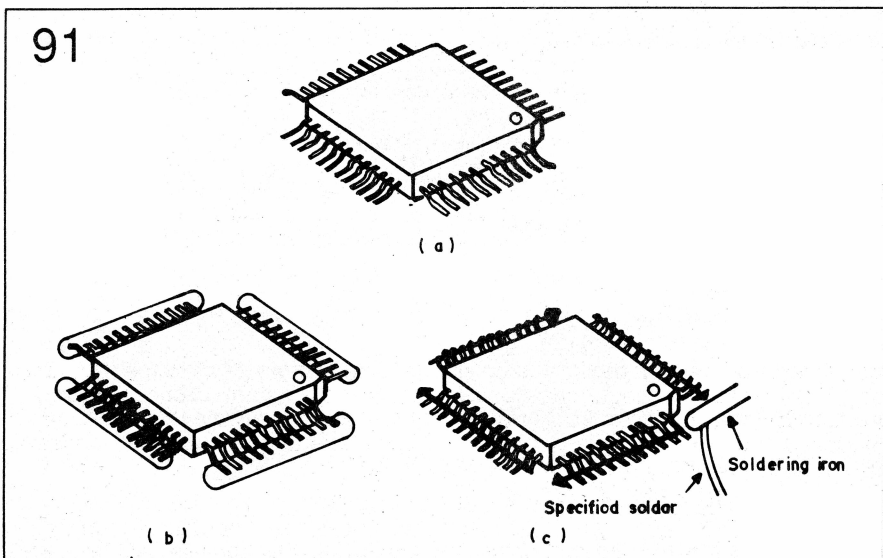
Após retirar o CI, limpe bem a placa com álcool isopropílico (poderá usar uma escovinha de dentes).

A soldagem do CI SMD deve ser feita com um ferro com temperatura controlada e com isolamento: é conveniente usar uma estação de solda.

Para não "grudar" um pino ao outro, utilize uma química anti-coagulante, podendo ser um WD-40. Veja as ilustrações na **figura 91**.

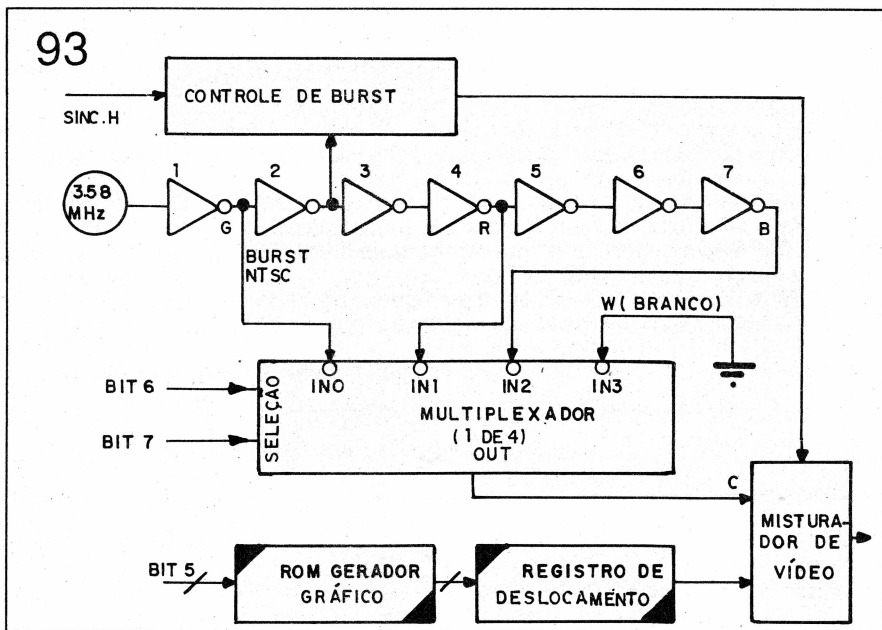
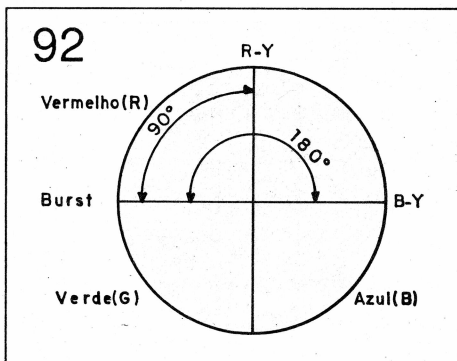
Coloque a placa no sentido vertical e com solda fina de boa qualidade solde os terminais de cima para baixo, para que a gota de solda derretida escorra pelos terminais de cima para baixo.

Após esta operação, certifique-se de que todos os terminais foram soldados e que não há curto entre eles.



## TRANSCODIFICAÇÃO DE VÍDEO-GAMES

Apesar de haver vários fabricantes nacionais produzindo aparelhos de vídeo-games dos mais variados modelos, muitas vezes entram no mercado aparelhos importados produzidos especificamente para o consumidor americano ou japonês, ou seja, no sistema NTSC. Quando isto ocorrer, será necessário transcodificá-lo para que este funcione em seu TV PAL-M.

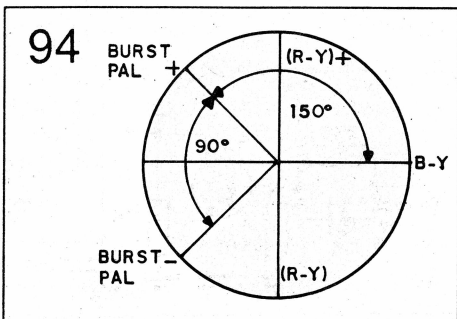


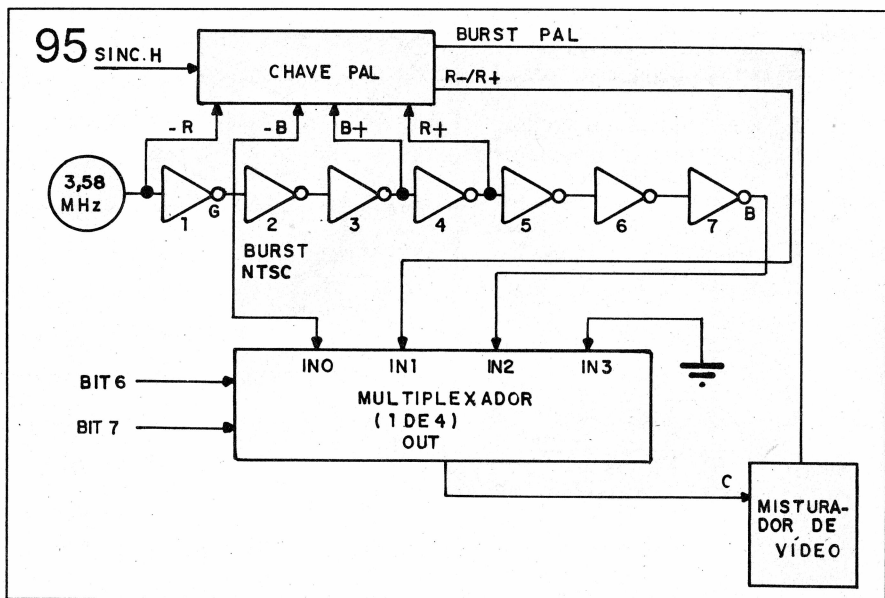
## CODIFICAÇÃO DE COR

A cor, no vídeo-game, é codificada como defasagem do burst como mostra a (figura 92).

Uma implementação digital está ilustrada na figura 93 (para NTSC).

Selecionando-se o número de gates que o burst e as cores primárias (vermelho, verde e azul) devem atravessar, obtém-se a defasagem desejada.





Como sabemos, os vídeo-games são projetados em NTSC. Para acoplá-los ao TV PAL-M, são necessárias as alterações ilustradas na **figura 94**, a saber:

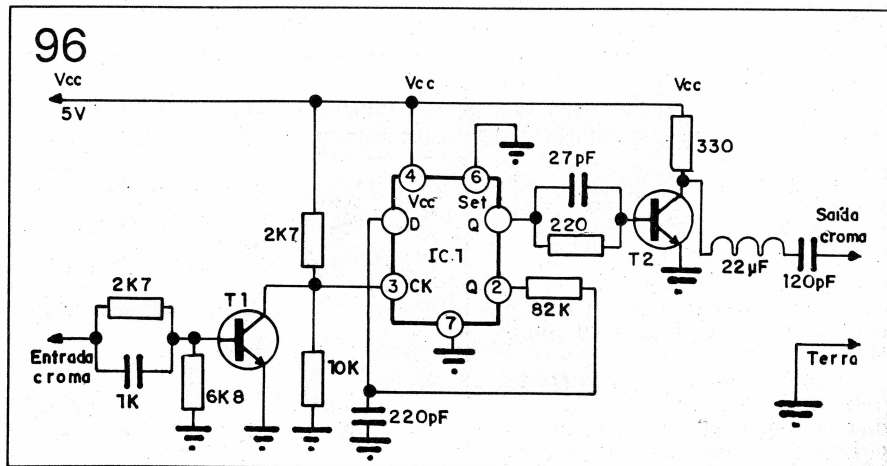
- avançar 45° o burst NTSC
- defasar alternadamente, linha por linha em 90° componente R-Y.

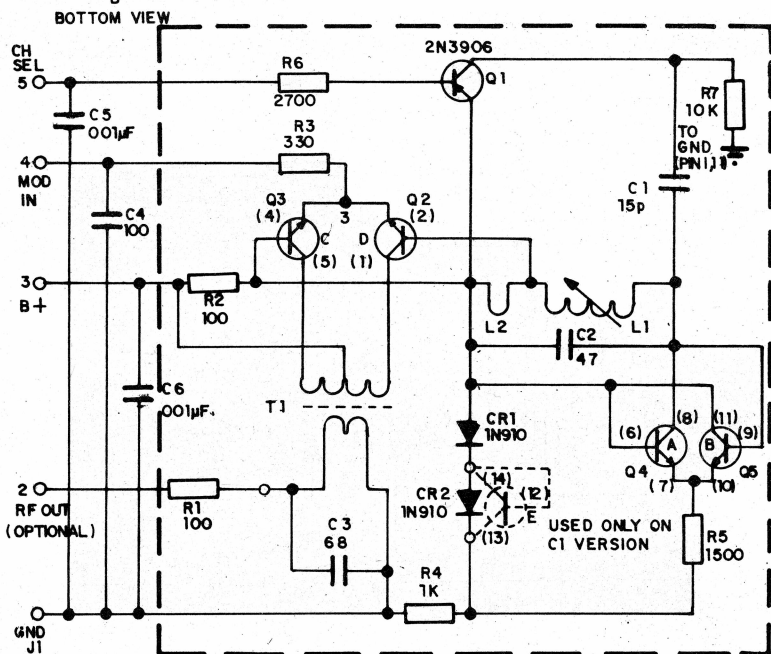
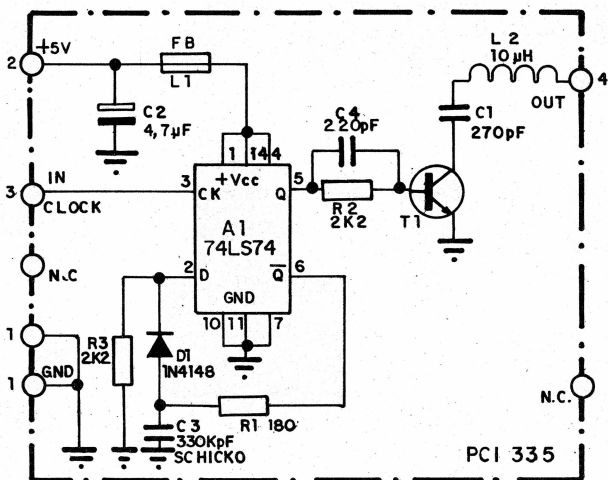
Um circuito que realiza as alterações acima é denominado conversor PAL digital, e está ilustrado na **figura 95**.

O ponto de tomada do burst NTSC é cancelado e os burst PAL B negativo e B positivo são gerados 1 gate antes e 1 gate após, respectivamente.

O sinal R torna-se R+ e através do gate é produzido o R.

Estes novos sinais entram na chave PAL que produzirá todas as inversões necessárias.







## CIRCUITOS PARA TRANSCODIFICAR VÍDEO-GAME

Na **figura 96** vemos um circuito que pode ser usado para transcodificar vídeo-game. Utilizou-se o CI 4013 para gerar a chave PAL e gerar a inversão linha a linha do burst e da componente R-Y.

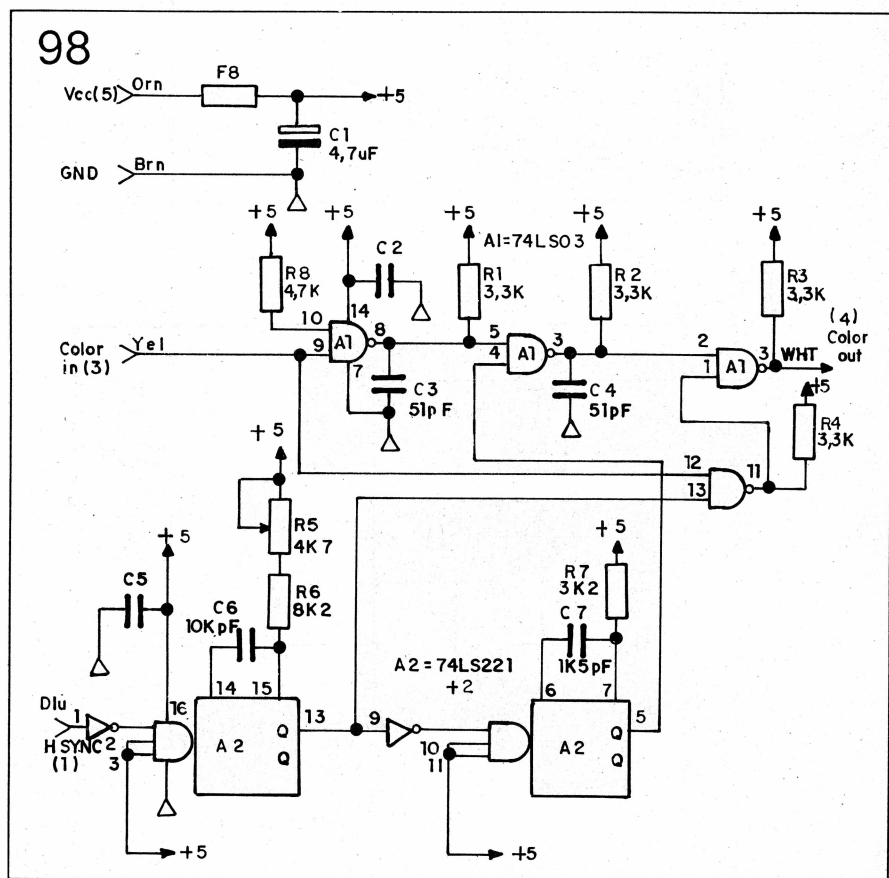
Nota: T1 e T2 são transistores BC 548.

### ATARI

Na **figura 97** vemos o circuito para transcodificar o Atari.

Além disto, deve-se substituir o cristal, colocando um trimer em série.

Outro circuito está ilustrado na **figura 98**. O flip-flop gera a chave PAL, enquanto que as portas NAND provocam os defasamentos digitais do sinal de cor.



## Índice

Apresentação	
<b>Capítulo 1 - INTRODUÇÃO AO SOFTWARE DO VÍDEO-GAME</b> .....	4
Introdução ao Hardware do Vídeo-game.....	8
Funcionamento do Vídeo-game (Diagramas em Blocos).....	20
Descrição Do Atari.....	35
Descrição do Odissey.....	45
Descrição do Mega Drive (Sega).....	46
Descrição do Nintendo.....	52
Descrição do Mega-CD-ROM.....	52
Conserte Seu Próprio Vídeo-game.....	53
Testes Por Componentes.....	55
Transcodificação de Vídeo-games.....	60

• • •



editora  
**FITTIPALDI**

