

Jozué Vieira Filho

Sistema de Aquisição de Dados controlado por
Microprocessador para Sinais na Faixa de Frequência
de 0 a 50 kHz

Dissertação apresentada à Faculdade de
Engenharia Elétrica da Universidade
Estadual de Campinas como parte dos
requisitos exigidos para obtenção do
título de MESTRE EM ENGENHARIA ELÉTRICA.

Este exemplar corresponde à relação final da tese
defendida por Jozué Vieira Filho
e aprovada pela Comissão
Jugadora em 14/02/91.

Orientador

du gunt

Orientador: Prof. Dr. José Geraldo Chiquito

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE COMUNICAÇÕES

Campinas - SP
Fevereiro 1991

82/9102/91

UNICAMP
BIBLIOTECA CENTRAL

Este trabalho tornou-se possível graças às instituições
financiadoras de pesquisas CNPq e FAPESP.

AGRADECIMENTOS

Ao professor Dr. José Geraldo Chiquito, pela orientação sincera e objetiva durante o decorrer deste trabalho. Sem dúvidas, o grande responsável pelos resultados obtidos.

Ao professor Dr. Wagner Chiepa Cunha, do Instituto Tecnológico de Aeronáutica (ITA). Co-orientador do desenvolvimento de vários circuitos digitais presentes neste trabalho e do "software" de operação do sistema, o professor Chiepa não se incomodou com as viagens entre S.J. dos Campos e Campinas, recebeu-me com paciência e atenção quando tive de ir à S.J. dos Campos e esteve sempre a disposição nas várias etapas do trabalho.

Aos professores da UFPb, Dr. Wilson Guerreiro, Msc. Rubem Alves e Msc. José Ewerton, pelo incentivo dado no final da graduação, ajudando-me a decidir pelo curso de mestrado.

Ao grande amigo Edson José Nagle, que esteve comigo desde o início da minha chegada em Campinas, incentivando-me nos momentos difíceis e, no final, ainda releu linha a linha o trabalho aqui apresentado, fez sugestões, corrigiu erros de português, etc. Valeu!

Aos estagiários, hoje engenheiros, Edson Bortolli e Paulo Bernardes, que tiveram participação importante nos desenvolvimentos e testes de vários circuitos presentes neste trabalho.

À Kenia Pena de Souza, amiga e companheira de laboratório, que, além de ter participado na elaboração de alguns circuitos, também fez questão de revisar todo o texto deste trabalho.

À Sonia Regina Paulino, figura humana maravilhosa, que teve papel importante nos momentos mais difíceis durante a redação deste trabalho.

A todos os amigos da Pós-Graduação que conviveram amigavelmente comigo durante três anos, nas festas, no futebol, enfim, em tudo que foi importante para que a vida acadêmica fosse vivida mais descontraidamente.

à minha família

RESUMO

Este trabalho descreve o projeto e implementação de um Sistema de Aquisição de Dados para sinais na faixa de frequência de 0 a 50 kHz, que possui como características principais flexibilidade e versatilidade.

As unidades de processamento do sinal analógico permitem diferentes opções, configuráveis através de "software". Isto inclui filtros anti-aliasing, amplificadores, acoplamento, resolução das unidades de conversão A/D e D/A, taxa de amostragem, etc. Para armazenar os dados digitalizados existe uma unidade de memória dinâmica de 2 Mbytes, expansível a 16 Mbytes.

O sistema foi construído de forma auto-suficiente, ou seja, sua capacidade de operação independe de microcomputadores. Este objetivo foi alcançado graças à incorporação de uma unidade para controle de aquisição de dados baseada no microprocessador 8088. As operações de conversões e a configuração das diferentes funções podem ser feitas através de uma unidade de teclado/display, no modo local, ou por meio de um computador externo acoplado aos "ports" de comunicação, no modo remoto.

O protótipo resultante, em funcionamento no Laboratório de Processamento de Sinais do DECOM/FEE/UNICAMP, está sendo de grande importância para estudos na área de processamento digital de sinais de voz e áudio.

ÍNDICE

1 - INTRODUÇÃO	1
2 - SISTEMAS DE AQUISIÇÃO DE DADOS	3
2.1 - Sistema Básico de Aquisição de Dados	3
2.1.1 - Condicionamento do sinal analógico	5
2.1.2 - Conversão analógico-digital	6
2.1.3 - Conversão digital-analógico	9
2.1.4 - Armazenamento, controle e processamento	10
2.2 - Arquiteturas Básicas de Sistemas de Aquisição de Dados	12
2.2.1 - Sistemas de placa de aquisição acopladas a microcomputador	12
2.2.2 - Sistemas de aquisição de dados auto-suficientes	15
2.3 - Transferência e Armazenamento de Dados em Sistemas de Aquisição de Dados	17
2.3.1 - Transferência de dados com controle da CPU	18
2.3.2 - Transferência de dados por DMA	19
2.3.3 - Dispositivos de armazenamento mais usados em sistemas de aquisição de dados	20
2.4 - Aplicações de Sistemas de Aquisição de Dados	22
2.4.1 - Instrumentos digitais	22
2.4.2 - Sistemas para controle de processos	24
2.4.3 - Sistemas de propósito geral	26

3 - CARACTERÍSTICAS GERAIS E UNIDADES ANALÓGICAS	29
3.1 - Uma Visão Geral	29
3.2 - Especificações	32
3.3 - Unidades de Aquisição do Sinal Analógico	34
3.3.1 - Unidade de controle de amplitude do sinal analógico	35
3.3.2 - Circuito para seleção de faixas do sinal analógico	36
3.3.3 - Circuito "sample-hold" e conversão A/D	38
3.3.4 - Conversão D/A e reconstrução do sinal analógico	41
3.3.5 - Reprodução de sinais de áudio e circuito de alimentação	43
4 - UNIDADES DE CONTROLE E MANIPULAÇÃO DE DADOS	59
4.1 - Unidade do Processador	61
4.1.1 - Unidade de controle de processamento (CPU)	61
4.1.2 - Unidade de comunicação e entrada/saída	63
4.2 - Unidade de Controle de DMA	65
4.2.1 - O controlador de DMA 8237-2	66
4.2.2 - Circuito de DMA do SAD	68
4.2.3 - Operações de DMA no SAD	70
4.3 - Unidade de Armazenamento de Dados	71
4.3.1 - Sistema de memória e endereçamento	72
4.3.2 - Circuito de endereçamento	72
4.3.3 - Sistema de "refresh"	73

4.4 - Unidade de Controle de Conversões A/D, D/A e das Funções Analógicas do SAD	74
4.4.1 - Circuito de seleção das funções analógicas e de geração de taxas de amostragem programáveis	74
4.4.2 - Lógica de geração de sinais de controle para as conversões A/D.	76
4.4.3 - Lógica de geração de sinais de controle para as conversões D/A.	78
5 - "SOFTWARE" DE OPERAÇÃO DO SAD	98
5.1 - O "Software" Monitor SBSAD.	98
5.1.1 - Subrotinas principais	102
5.1.2 - Subrotinas de uso geral	104
5.2 - O "Software" de Comunicação Operação SCOMSAD.	106
5.2.1 - Bloco de controle das funções do SAD.	106
5.2.2 - Bloco de manipulação do "hardware" do SAD	108
6 - CONSIDERAÇÕES FINAIS.	114
6.1 - Aspectos Práticos de Implementação.	114
6.2 - Resultados.	115
6.3 - Pontos Passíveis de Melhoramentos	116
REFERÊNCIAS BIBLIOGRÁFICAS	124
APÊNDICE A	126
APÊNDICE B	132

CAPÍTULO 1 - INTRODUÇÃO

1.1 - IMPORTÂNCIA DO TRABALHO

O avanço da microeletrônica veio possibilitar o uso e aprimoramento das técnicas de processamento digital de sinais. A eficiência destas técnicas possibilita as mais complexas operações, que seriam difíceis, ou até mesmo impossíveis, de serem realizadas com os métodos analógicos.

Como consequência, os Sistemas de Aquisição de Dados [1] têm obtido cada vez mais importância. Estes sistemas são muito utilizados na aquisição de sinais dos mais variados tipos, possibilitando o estudo e desenvolvimento de algoritmos para processamento digital, registro de sinais, análise e processamento em geral de sinais analógicos.

Entre os sistemas de aquisição de dados mais utilizados estão os "Data-Loggers". Estes se caracterizam por possuírem várias entradas analógicas, normalmente multiplexadas para um único conversor A/D, e são usados quando se quer registro ou análise de processos envolvendo variáveis físicas como força, temperatura, pressão, etc. [2]

As aplicações atuais dos Sistemas de Aquisição de Dados são das mais variadas, indo desde instrumentos de análise de formas de ondas, como os osciloscópios e multímetros digitais, até as complexas estações de controle de processos.

A maioria dos sistemas de aquisição de dados aparecem hoje relacionados de alguma forma com microcomputadores. Esta relação determina praticamente dois tipos de implementação: Placas de Aquisição de Dados incorporáveis a microcomputadores do tipo PC e Sistemas de Aquisição de Dados Auto-suficientes.

As placas de aquisição se caracterizam por apresentarem custo relativamente baixo, pois utilizam todos os recursos do microcomputador ao qual é acoplada. A nível nacional é o tipo mais usado em laboratórios, sendo também o único modelo disponível comercialmente, apesar de seu uso não ser ainda muito difundido. Os sistemas de aquisição Auto-suficientes possuem maior custo e complexidade em termos de circuitos, existindo, na grande maioria dos casos, uma unidade de controle das operações de aquisição de dados baseada em microprocessador. É o tipo de sistema mais adequado para acompanhar as rápidas mudanças que ocorrem nos mais variados campos de aplicações. No Brasil,

estes sistemas ainda não são disponíveis comercialmente. Por esta razão, decidiu-se desenvolver e implementar um sistema auto-suficiente.

O desenvolvimento deste trabalho envolveu o estudo de diferentes tipos de sistemas de aquisição de dados e suas aplicações, resultando na implementação de um sistema com características bastante satisfatórias a nível nacional.

1.2 - OBJETIVOS E ORGANIZAÇÃO DO TEXTO

O objetivo deste trabalho é dar uma visão geral sobre sistemas de aquisição de dados e descrever o sistema desenvolvido e implementado. Os capítulos seguintes estão assim organizados:

Capítulo 2 - São introduzidos os sistemas de aquisição de dados, enfocando-se os diferentes tipos de arquitetura, as unidades principais, os métodos de transferência de dados e as várias áreas de aplicações;

Capítulo 3 - É iniciada a apresentação do sistema desenvolvido, descrevendo-se as características gerais e as unidades de processamento do sinal analógico a ser digitalizado;

Capítulo 4 - É descrita a unidade de controle das operações de aquisição de dados, destacando-se as unidades do processador, de controle de DMA, de armazenamento de dados e de controle das operações das unidades analógicas;

Capítulo 5 - São apresentadas as unidades básicas do "software" desenvolvido para operação do sistema no modo remoto;

Capítulo 6 - São discutidos os aspectos finais da implementação, são sugeridos possíveis melhoramentos e são apresentados alguns sinais digitalizados no sistema.

CAPÍTULO 2 - SISTEMAS DE AQUISIÇÃO DE DADOS

2.1 - SISTEMA BÁSICO DE AQUISIÇÃO DE DADOS

A concepção de um sistema de aquisição de dados segue um esquema básico (figura 2.1), com possíveis variações que dependem quase sempre das aplicações.

Normalmente, a operação inicial consiste em um condicionamento do sinal analógico com relação a sua amplitude e largura de faixa. Um ajuste de amplitude condiciona o sinal à gama de operação do conversor A/D. A seleção de uma determinada faixa de frequência, feita através de uma filtragem passa-baixas, evita a distorção por "aliasing".

Na unidade de conversão A/D ocorre a transformação do sinal analógico (ou contínuo) para discreto, com o valor das amostras representadas na forma digital.

Os dados digitais gerados pelo conversor A/D podem ser utilizados basicamente de duas formas: a primeira envolve um armazenamento para posterior processamento. É o caso, por exemplo, dos sistemas usados para auxiliar no desenvolvimento de algoritmo de processamento digital de sinais, dos "Data-Loggers", etc. A outra opção consiste de um processamento imediato dos dados digitais, que é característica básica dos sistemas para processamento em tempo real, como, por exemplo, os sistemas usados em controle de processos.

O retorno do sinal para a forma analógica, quando necessário, é feito através de uma conversão D/A, seguida por uma filtragem passa-baixas. A unidade de saída não é usada, por exemplo, nos instrumentos de medição digital (voltímetros, freqüencímetros, etc.).

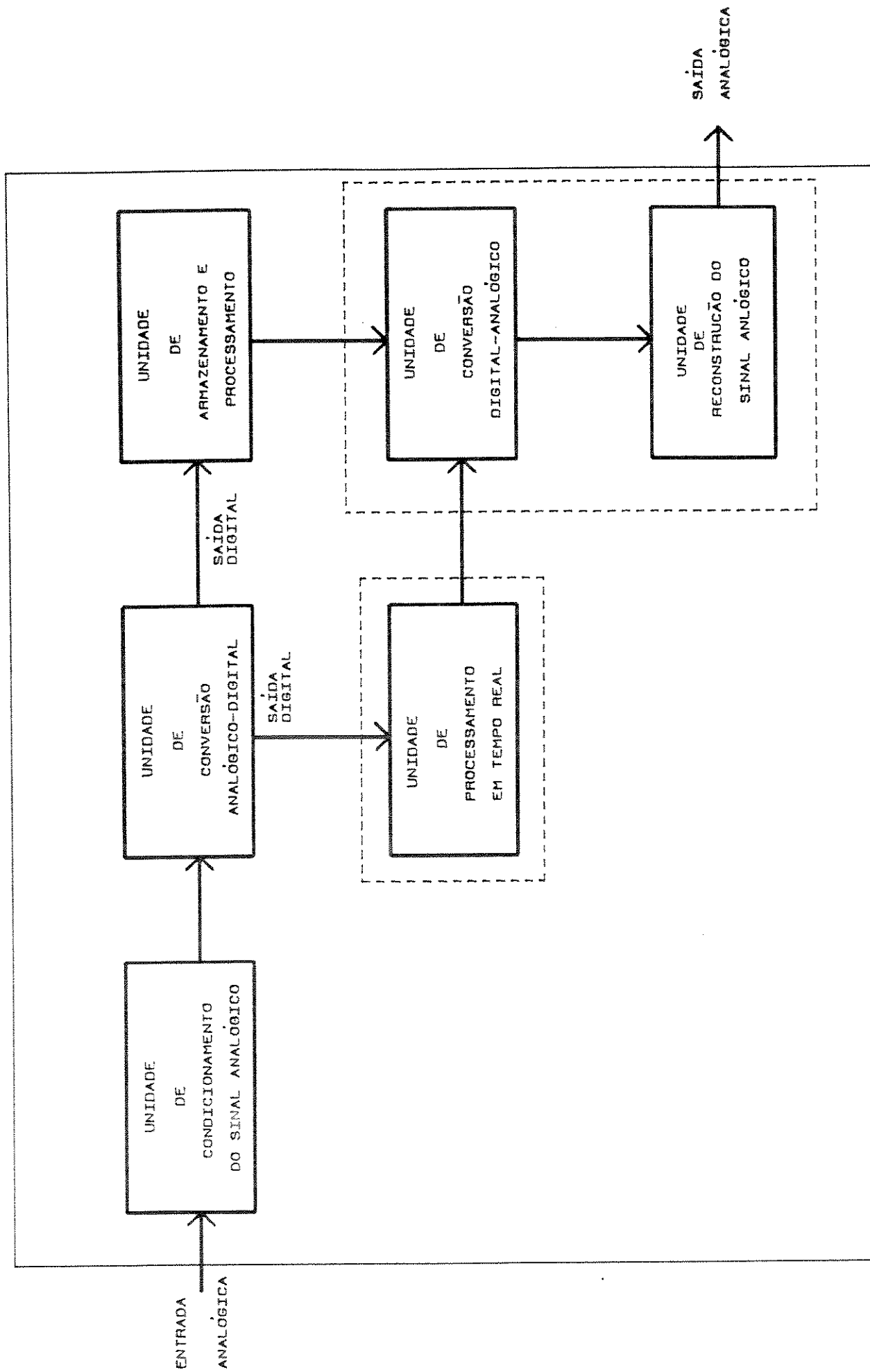


FIGURA 1.1 - SISTEMA BÁSICO DE AQUISIÇÃO DE DADOS

2.1.1 - Condicionamento do Sinal Analógico

A - Controle da Amplitude

São comuns as operações de aquisição de dados para análise de variáveis físicas. Para tal são utilizados sensores adequados de modo a detectar um grande número destas variáveis, como pressão, temperatura, força, etc. A maioria destes sensores apresentam sinais com níveis muito baixos, exigindo quase sempre uma amplificação. Já outros sinais, como voz e áudio provenientes de algum tipo de sistema de reprodução, podem apresentar níveis elevados.

O objetivo de um controle da amplitude é fazer com que estes sinais apresentem níveis padrão, tais que a faixa de operação do conversor A/D seja utilizada de forma adequada. Um sinal na entrada do conversor com amplitude muito baixa será prejudicado pelo ruído de quantização. Por outro lado, sinais com amplitude muito elevada podem levar a uma saturação do conversor. Para evitar estas situações, são utilizados amplificadores (ou atenuadores) adequados. Alguns sistemas usam circuitos amplificadores com ganhos variáveis, comumente denominados de PGA (Programmable Gain Amplifier).

B - Filtragem

A amostragem de sinais analógicos deve ser feita obedecendo o teorema da amostragem, que diz que um sinal contínuo no tempo e limitado em frequência pode ser discretizado e posteriormente recuperado fielmente, se e somente se suas amostras forem geradas a uma taxa no mínimo igual a duas vezes a maior componente de frequência presente no sinal. A violação do teorema da amostragem pode levar a uma distorção denominada "aliasing", ilustrada na figura 2.2(d), que torna impossível a recuperação exata do sinal.

A maneira mais simples de se evitar o "aliasing" é fazer uma superamostragem. Esta solução não é, no entanto, a mais conveniente. Ela exige sistemas rápidos com grande capacidade de armazenamento, o que pode limitar de maneira significativa as aplicações de sistemas de aquisição de dados.

A solução mais utilizada é a limitação da faixa de interesse do sinal através de uma filtragem passa-baixas, compatível com a taxa de

amostragem. O filtro usado para realizar esta função deve ter uma frequência de corte igual à maior frequência de interesse. O projeto deste filtro é da maior importância, pois tem influência direta na qualidade do sistema.

2.1.2 - Conversão Analógico-Digital (A/D)

A maneira idealizada de se obter um sinal discreto a partir de um sinal contínuo, é multiplicar este último por uma seqüência de impulsos regularmente espaçados, como mostrado na figura 2.2.

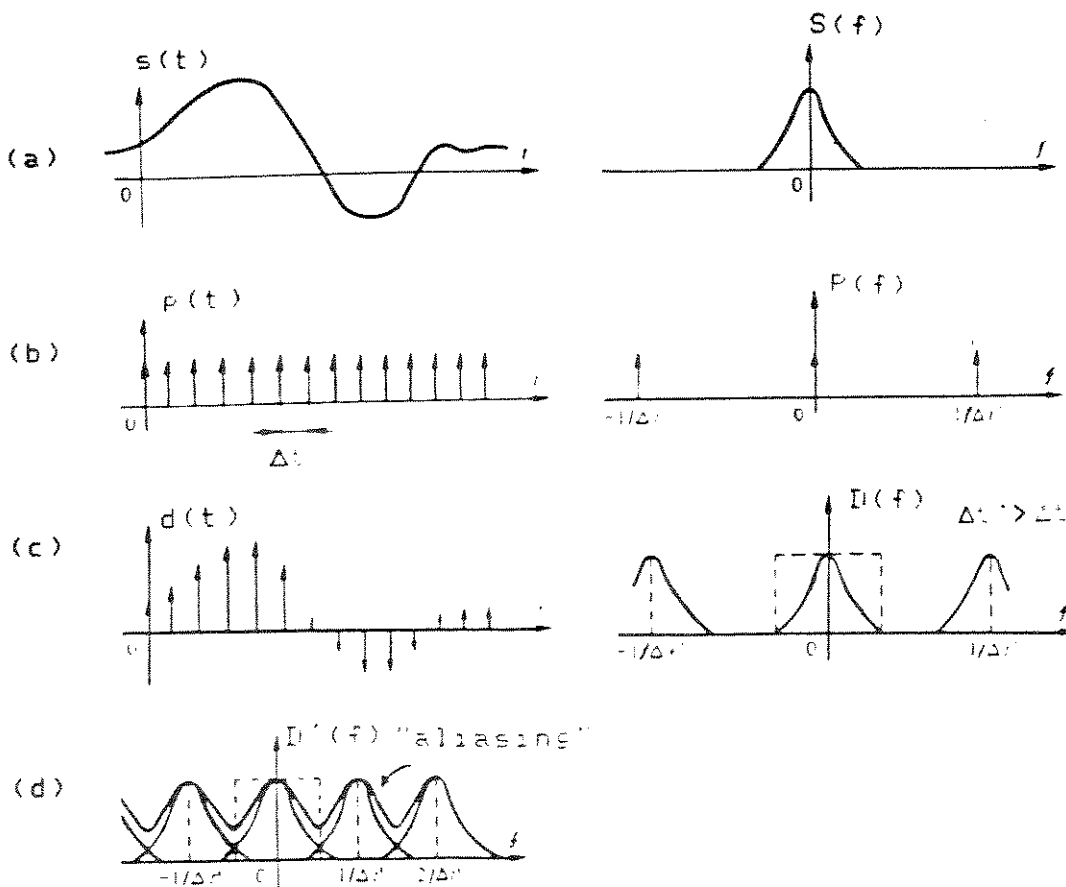


FIGURA 2.2 - Amostragem Ideal de um Sinal (a) sinal a ser amostrado (b) seqüência de impulsos para amostragem (c) sinal amostrado (d) espectro de frequência do sinal quando ocorre a distorção por "aliasing"

Na prática o processo acontece de forma diferente, devido à impossibilidade de geração de impulsos. O valor amostrado é representado por pulsos, espaçados de acordo com o período de amostragem, e está normalmente relacionado a uma conversão analógico-digital. O pulso de amostragem é feito por um circuito denominado "sample-hold" e a transformação do valor analógico para digital por um conversor A/D.

A - Circuito "Sample-Hold"

Todos os conversores A/D levam um determinado tempo para transformar um valor analógico em digital. Assim, sinais variando muito rapidamente podem apresentar incertezas nas conversões [3], como mostrado na figura 2.3. Este erro, ou incerteza, praticamente desaparece quando se trabalha com dispositivos muito rápidos (com tempo de conversão da ordem de alguns nanossegundos) que, no entanto, são de custo elevado e apresentam resolução limitada (veja item 2.1.4).

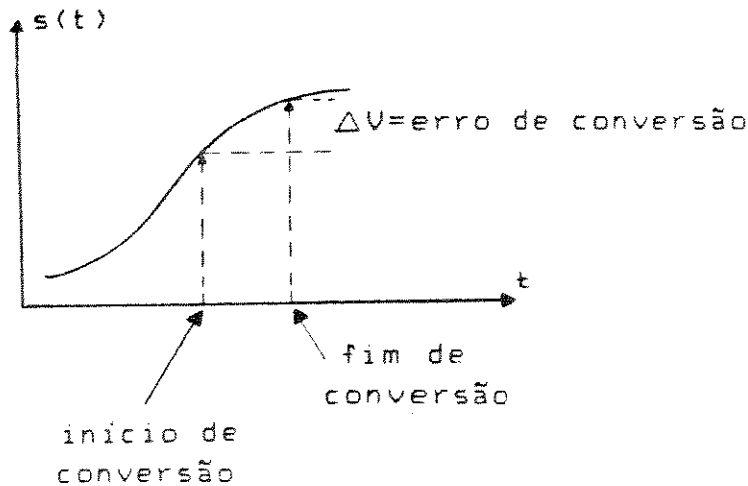


FIGURA 2.3 - Erro causado por uma variação rápida de sinal na entrada de um conversor A/D

A utilização direta do conversor A/D, só é possível se a incerteza produzida durante a conversão ficar abaixo da resolução do conversor.

Uma análise feita em [4], mostra que, para um conversor de n bits, tendo um tempo de conversão t_c , a variação máxima permitida para o sinal é:

$$\left[\frac{dS(t)}{dt} \right]_{\max} = \left(V_{\text{oscila}} / 2^n t_c \pi \right) \quad (1)$$

O valor V_{oscila} é a máxima variação permitida para o sinal na entrada do conversor. Por exemplo, considerando uma senóide de amplitude A e frequência f .

$$S(t) = A \text{ sen}(2 \pi f t) \Rightarrow \frac{dS(t)}{dt} = 2 \pi f A \cos(2 \pi f t)$$

$$\Rightarrow \left[\frac{dS(t)}{dt} \right]_{\max} = 2 \pi f A$$

Comparando este resultado com a relação (1), tem-se $f \leq 1/2^n t_c \pi$.

Para um conversor de 12 bits, com um tempo de conversão de 10 microssegundos, a máxima frequência permitida para a senóide é :

$$f_{\max} = 1/2^n t_c \pi \implies f_{\max} = 7,77 \text{ Hz}$$

Com o valor calculado as aplicações seriam muito restritas. Para contornar este problema utiliza-se um circuito "sample-hold" entre o sinal e o conversor A/D usado. A função do circuito é amostrar o sinal e reter o valor da amostra durante o intervalo de conversão. No cálculo da frequência máxima permitida, no lugar de t_c deve-se agora usar um valor alternativo t_{m} , que é o tempo de captura do circuito S/H. Normalmente este tempo é da ordem de alguns nanossegundos. Para o exemplo mostrado anteriormente, se for utilizado um circuito S/H com um tempo de captura de 1 ns, poderá se ter uma senóide com uma frequência máxima de:

$$f_{\max} = 1/2^n t_{\text{m}} = 77,7 \text{ kHz.}$$

Devido à grande utilização dos circuitos "sample-hold", vários fabricantes o produzem na forma integrada.

B - Conversores A/D

A escolha de conversores A/D é fator importante na construção de sistemas de aquisição de dados, e depende fundamentalmente da aplicação.

Existe uma variedade de conversores A/D na forma integrada, utilizando as técnicas mais eficientes para este processo [3], que são: Dupla-Rampa, Aproximação Sucessiva e Comparadores Paralelos.

A avaliação dos conversores A/D é feita pela análise de várias características, das quais duas são provavelmente as mais importantes: Resolução e Tempo de Conversão. A resolução de um conversor é o resultado de uma aproximação na transformação de um valor analógico em digital, sendo definido normalmente pelo número de bits utilizado para representar o valor analógico. A resolução típica dos conversores A/D está entre 6 e 16 bits. O tempo de conversão é definido como o tempo necessário para o conversor apresentar um valor digital estável, correspondente a uma entrada analógica, a partir de um disparo de conversão. A definição da resolução e do tempo de conversão na escolha de um conversor depende da aplicação associada ao custo.

Em sistemas de aquisição de dados para operar com sinais de baixas frequências (menor que 1 kHz), geralmente é escolhido um conversor de dupla-rampa. A vantagem obtida é uma boa resolução aliada a um baixo custo. Para sistemas que atuam com sinais na faixa de algumas dezenas de kHz, como voz, áudio, etc., normalmente são utilizados conversores que operam com a técnica de aproximação sucessiva. Neste caso, obtém-se sistemas de boa resolução com tempo de conversão relativamente pequeno. A outra opção são os sistemas para operação com sinais de alta frequência, que utilizam conversores com comparadores paralelos. Esta técnica permite obter-se conversores muito rápidos, mas que apresentam resolução limitada (normalmente abaixo de 8 bits).

2.1.3 - Conversão Digital-Analógica (D/A)

Nos sistemas de aquisição de dados usados em controle, ou naqueles em que é preciso analisar resultados de algum tipo de processamento digital, é necessário a implementação de uma unidade de saída.

A primeira função da unidade de saída é transformar o valor digital das amostras de um determinado sinal para analógico, tarefa realizável por um conversor D/A.

Com a transformação feita pelo conversor D/A, ainda não se tem o sinal na sua forma original. Nesta etapa o sinal é composto por uma seqüência de pulsos de largura igual ao período de amostragem, como ilustrado na figura 2.4. Em freqüência, isto equivale a uma distorção espectral do tipo $\text{sen}(x)/x$.

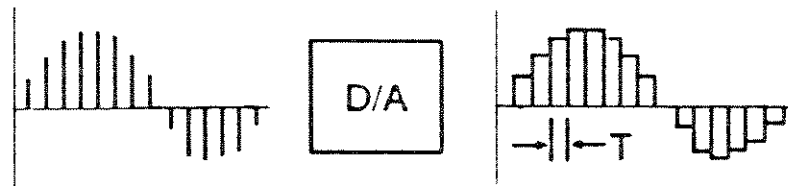


FIGURA 2.4 - Conversão Digital-Analógica

O método mais utilizado para eliminar a distorção gerada pelo processo de conversão D/A, assim como remover as componentes de alta freqüência, é fazer uma filtragem passa-baixas adequada. O filtro passa-baixas usado para esta função normalmente é equivalente ao filtro usado para limitação da faixa de freqüência do sinal para conversão A/D. A diferença é que a resposta em freqüência do filtro, na faixa de passagem, deve compensar a função $\text{sen}(\pi fT)/(\pi fT)$, onde f é a faixa do sinal amostrado (normalmente dado pelo FPB de entrada) e T o período de amostragem.

2.1.4 - Armazenamento, Controle e Processamento

Em sistemas para processamento em tempo real, os dados gerados pelo conversor A/D são utilizados diretamente por um processador de sinais dedicado, que faz o controle e o processamento simultaneamente com a aquisição. Neste caso, as unidades de armazenamento, quando necessárias, não costumam ser de grande capacidade. Este tipo de sistema tem sido muito utilizado, principalmente com o desenvolvimento de

processadores digitais de grande velocidade, conhecidos como DSP's (Digital Signal Processors). Um exemplo é o processador TMS320C-25, da TEXAS INSTRUMENTS [5], que possui um ciclo de instrução de 100 ns.

Os sistemas que não operam em tempo real necessitam que os dados gerados pelo conversor A/D sejam armazenados previamente, para depois serem processados. Neste caso, a capacidade de armazenamento passa a ser uma característica importante do sistema de aquisição de dados.

Controle e processamento em sistemas que não operam em tempo real podem ser feitos de várias maneiras. Nas seções seguintes, onde serão apresentadas as arquiteturas dos sistemas de aquisição de dados, se fará uma discussão sobre as opções de controle e processamento.

2.2 - ARQUITETURAS BÁSICAS DOS SISTEMAS DE AQUISIÇÃO DE DADOS

A implementação de sistemas de aquisição de dados pode ser feita basicamente a partir de dois tipos de arquitetura: Placas de Aquisição Acopladas a Microcomputador, que tem como grandes atrativos a simplicidade e o baixo custo, e os Sistemas de Aquisição Auto-suficientes, que independem de microcomputadores e apresentam grande flexibilidade.

2.2.1 - Sistema de Placas de Aquisição acopladas a Microcomputador

A abordagem mais econômica na construção de um sistema de aquisição de dados é a incorporação de uma placa de conversão A/D - D/A num dos "slots" de expansão de um microcomputador. A idéia principal neste caso é aproveitar toda a estrutura de "hardware" e "software" do microcomputador, facilitando bastante o desenvolvimento do sistema [6]. Este tipo de implementação é dirigido principalmente a aplicações que envolvem sinais na faixa de algumas unidades de kHz.

Uma placa de aquisição de dados típica possui basicamente unidades de conversão A/D e D/A e de condicionamento de sinal, além da interface com o computador. A figura 2.5 apresenta o diagrama de um sistema de aquisição de dados de uso geral. O microcomputador faz todo o controle das operações de conversão, além de oferecer outros recursos necessários para a aquisição de dados, como memória, sistema de DMA, etc.

A operação de uma placa de aquisição de dados é feita usando as funções de entrada e saída (E/S) ou de acesso à memória do microcomputador. A operação como um dispositivo de E/S é a mais utilizada, pois normalmente os microcomputadores possuem endereços específicos destinados a esta função. No caso de se utilizar a funções de acesso à memória, é necessário criar um sistema de endereçamento específico para a placa dentro do espaço de memória do microcomputador. Além de diminuir o espaço de memória, cria-se também a necessidade de cuidar para que nenhum "software" a ser executado no microcomputador utilize o espaço de memória reservado.

A presença de um microcomputador possibilita a implementação de sistemas eficientes. Por outro lado, o fato de se instalar a placa de

aquisição de dados no ambiente do microcomputador, pode criar alguns inconvenientes e/ou limitações. Isto é tanto mais acentuado quanto mais rápido se quer fazer aquisições ou quanto menor for o nível do sinal analógico de entrada. Os pontos considerados críticos nos sistemas de aquisição com placas são:

1 - Baixa imunidade ao ruído

A conexão direta destas placas pode criar problemas com o ruído introduzido pelo computador, gerado por seus circuitos digitais de alta velocidade, osciladores, microprocessadores, co-processador, etc., que normalmente operam com nível TTL (0 e 5 volts). Como os sinais analógicos geralmente apresentam baixa amplitude, eles podem ser completamente degradados pelo ruído gerado no ambiente do microcomputador.

Para combater este efeito, quase sempre se faz uso de amplificadores para elevar o nível do sinal. Vale ressaltar, no entanto, que os próprios amplificadores podem ser vítimas deste ruído, exigindo cuidados especiais, como sistema de aterramento e ambiente elétrico isolado. Assim, a simplicidade normalmente buscada neste tipo de arquitetura pode ser prejudicada pela necessidade de circuitos externos, que possam garantir níveis adequados para operações no ambiente do computador.

2 - Velocidade limitada de transferência dos dados digitalizados

Como as placas de aquisição geralmente não possuem memória própria, os dados gerados a partir da conversão A/D são armazenados nas unidades de memória do microcomputador utilizado. No caso de se usar a memória RAM do sistema, cria-se uma limitação de capacidade, que pode prejudicar algumas aplicações com sinais de faixa elevada. Uma solução alternativa são as unidades de memória de massa, normalmente os discos rígidos.

As unidades de discos apresentam como principal vantagem a grande capacidade de armazenamento, que possibilita obter-se arquivos longos com os dados digitalizados. Entretanto, a forma de armazenamento em disco rígido também impõe limitações na velocidade de transferência dos dados, devido às restrições impostas principalmente pelo sistema mecânico da cabeça de leitura/escrita destes dispositivos. A maioria dos sistemas de discos rígidos apresenta um tempo de acesso médio da ordem de alguns milissegundos.

3 - Flexibilidade limitada pelos recursos do microcomputador

Uma outra limitação dos sistemas com esta arquitetura é o nível de flexibilidade, principalmente pelo espaço restrito oferecido pelo computador. Isto exige que cada aplicação seja devidamente estudada antes que se faça a implementação da placa. Como exemplo, pode-se citar dificuldades para expansão de memória, muito importante em determinadas aplicações.

A grande aceitação das placas de aquisição está, porém, fazendo com que este tipo de arquitetura passe por constantes aperfeiçoamentos, atingindo níveis de desempenho altamente satisfatórios [7].

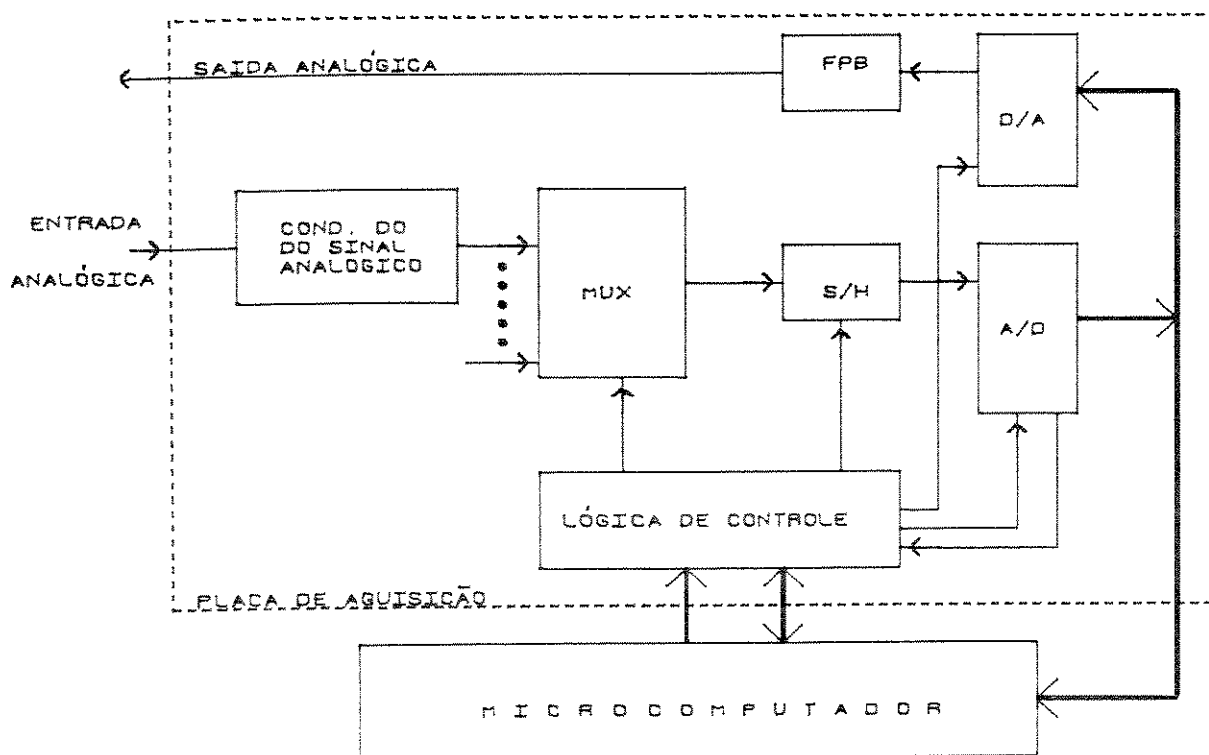


FIGURA 2.5 - Placa de Aquisição de Dados acoplada a Microcomputador

2.2.2 - Sistemas de Aquisição de Dados Auto-suficientes

Os Sistemas de Aquisição de Dados Auto-suficientes podem operar independentemente de microcomputadores, prevendo-se, porém, interfaces de conexão com outros sistemas. Os sistemas Auto-suficientes, conforme mostra o diagrama de blocos simplificado da figura 2.6, possuem sistema próprio de controle e armazenamento, o que possibilita operações de aquisição de dados com mais eficiência e flexibilidade.

O controle das operações podem ser feito de duas formas: através de circuitos lógicos convencionais [8] ou com microprocessador [9]. No primeiro caso, toda a parte de controle é previamente definida e implementada através de portas lógicas comuns. Este tipo de circuito cria uma grande limitação, que é a inflexibilidade imposta ao sistema. Assim, se for preciso fazer qualquer alteração, poderá haver a necessidade de uma completa modificação na parte de controle. No caso de se utilizar um microprocessador, o sistema se torna "inteligente", podendo ter todas as suas funções manipuladas por "software". Uma outra facilidade encontrada nos sistemas dotados de microprocessador é a possibilidade de comunicação com sistemas de computação, hoje imprescindível nos sistemas desta natureza.

A arquitetura do tipo auto-suficiente apresenta várias vantagens, das quais pode-se destacar:

- 1 - O sistema pode atuar na tarefa de aquisição de dados distante de centros computacionais, facilitando as operações de campo;
- 2 - Todos os inconvenientes do ambiente elétrico de um microcomputador podem ser cuidados de forma adequada, possibilitando um alto índice de integridade dos sinais a serem adquiridos. Por exemplo, pode-se isolar os circuitos analógicos dos digitais, providenciar blindagens das partes analógicas, como amplificadores e filtros usados na unidade de condicionamento de sinal, etc.,
- 3 - Capacidade de se trabalhar com praticamente qualquer tipo de sinal analógico. Com esta arquitetura pode-se usar uma unidade de memória semicondutora de grande capacidade, o que possibilita armazenamento de sinais de frequências mais elevadas, com tempo suficiente para se fazer análise e processamento digital;
- 4 - Facilidade de centralizar operações de aquisição de dados multipontos. Por exemplo, numa indústria onde existem pontos de controle distantes, pode-se conectar vários destes sistemas a um

computador central. As informações podem ser transferidas através de um enlace de comunicação, como por exemplo uma interface serial RS-232.

Os sistemas Auto-suficientes apresentam grande penetração nos mais variados tipos de aplicações. Um exemplo é a estação Sona-Graph [10]. Este sistema permite se trabalhar com sinais na faixa de frequência de 0 a 32 kHz, com a análise das amostras praticamente em tempo real.

A maior dificuldade encontrada no desenvolvimento de sistemas auto-suficientes, além da sua complexidade em termos de circuito, é o custo. Um sistema auto-suficiente nada mais é que um microcomputador dedicado, que utiliza componentes de custo relativamente elevado.

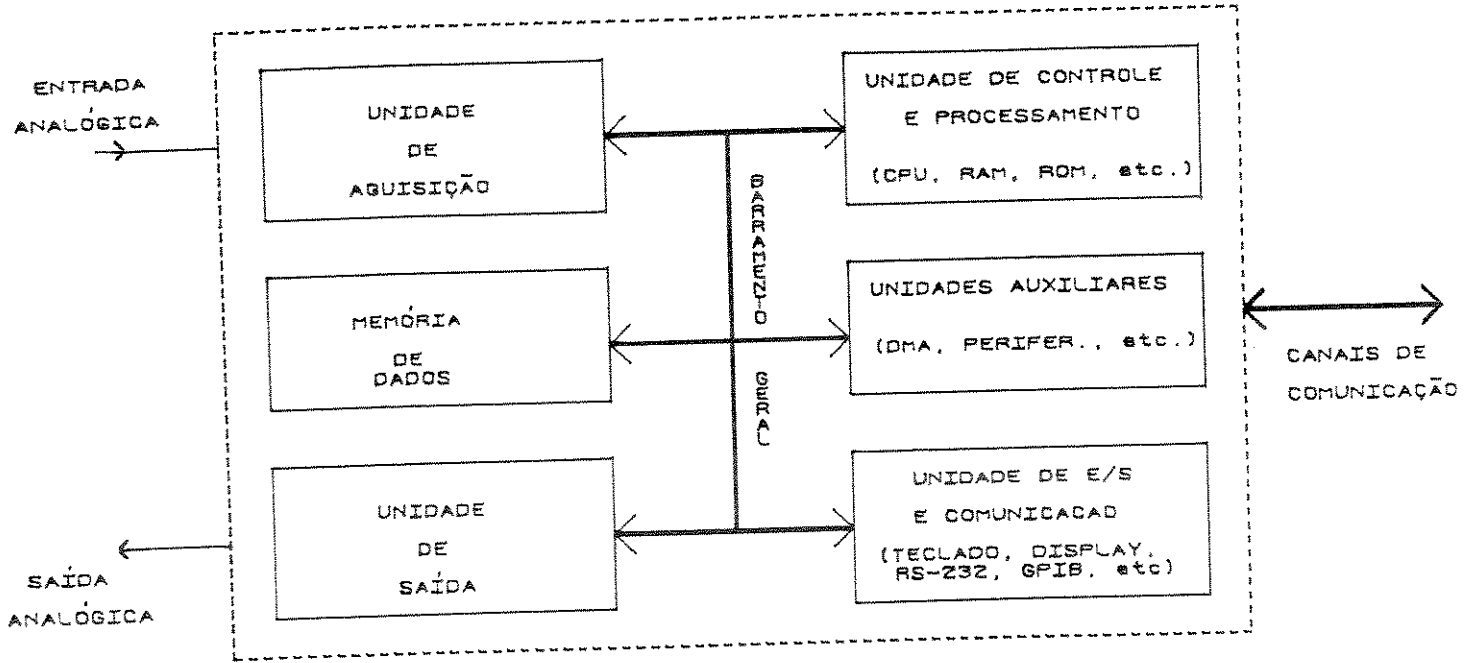


FIGURA 2.6 - Sistema de Aquisição de Dados tipo Auto-suficiente

2.3 - TRANSFERÊNCIA E ARMAZENAMENTO DE DADOS EM SISTEMAS DE AQUISIÇÃO

Um fator importante nos sistemas de aquisição de dados é a velocidade de transferência dos dados gerados pelo processo de conversão A/D. Como discutido no item 2.2.2, o primeiro limitante na velocidade de geração de dados é o conversor A/D.

A velocidade está associada diretamente com o método usado para transferir os dados e com o dispositivo de armazenamento. Os métodos usados para transferir dados de um dispositivo para outro são dois: Transferência por controle da CPU e Transferência por Acesso Direto à Memória (DMA). A figura 2.7 mostra o fluxo de dados para os dois métodos [11]. No primeiro caso os dados passam necessariamente pelo processador, enquanto que por DMA os dados são transferidos diretamente entre a memória e o dispositivo de E/S

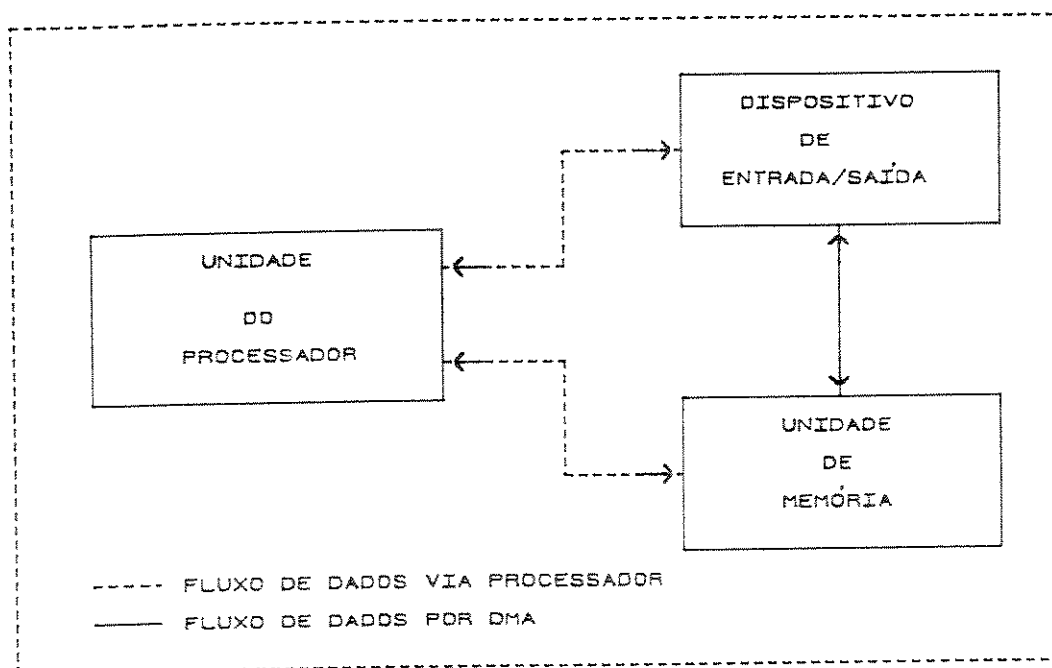


FIGURA 2.7 - Fluxo de Dados nas transferências com controle da CPU e por DMA

Para armazenamento dos dados existem várias opções, mas apenas duas mostra-se mais convenientes: Disco Rígido e Memória Semicondutora. Dependendo da aplicação do sistema, o método de transferência e o tipo de unidade de armazenamento tornam-se pontos críticos.

2.3.1 - Transferência de Dados com Controle da CPU

Este tipo de Transferência exige participação direta da CPU, através do seu conjunto de instruções. Esta tarefa pode ser realizada de duas formas: por "POLLING" ou por INTERRUPTÃO.

A técnica de "polling" consiste em leituras periódicas no dispositivo que gera os dados, com a CPU sempre pronta para fazer transferências.

Nos sistemas de aquisição de dados, quando se usa "polling", a CPU lê constantemente um bit indicativo de geração de dados, presente tanto na conversão A/D como na conversão D/A. Quando este bit dá uma indicação positiva, o dado gerado é transferido imediatamente. Este método é muito utilizado quando a CPU não precisa realizar outras tarefas, pois é necessária a execução de instruções dedicadas à transferência.

Quando a CPU não dispõe de tempo para operação por "polling", usa-se interrupção. Neste método a CPU não necessita vistoriar periodicamente o dispositivo em busca de dados para transferências. Quando um dado está pronto para ser transferido, o dispositivo envia um sinal de interrupção à CPU, que executa um conjunto de instruções necessárias para realização da tarefa.

Uma interrupção pode ser gerada em qualquer instante, independentemente das funções que estejam sendo executadas pelo processador. Isto faz com que esta técnica tenha muita aceitação nos sistemas de aquisição de dados, principalmente nos que trabalham com sinais de baixa frequência.

Os tempos envolvidos numa transferência de dados por controle da CPU são diferentes para as duas técnicas. Para se ter uma idéia da diferença, fez-se uma análise simples [APÊNDICE A] para transferência de um bloco de 64 kbytes gerados a partir de uma conversão A/D de 8 bits, supondo-se que o sistema é baseado num microcomputador do tipo PC/XT. O resultado foi

OPERAÇÃO POR POLLING : 103 Ciclos de Relógio
OPERAÇÃO POR INTERRUPTÃO: 146 Ciclos de Relógio

2.3.2 - Transferência por DMA

As transferências de dados utilizando "polling" ou interrupção podem levar um tempo considerável. A técnica de acesso direto à memória (DMA) permite que a transferência possa ser feita de maneira bem mais rápida, sem participação intermediária da CPU. Normalmente a CPU é envolvida apenas no início da operação, fornecendo as condições em que os dados devem ser transferidos, como número de bytes, endereços da fonte e do destino dos dados e outros parâmetros relacionados. A seguir é apresentada a seqüência de operações comumente utilizadas numa operação de DMA.

- 1 - Um dispositivo de E/S, na necessidade de leitura/escrita na memória do sistema, requisita serviço ao dispositivo de controle de DMA;
- 2 - O controlador, após reconhecer o pedido, solicita à CPU o controle das vias de dados, controle e endereços, através de um sinal "HOLD";
- 3 - Reconhecendo o pedido, a CPU gera um sinal indicando a liberação das vias (sinal "HOLDA");
- 4 - O controlador de DMA, de acordo com a programação inicialmente estabelecida pela CPU, gera os sinais de controle e endereços necessários para a operação de transferência desejada;
- 5 - No final da(s) transferência(s), o controlador desativa o sinal "HOLD" e a CPU reestabelece sua operação, a partir do ponto onde foi reconhecido o pedido de DMA.

As vantagens de se utilizar um sistema de DMA são muitas, destacando-se principalmente:

- A CPU não precisa ser envolvida por longos períodos de tempo, bastando apenas liberar as vias por um determinado período, que depende da quantidade de dados a serem transferidos,
- As taxas de transferência podem ser muito elevadas, superando significativamente as obtidas com controle da CPU.

Mesmo com as importantes vantagens apresentadas, as operações com DMA também impõem limitações nos sistemas que a utilizam. Uma limitação destacável é o fato de a CPU permanecer fora de operação durante todo o processo. Isto impede, por exemplo, que possíveis pedidos de interrupção possam ser atendidos, o que torna necessária uma avaliação mais cuidadosa de DMA em determinados tipos de sistemas. Por exemplo, usar sistema de DMA para fazer transferência contínua de longos blocos de dados pode não ser aconselhável.

Os serviços de DMA são realizados por circuitos complexos desenvolvidos em pastilhas LSI, normalmente denominados de controladores de DMA. Um exemplo de dispositivo que executa este tipo de função é o CI-8237, da INTEL, que é o controlador utilizado nos microcomputadores do tipo PC/XT. O tempo envolvido para transferir os dados nas mesmas condições do item anterior, considerando uma situação crítica com dois canais de DMA em operação, é consideravelmente menor. O tempo avaliado para a transferência [ver APÊNDICE A] foi de, aproximadamente, 22 ciclos de relógio (contra 103 e 146 ciclos necessários para transferência por "polling" e por interrupção, respectivamente). Este resultado permite afirmar-se que a técnica de DMA é fundamental nos sistemas de aquisição de dados para operar com sinais acima de algumas dezenas de kHz.

2.3.3 - Dispositivos de Armazenamentos mais usados nos Sistemas de Aquisição de Dados

Para armazenamento dos dados digitais são usados basicamente unidades de disco (rígido ou flexível) e memórias semicondutoras. Na escolha do dispositivo adequado, dois fatores são particularmente importantes: velocidade e capacidade de armazenamento.

Quando se trabalha com sinais abaixo de 1 kHz, praticamente não existem problemas de velocidade. Estes problemas só aparecem nos sistemas que operam com sinais acima de 3 kHz. Na maioria dos casos a opção é por uma memória do tipo semicondutora.

A principal limitação de um disco rígido está na velocidade de gravação/leitura de dados. Por exemplo, a escrita ou leitura em um disco é feita posicionando-se a cabeça de leitura/escrita numa determinada posição (setor). Neste ponto o dado é lido/escrito muito rapidamente. O problema de velocidade de transferência ocorre quando a cabeça tem que mudar de posição (de um setor para outro, por exemplo), por ser necessário um tempo de espera para estabilização da posição da mesma. Na

maioria dos discos rígidos, este tempo é da ordem de alguns milissegundos, o que limitaria qualquer sistema de aquisição para sinais acima de alguns kHz. Para aproveitar as unidades de discos [12], é necessário um sistema auxiliar, como, por exemplo, o mostrado na figura 2.8.

Este esquema não resolve definitivamente o problema de velocidade de transferência nos discos, mas mostra-se adequado para operação com sinais acima de alguns kHz.

A outra opção, usada na maioria dos sistemas de aquisição, é a memória semicondutora. Este tipo de memória apresenta tempo de acesso muito pequeno e praticamente não oferece obstáculo para transferências rápidas. Entretanto, quando se trabalha com sinais de frequências mais altas (por exemplo, acima de 30 kHz), torna-se necessário o uso de memórias de grande capacidade. Apesar do avanço da microeletrônica, estes dispositivos ainda apresentam custo elevado. Assim, é fundamental que no desenvolvimento de sistemas de aquisição a definição da memória seja feita de modo a viabilizar o compromisso custo x capacidade de armazenamento. Sistemas modernos, como o Analisador de Fourier TEK-2000 [13], já operam com memórias acima de 1 Mbytes.

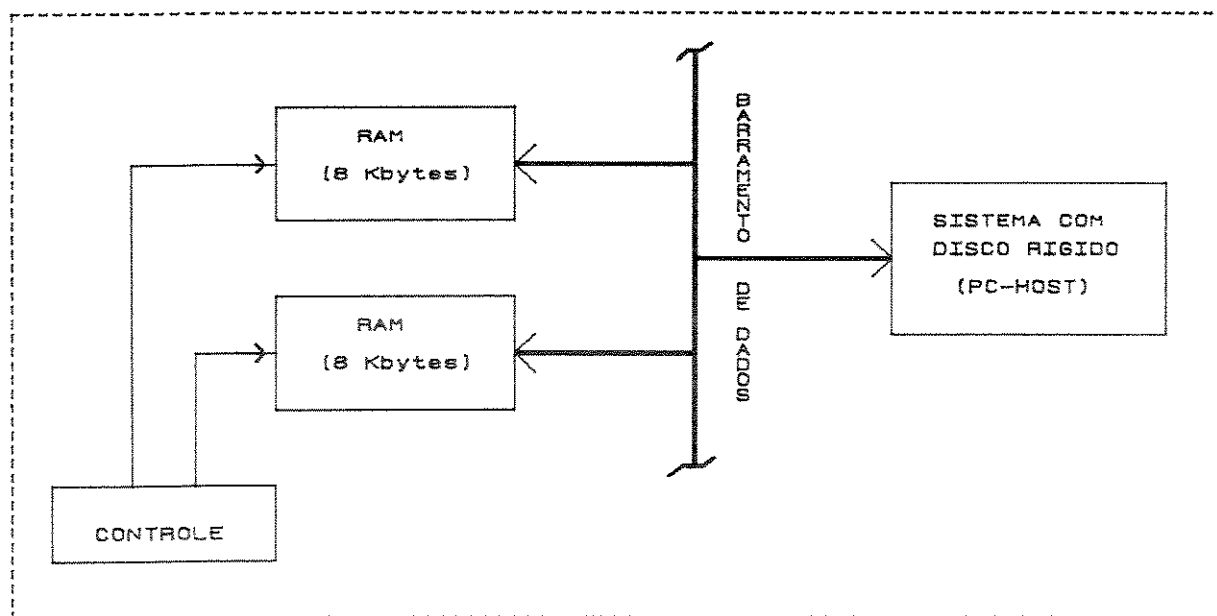


FIGURA 2.8 - Sistema de Transferência para Disco Rígido

2.4 - APLICAÇÕES DE SISTEMAS DE AQUISIÇÃO DE DADOS

O resultado de um processo de aquisição de dados é uma coletânea de valores, que fornecem informações de um determinado sistema. A partir de análises sobre os dados obtidos, pode-se atuar sobre o sistema basicamente de duas formas: em tempo real ou não. Os sistemas que exigem atuação em tempo real são caracterizados por sua capacidade de realizar determinadas tarefas num espaço de tempo suficientemente pequeno e, dependendo da aplicação, podem ter diferentes características de velocidade e precisão. Exemplos são o controle de velocidade de um motor num sistema de controle de processo, e uma filtragem adaptativa num sistema de processamento digital de sinais de voz.

Quando se exige apenas análise de dados, sem a necessidade de nenhum tipo de atuação imediata, são usados os sistemas para operação em tempo não-real. Exemplos de aquisição de dados com este objetivo são a captura e armazenamento de sinais de voz, feitas para que se possa, por exemplo, estudar formas de codificação para reduzir taxas de transmissão, e de sinais biomédicos, para se estudar algum tipo de anomalia.

Considerando a relação entre sistema e aplicação, os sistemas de aquisição de dados podem ser divididos em três grupos: Instrumentos Digitais para captura de formas de ondas e medições em geral, Sistemas para Controle de Processos e Sistemas de Aquisição de Dados de propósito geral.

2.4.1 - Instrumentos Digitais

Atualmente, os instrumentos eletrônicos utilizam as mais diversificadas técnicas digitais na sua construção. Nestes instrumentos, verdadeiras estações de aquisição de dados são incorporadas, de modo a ampliar de maneira significativa a sua capacidade de operação. Assim, instrumentos como voltímetros, osciloscópios, analisadores dinâmicos de sinais, etc. incorporam hoje diferentes sistemas de digitalização de sinais, que os tornam altamente poderosos nas suas aplicações.

Os instrumentos, como voltímetros e amperímetros digitais, possuem não muito mais que a unidade de conversão A/D. Nestes, o sinal geralmente presente é de baixa frequência, o que permite o uso da técnica de conversão A/D de dupla-rampa. Além da confiabilidade e

precisão, pode-se destacar neste caso a facilidade de avaliação de uma dada medida pelo usuário.

Instrumentos mais elaborados, como os osciloscópios digitais, possuem sistemas completos de aquisição de dados incorporados, com complexas unidades de condicionamento de sinal, conversão A/D e D/A, reconstrução do sinal e um sistema de controle geralmente baseado em microprocessador. Estes instrumentos são altamente eficientes, chegando a fazer aquisições de dados a taxas de Giga amostras/s. A função básica das unidades de aquisição é fazer captura de formas de ondas em alta velocidade. Neste caso pouca memória é necessária e o que mais importa, em geral, é a velocidade de conversão A/D. Os conversores A/D mais usados nesses instrumentos usam a técnica de conversão com comparadores paralelos.

Como exemplo, e para se ter uma idéia do potencial de alguns instrumentos, são dados as características mais relevantes de dois sistemas:

A - Osciloscópio Digital Kikosui-7201A [14]

- Taxa de 500 Mamostras/s
- Largura de Faixa de 200 MHz
- Resolução de 8 bits
- Memória de 1 kbytes
- Interface GPIB completa (controle e transferência de dados)

B - Analisador Dinâmico de Sinais HP-35660A [15]

- Aquisição de sinais na faixa de 102,4 kHz
- Operação em tempo real para sinais de até 3,2 kHz
- Memória RAM de 2 Mbytes
- Unidade de disco flexível de 3,5 "
- Interface HP-IB completa (padrão IEEE-488)

2.4.2 - Sistemas para Controle de Processos

O uso de sistemas automáticos tem provocado grandes mudanças nos sistemas de produção industrial, influenciando diretamente na capacidade das indústrias e na qualidade dos produtos. Estas mudanças requerem sistemas de grande confiabilidade e flexibilidade.

A configuração dos sistemas para controle de processos segue geralmente um esquema básico, como o da figura 2.9. Sobre o bloco de experimento ou teste são feitas as aquisições dos dados para análises, que podem resultar em mudanças no processo. Estas mudanças são realizadas pelos blocos restantes, que constituem o sistema de controle de processos, e podem ocorrer dinamicamente.

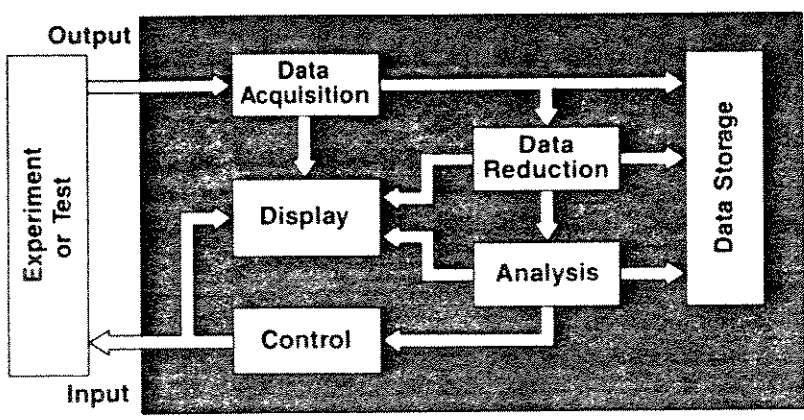


FIGURA 2.9 - Esquema básico de um Sistema para Automação [7]

Sistemas altamente especializados podem hoje ser encontrados, com as mais variadas funções, que possibilitam controle de velocidade, fluxo, deslocamento, temperatura, etc. A grande variedade de aplicações faz surgir sistemas altamente flexíveis e eficientes. É o caso, por exemplo, do MULTIPROGRAMMER SYSTEMS (HP-6944A e HP-6942A) [16], que usa um conjunto de placas para diferentes funções. As ilustrações da figura 2.10 mostram a capacidade deste sistema.

Multiprogrammer Capabilities

Acquisition

Scanner Systems

Analog measurements from up to 96 channels may be acquired at 100,000 readings per second depending upon the scanner system configuration. Digital data may be acquired at 100,000 readings per second. For digital data, the scan rate may be changed as well as the number of channels. For analog data, the scan rate may be changed as well as the number of channels. For digital data, the scan rate may be changed as well as the number of channels. For analog data, the scan rate may be changed as well as the number of channels.

Cards used: Counter, HP 69755A; Timer, Pacer, HP 69736A; Interrupt card, HP 69731A; Memory card, HP 69730A; High-Speed ADC, HP 69754A; Memory card, HP 69730A; HP 69731A.

Offline Analog Acquisition

Signals may be digitized at rates up to 500 kHz by the A/D and stored on the Memory card. Each Memory card can store up to 1,048,576 readings. The digitizing process can take place independent of other Multiprogrammer activities. For Analog-to-Digital Conversion, see the Multiprogrammer manual.

Cards used: High-Speed ADC, HP 69754A; Memory card, HP 69730A.

Offline Digital Acquisition

The Memory card can store up to 1,048,576 bits of digital words at rates up to 1 MB/s. When combined with the Interrupt card and the timer card, data may be stored just prior to, and just after a trigger event.

Cards used: Memory card, HP 69730A; Interrupt card, HP 69731A; Timer/Pacer, HP 69736A.

Measurement

Time Interval Measurement

Elapsed time between two events can be measured in the range of 10 ns to 65,000 days. The Counter card can measure time intervals from 10 ns to 65,000 days. The counter is divided by the chosen time base to determine the interval. For resolution of 0.1 ns, the built-in real-time clock always may be used. The real-time clock also provides time-of-day capability.

Cards used: Counter, HP 69755A; Timer, Pacer, HP 69736A.

Frequency Measurements

The Pulse Counter card accumulates counts over a precise time interval when a timer card is connected to the stable line of the Counter. The program displays the count by the time interval to measure frequencies from 1 MHz to less than 0.1 MHz.

Cards used: Counter, HP 69755A; Timer, Pacer, HP 69736A.

Voltage, Current, and Resistance Measurements

A/D converters may be used to measure voltages from 500 μ V to 100 V in the presence of 250 V of common mode voltage. Connecting a resistor in series with the input permits current measurements for 4.20 mA current loops used in process control. Combine the A/D with the current DAC for resistance measurements.

Card used: High-Speed ADC, HP 69754A; or HP 69750A.

Preset and Up/Down Counting

The Counter may be preset to give values within the count range of 0 to 65,535 and can cause an interrupt when it rolls over. The Counter may be enabled and disabled by the timer card. The computer may read the count without disturbing the counting process.

Card used: Counter, HP 69755A.

Level Detecting

When signals cross preset levels, the Digital Input card can trigger the Interrupt card to interrupt the computer. The alarm trigger levels can be programmed with the D/A on board with resistors.

Cards used: Digital Input, HP 69731A; Interrupt card, HP 69731A.

Event Sensing

A digital word may be used to trigger a computer response with the Interrupt card. The computer responds to the word with a software routine. The interrupt may also cause immediate data response by triggering a pre-loaded output card.

Card used: Interrupt card, HP 69736A.

Control

Digital Output and Switching

Seven bits of data in TTL output buffer can produce 1 to 32 mA pulses at either of two outputs. (W relays, indicators, and solid state AC relays.)

Cards used: Digital Output, HP 69731B; Relay Output, HP 69730A.

Stepping Motor Control

The Stepping Motor Control card produces from 1 to 32 mA pulses at either of two outputs. (W relays, and other digital devices in the form of logic levels for motor drive up to 100 mA of supplementary control systems. The pulse rate (motor speed) is also programmable.)

Card used: Pulse Train, Stepping Motor Control, HP 69735A.

Digital Input

Digital input cards accept 16 bits of data from digital measuring instruments, push buttons, switch boxes, relays, and other digital devices in the form of logic levels for motor drive up to 100 mA of supplementary control systems. The pulse rate (motor speed) is also programmable.

Cards used: Digital Input, HP 69731A; Isolated Digital Input, HP 69731A.

Time and Frequency Reference

Crystal controlled timing pulses, programmable from 1 μ s to 10 hours, may be used as a time base reference for control, measurement, and data acquisition. Format and number of pulses are programmable.

Cards used: Timer, HP 69736A; or Pulse Train, HP 69735A.

Synthesis

Programmable Pulse Generation

Pulses with an amplitude in the range of 1.0 V may be generated with a DAC and Timer. Pacer. Duration of the pulse is programmable from 20 μ s to over 10 hours. Continuous pulse generation is also possible. If a specific number of pulses is required, a Pulse Train card should be used.

Cards used: DAC, HP 69720A; Timer/Pacer, HP 69736A; Pulse Train, HP 69735A.

Programmable DC Voltage and Current

HP 69720A and HP 69720A 200 mA HP 69720A, HP 69720A, HP 69720A 1000 watt Autotransforming DC Power Supplies may be controlled with the Power Supply Control and Output Voltage and Current cards. The output voltage and current can be programmed with the Power Supply Control and Output Voltage and Current cards. The output voltage and current can be programmed with the Power Supply Control and Output Voltage and Current cards. The output voltage and current can be programmed with the Power Supply Control and Output Voltage and Current cards.

Cards used: Power Supply Control, HP 69720A; Resistance Output, HP 69700A; HP 69700A 69700A.

Analog Waveform Synthesis

The Memory card automatically supplies pre-loaded data to the D/A card at rates of up to 125 kHz. Special instructions may be loaded into the Memory card from the computer and used as stimuli for test and processes.

Cards used: Memory card, HP 69730A; or HP 69731A; Voltage DAC, HP 69720A; or Current DAC, HP 69721A.

Offline Digital Stimulus

Up to 1,048,576 bits of digital words may be preloaded on the Memory card and then sent to a device under test at a process rate of up to 600 kHz.

Cards used: Memory card, HP 69730A; or HP 69731A.

Digital-to-Analog Conversion: Voltage

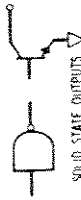
Two-bit voltage DAC's provide outputs for strip chart, x-y, and analog tape recorders as well as control of analog programmable instruments and stimulus of cards under test.

Card used: Voltage DAC, HP 69720A.

Digital-to-Analog Conversion: Current

Two-bit current DAC's provide control of devices requiring a 0-20 mA input as well as stimulus of cards under test and also provide a sensing current for measuring resistance.

Card used: Current DAC, HP 69721A.



80-10 STATE OUTPUTS

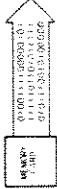
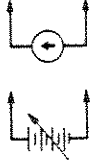
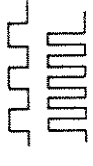
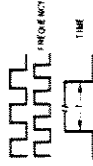
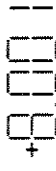


FIGURA 2.10 [16]

Os sistemas aplicados a controle de processos apresentam algumas características peculiares, das quais pode-se citar:

- Grande número de canais de entrada (normalmente diferenciais e com proteção) e de saída;
- Baixa capacidade de armazenamento (geralmente menor que 64 kbytes);
- Operação com taxas de aquisição não muito elevadas (em geral abaixo de 1 kHz);

A área de controle de processos é talvez a que mais oferece aplicações para os sistemas de aquisição de dados.

2.4.3 - Sistemas de Aquisição de Propósito Geral

As muitas aplicações dos sistemas de aquisição de dados terminam exigindo sistemas que possam ser utilizados para os mais variados fins. O objetivo que se quer para este tipo de sistema é a múltipla função, de modo que ele possa atuar como um simples "Data-Logger" e, também, como um completo sistema para controle de processos.

Os sistemas de aquisição de propósito geral são muito usados nas aplicações em que se quer trabalhar com sinais para processamento em tempo não real. Eles se caracterizam por, geralmente, apresentar arquitetura do tipo auto-suficiente e possuir grande capacidade de transferência e armazenamento de dados. A aquisição de sinais de áudio, por exemplo, requer taxas de amostragem acima de 30 kHz, o que exige um sistema com capacidade de armazenamento acima de 256 kbytes (um número mínimo de amostras que possibilita algum tipo de análise).

Um exemplo de sistema com estes objetivos é a estação de análise de sinal da Rockland [9], que pode atuar como gravador de sinais, osciloscópio digital, analisador de espectro, microcomputador, etc. Devido à grande flexibilidade deste equipamento, ele é sugerido para aplicações nas áreas de:

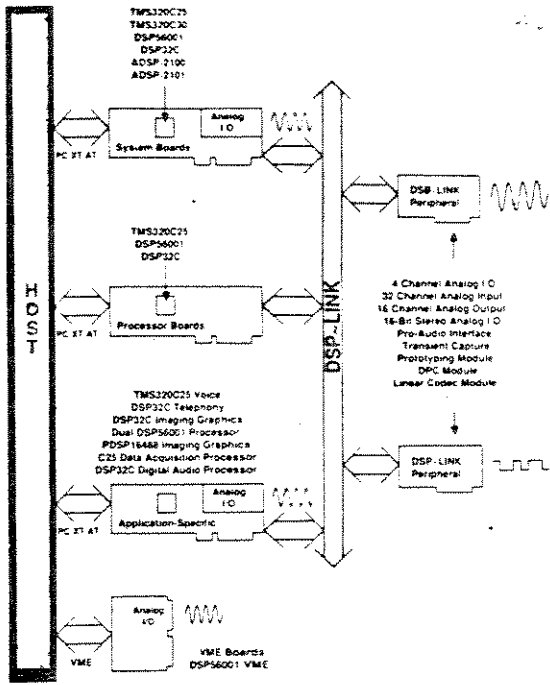
- Eletrônica Geral

- Acústica
- Vibrações
- Servomecanismos
- Análise Estrutural
- Manutenção de Máquinas em Geral

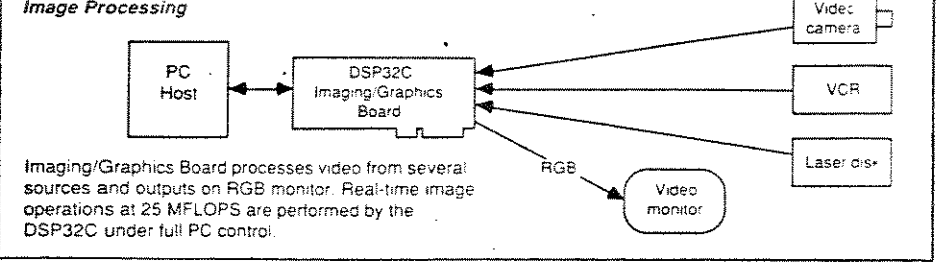
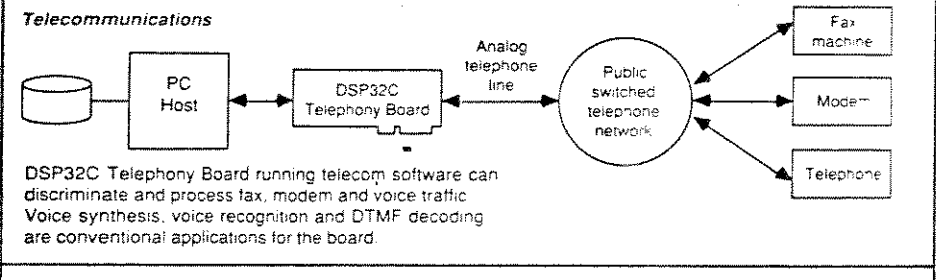
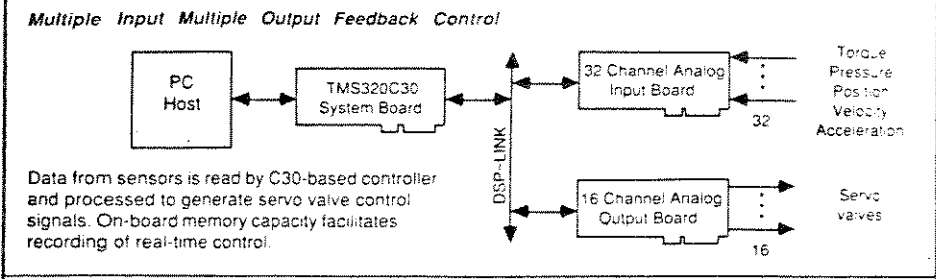
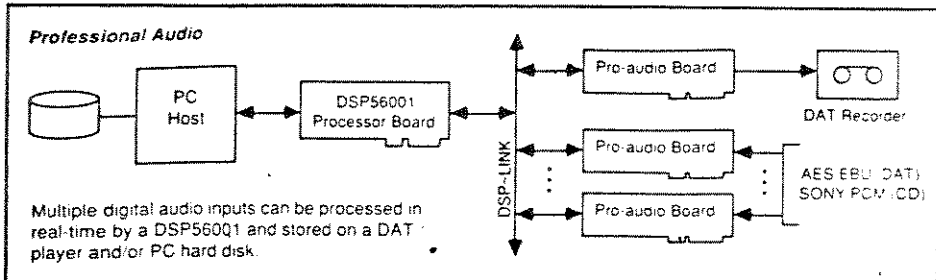
Como características importantes desse sistema pode-se destacar a alta taxa de aquisição (aproximadamente 51 Kamostras/s), a grande capacidade de armazenamento, com 1 Mbytes de RAM e uma unidade de disco rígido de 40 Mbytes, e o sistema de controle e processamento, que é baseado no processador 80286 (para a versão 9040).

Existe também sistemas de propósito geral que aparecem na forma de placas. É o caso, por exemplo, de um conjunto de placas desenvolvida pela SPECTRUM SIGNAL PROCESSING INC. [17], para os mais diversos tipos de aplicações, conforme ilustrações da figura 2.11.

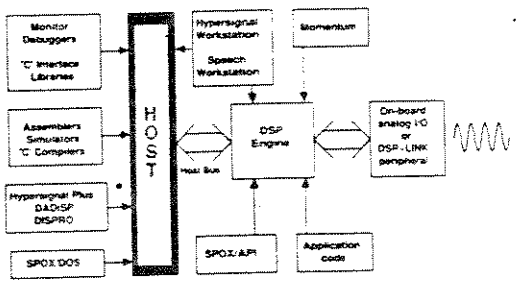
A importância dos sistemas de propósito geral está na capacidade que eles oferecem de adequação aos mais variados tipos de aplicações, mesmo as que não são a priori especificadas.



(a)



(c)



(b)

FIGURA 2.11 - Sistema para Processamento Digital de Sinais (a) Opções de "Hardware" (b) Opções de "Software" e (c) Sugestões para Aplicações [17]

CAPÍTULO 3 - CARACTERÍSTICAS GERAIS E UNIDADES ANALÓGICAS

3.1 - UMA VISÃO GERAL DO SAD

O Sistema de Aquisição de Dados desenvolvido (SAD) se enquadra na categoria dos sistemas auto-suficientes, com uma unidade de controle baseada em microprocessador. Na figura 3.1 tem-se um diagrama de blocos simplificado do SAD.

O sinal analógico a ser adquirido passa inicialmente por uma etapa de amplificação e filtragem, como discutido no capítulo anterior. O processo de amplificação permite adquirir sinais com gama dinâmica de entrada de $\pm 1,0$ V, ± 320 mV, ± 100 mV, ..., $\pm 3,2$ mV, ± 1 mV e ± 320 μ V.

O sinal amplificado é enviado para uma filtragem passa-baixas, podendo-se optar entre quatro faixas de passagem. A primeira, de 0 a 3,4 kHz, reproduz o padrão de qualidade usual dos sistemas telefônicos, onde os interesses maiores são inteligibilidade e economia de faixa. A segunda opção é a faixa de 0 a 8 kHz, que reproduz um padrão médio de qualidade, acima da típica dos sistemas AM de radiodifusão. A terceira faixa, de 0 a 16 kHz, garante um padrão de qualidade um pouco acima dos sistemas FM de radiodifusão. Estas opções de filtragem incorporadas direcionam o sistema para aplicações imediatas em sinais de voz e áudio. No entanto, devido às suas características mais abrangentes, implementou-se também um canal de faixa plana. Esta quarta opção permite ao SAD operar com sinais dentro da faixa máxima especificada (50 kHz) ou usar uma filtragem externa, a critério do usuário.

Da saída dos filtros, o sinal é enviado a um circuito "sample-and-hold", que amostra o sinal analógico, a uma taxa variável entre 0 e 100 kHz. Os valores retidos são enviados ao conversor A/D de 12 bits, cujas saídas digitais são armazenadas temporariamente em "latches", para serem enviados em grupos de 8 bits (um byte) à memória de dados. Existem duas possibilidades de codificação. Na primeira, são aproveitados todos os 12 bits gerados pelo A/D, sendo enviados à memória inicialmente os 8

bits menos significativos da primeira amostra, em seguida os 4 bits mais significativos da segunda amostra junto com os 4 bits mais significativos da primeira amostra, depois os 8 bits menos significativos da segunda amostra, e assim por diante. Na segunda possibilidade, os 12 bits gerados são convertidos em 8 com o auxílio de uma tabela de conversão gravada em EPROM ("table look-up"), de forma a transformar a conversão uniforme original numa conversão não uniforme, segundo a lei A de compressão utilizada na codificação MCP de sinais de voz dos sistemas telefônicos nacionais. Este esquema apresenta o ponto importante de economizar memória, com pequena degradação de desempenho.

Para avaliação do resultado de processamentos ou geração programada de sinais, desenvolveu-se uma unidade de conversão D/A, que possui características compatíveis com a unidade de conversão A/D. A unidade de conversão D/A recebe os dados digitais em blocos de 8 bits (bytes) vindos da memória de dados e os converte à forma analógica. Quando se opera com a codificação das amostras em 12 bits, o primeiro byte recebido da memória representa os 8 bits menos significativos da primeira amostra, o segundo byte contém os 4 bits mais significativos da primeira amostra e os 4 mais significativos da segunda amostra. O terceiro byte corresponde aos 8 bits menos significativos da segunda amostra, e assim por diante. Quando se opera no sistema de compressão, cada byte recebido da memória é traduzido numa palavra de 12 bits, através de uma tabela de conversão, num processo inverso ao utilizado na conversão A/D.

O sinal analógico na saída do D/A sofre uma operação de filtragem passa-baixas para remover as componentes de alta frequência geradas pelo processo de conversão e para compensar o efeito $\text{sen}(x)/x$ gerado pela largura não nula dos pulsos na saída do conversor D/A. Foram previstos três filtros passa-baixas com frequências de corte em 3,4 kHz, 8 kHz e 16 kHz. A compensação do efeito $\text{sen}(x)/x$ é realizada supondo-se que as frequências de amostragem são, respectivamente, 8 kHz, 20 kHz e 40 kHz. Os filtros possuem a mesma topologia dos filtros da unidade de conversão D/A, com os valores ligeiramente modificados para a compensação do efeito $\text{sen}(x)/x$. É também utilizado um canal de faixa plana para usos especiais com filtros externos. O sinal analógico de saída fica condicionado entre os valores ± 5 V.

O controle das operações do SAD é baseado em microprocessador. A unidade de controle é praticamente um microcomputador, possuindo

sistema próprio de entrada e saída de dados, memórias RAM e ROM, interfaces de comunicação, etc.

A operação do SAD pode ser realizada através de um sistema teclado/display ou de um computador externo. A operação por meio do sistema teclado/display permite que o SAD atue de forma independente, facilitando também sua utilização fora de laboratórios. A operação por meio de um computador externo pode ser feita através de dois canais de comunicação. O primeiro é uma interface RS-232, que permite uma comunicação a baixas taxas (até 19,2 kbits/s), o suficiente para uma manipulação remota. O segundo canal usa uma interface GPIB (General Purpose Interface Bus), padrão IEEE-488. Esta interface pode ser usada para transferências rápidas de dados entre o SAD e outros sistemas, possibilitando transmissões a taxas de até 300 kbytes/s.

Dada a taxa relativamente alta em que os dados podem ser gerados pelo conversor A/D (até 150 kbytes/s), a transferência de dados entre os conversores A/D e D/A e memória de dados é realizada por DMA (Direct Memory Access). A capacidade da memória de dados é de 2 Mbytes, que permite a gravação de até 33 segundos de um sinal amostrado a uma taxa de 100 kHz.

Quando as operações de aquisição envolvem sinais de voz ou áudio, pode-se desejar um acompanhamento do processo de geração dos dados digitais e avaliação do resultado do processamento digital destes dados. Para tal, foi incorporado no SAD um amplificador de áudio de potência que alimenta uma caixa acústica ou um fone de ouvido. O amplificador possui controles de tonalidade para compensação de deficiências na resposta de algum elo do sistema de audição, reforço ou atenuação de alguma parte do espectro de áudio, ou simplesmente para satisfazer o gosto auditivo do usuário. Para controle do nível do sinal utiliza-se um VU analógico convencional. São usados também LED's auxiliares como indicadores de saturação do conversor D/A, para monitoração da função utilizada.

Todos os circuitos do SAD são alimentados por uma fonte de saída múltipla que fornece as tensões ± 20 V, ± 15 V, ± 12 V e $+5$ V. Com exceção das tensões de ± 20 V, que são utilizadas exclusivamente pelo estágio de saída do amplificador de potência de áudio, todas as tensões são estabilizadas por reguladores eletrônicos integrados.

3.2 - ESPECIFICAÇÕES

A - Parte Analógica

1. Faixa de Frequências: CC a 50kHz
2. Sensibilidades:
 - a) ± 1 V
 - b) ± 320 mV
 - c) ± 100 mV
 - d) ± 32 mV
 - e) ± 10 mV
 - f) $\pm 3,2$ mV
 - g) ± 1 mV
 - h) ± 320 μ V
3. Filtros: Faixa Plana
 - CC a 16 kHz (Chebyshev, 5ª ordem, $\pm 0,25$ dB)
 - CC a 8 kHz (Chebyshev, 5ª ordem, $\pm 0,25$ dB)
 - CC a 3,4 kHz (Chebyshev, 7ª ordem, $\pm 0,25$ dB)
4. Acoplamento: CC ou CA
5. Frequência de Amostragem: variável de 0 a 100 kHz
padrão: 8 kHz, 20 kHz e 40 kHz
6. Número de bits de conversão:
 - 12 bits: quantização uniforme
 - 8 bits: compensação, lei A
7. Compensação do efeito $\text{sen}(x)/x$ do D/A para frequências de amostragem de: 8 kHz, 20 kHz e 40 kHz
8. Configuração das funções: por um sistema teclado/display ou pelas interfaces GPIB ou RS-232

B - Parte Digital

1. Microprocessador: 8088
2. Co-processador numérico: 8087 (opção futura)
3. Memória ROM: 16 kbytes
4. Memória RAM para programa: 8 kbytes
5. Memória RAM para dados: 2 Mbytes
6. Interfaces de comunicação: GPIB e RS-232
7. Canal de transferência de dados à memória: DMA

C - Interfaces de Comunicação SAD-Usuário

1. Teclado: 16 teclas hexadecimais e 16 teclas de comandos
2. Displays: 2 conjuntos com 8 unidades de 7 segmentos
3. LED's: indicadores de funcionamento das várias unidades, saturação do conversor D/A, etc.
4. VU: indicador do nível do sinal analógico
5. Saída acústica: alto-falante ou fone de ouvido

D - Interfaces de Comunicação SAD-Computador

1. GPIB: paralela com taxa máxima de 300 kbytes/s
transmissão rápida de dados ou comandos
2. RS-232: serial com taxa máxima de 9,6 kbytes/s
transmissão lenta de dados

3.3 - UNIDADES DE AQUISIÇÃO DO SINAL ANALÓGICO

A capacidade de manipulação do sinal analógico, sem distorcer suas características originais, é requisito básico em qualquer sistema de aquisição de dados.

O processamento de sinais analógicos requer cuidados especiais, principalmente em relação ao ruído. No sistema desenvolvido a arquitetura adotada possibilita se trabalhar melhor os estágios analógicos, facilitando o uso de blindagens adequadas e de um sistema de aterramento mais elaborado. Estes cuidados são fundamentais para se garantir um bom desempenho de circuitos analógicos.

Os circuitos analógicos desenvolvidos para o SAD são responsáveis por uma das principais características do sistema: a flexibilidade na aquisição de dados. Em termos gerais o sistema possibilita: aquisição de sinais com níveis de amplitude variada, seleção de quatro diferentes faixas de frequência para aquisição, mudança no tipo de acoplamento entre os estágios analógicos, conversão A/D e D/A linear de 12 bits ou com compressão 8 bits, frequência de amostragem variável, opção de acompanhamento do sinal a ser adquirido através de um medidor de nível de sinal e, quando se tratar de sinais de áudio, de um sistema de reprodução de áudio. Faz parte ainda dos estágios analógicos a fonte de alimentação, que fornece diferentes níveis de tensão.

3.3.1 - Unidade de Controle de Amplitude do Sinal Analógico

Gracias ao uso de um de um amplificador de entrada com ganho programável, o SAD pode digitalizar sinais com níveis variados de amplitude.

O amplificador de entrada possui dois estágios: o primeiro é dotado de um ganho fixo de 40 dB. Este estágio pode ou não estar incorporado ao fluxo do sinal, de acordo com o ganho necessário. O segundo estágio, conectado em cascata com o primeiro, pode apresentar ganho de 0 dB, 10 dB, 20 dB e 30 dB, também de acordo com o ganho geral necessário. Assim, o amplificador de entrada pode ter seu ganho programado entre 0 dB e 70 dB, com passo variável de 10 dB.

Na figura 3.2 é apresentado um diagrama de blocos do sistema de controle de amplitude. Através do indicador de nível de sinal (VU), é possível verificar se o sinal analógico a ser digitalizado deve ou não ser amplificado. A partir desta informação, é feita a configuração do ganho por meio de sinais de controle adequados, provenientes da unidade de controle das funções analógicas do sistema (capítulo 4, item 4.4.1). Pode-se também definir o tipo de acoplamento do amplificador de entrada, necessário principalmente quando se tem interesse nas frequências muito baixas, ou quando o sinal de entrada apresenta nível CC.

A célula básica dos amplificadores desenvolvidos é apresentada na figura 3.3. O circuito é baseado no CI ICL-7650 (INTERSIL), amplificador que possui internamente um controle de "off-set" estabilizado a "chopper", com valor típico de 0,7 μ V, e também um controle de saturação. Estas características garantem um desempenho altamente satisfatório no sistema de controle de amplitude, além de possibilitar um circuito bastante compacto.

3.3.2 - Circuito para Seleção de Faixa do Sinal Analógico

Para evitar o efeito da distorção por "aliasing", o SAD limita a faixa de frequência do sinal de interesse através de uma filtragem passa-baixas. Existe também a opção de um canal de faixa plana, que pode ser usado para incorporação de outros filtros ou simplesmente para aquisição do sinal sem filtragem.

A redução da faixa do sinal que se quer digitalizar é feita por um conjunto de três filtros passa-baixas. Na figura 3.4 é apresentado um diagrama de blocos do conjunto de filtros. O sinal é filtrado simultaneamente por todos os filtros e a saída desejada é selecionada através de chaves formadas por relés. A manipulação das chaves é feita pelo circuito de controle das funções analógicas. É possível também introduzir um corte nas baixas frequências pela seleção de acoplamento AC na saída dos filtros.

O sinal na entrada dos filtros, proveniente do estágio amplificador pode excursionar entre -1 V e $+1\text{ V}$. Para garantir um sinal de saída variando entre -5 V e $+5\text{ V}$, que é a faixa de excursão do conversor A/D utilizado, fez-se o projeto dos filtros, e também do canal de faixa plana, com um ganho de 14 dB .

As faixas de operação do sistema de filtragem são $3,4\text{ kHz}$, 8 kHz , 16 kHz , além do canal de faixa plana.

No projeto do filtro de $3,4\text{ kHz}$ utilizou-se um modelo Chebyshev de ordem 7, com uma ondulação de $0,25\text{ dB}$ e função de transferência normalizada dada pela equação:

2,2516	1,5208	0,6093	0,3076
-----	-----	-----	-----
x	x	x	x
-----	-----	-----	-----
$s^2+0,1368s+1,0451$	$s^2+0,3836s+0,7059$	$s^2+0,5542s+0,2828$	$s+0,3036$
ESTÁGIO 1	ESTÁGIO 2	ESTÁGIO 3	ESTÁGIO RC

A implementação foi feita utilizando-se três estágios ativos de ordem 2 e um estágio RC de ordem 1. Cada estágio de ordem 2 apresenta um ganho de 20/3 dB e o conjunto formado pelo estágio RC e acoplamento tem um ganho de -6 dB.

Para os filtros de 8 kHz e 16 kHz foi utilizado também um modelo de Chebyshev, com uma ondulação de 0,25 dB e função de transferência normalizada dada por:

$$H(s) = \frac{3,4663}{s^2 + 0,27s + 1,0955} \times \frac{1,6966}{s^2 + 0,707s + 0,5365} \times \frac{0,4369}{s + 0,4369}$$

ESTÁGIO 1

ESTÁGIO 2

ESTÁGIO RC

Os estágios 1 e 2 possuem ganho de 10 dB e o conjunto formado pelo estágio RC e acoplamento CA/CC tem ganho de -6 dB. A implementação da equação anterior é análoga à dos filtros de 8 kHz e 16 kHz, diferenciando-se apenas na desnormalização das respectivas frequências.

Todos os estágios de ordem 2 foram implementados com o circuito genérico da figura 3.5, que apresenta uma função de transferência dada por [18]:

$$A_V = \frac{G_E \cdot G_E \cdot (G_A + G_E)}{s^2 \cdot C_1 \cdot C_3 \cdot G_E + s \cdot C_3 \cdot G_1 \cdot G_E + G_E \cdot G_A \cdot G_E}$$

A partir da função de transferência de cada célula base, foram determinados os valores dos componentes de cada estágio dos filtros. Antes da implementação final fez-se um grande número de simulações e ajustes, de forma a se obter os valores dos componentes compatíveis com os disponíveis comercialmente. Nas figuras 3.6 (a,b,c) são apresentadas as curvas finais dos filtros.

O canal de faixa plana foi implementado com base num amplificador faixa larga, com corte em frequência acima de 1,8 MHz.

3.3.3 - Circuito "Sample-Hold" e Conversão A/D

A - Circuito "Sample-hold"

O processo de conversão A/D no SAD é precedido por um circuito "sample-hold", necessário devido às características do conversor A/D utilizado (ver capítulo 2, item 2.1.2-A).

Na figura 3.7 é apresentado um esquema simplificado do circuito desenvolvido. O sinal a ser amostrado passa pelo seguidor A, que deve ter baixa impedância de saída e alta disponibilidade de corrente, requisitos fundamentais para se garantir precisão no processo de retenção das amostras. Da saída do seguidor o sinal é transferido pela chave MOS (quando fechada, posição 1) para o capacitor C2, que se carrega com a tensão do sinal de entrada. O sinal é seguido constantemente e a amostra é retida no instante em que é disparado o sinal de amostragem (SH), que atua sobre a chave MOS. A possibilidade da tensão sobre o capacitor acompanhar o sinal depende da impedância de saída do circuito seguidor A, da resistência série da chave, do valor do capacitor e da frequência do sinal. Também deve-se tomar cuidado com o fenômeno da absorção dielétrica do capacitor, que para certos tipos de materiais, como a cerâmica, pode ser significativa. Para minimizar os transitórios de carga e descarga de C2, utiliza-se o capacitor C1, que participa de um processo de transferência de carga através da chave MOS.

O valor do capacitor C2 utilizado no circuito deve ser o menor possível, compatível com a impedância de entrada do circuito seguidor B.

B - Circuito de Conversão Analógico-Digital

O circuito de conversão analógico-digital do SAD utiliza o conversor A/D rápido de 12 bits, AD-7672, da Analog Devices, que funciona pelo processo de aproximação sucessiva. Na versão disponível para o projeto, o conversor AD-7672 realiza uma conversão em $10 \mu\text{s}$, suficiente para o processamento de sinais com espectro de frequência na faixa de 0 a 50 kHz. A resolução de 12 bits garante um ruído de quantização baixo.

Na figura 3.8 tem-se o circuito completo para conversão A/D. O sinal analógico bipolar, com valores entre $\pm 5 \text{ V}$, é aplicado em AIN1. O início do ciclo de conversão é dado pela transição do sinal CSTART/ ao nível "0" (/ indica sinal ativo quando em nível baixo e será usado no decorrer deste texto). O sinal CSTART/ é aplicado aos terminais CS/ e RD/, após sofrer um atraso de aproximadamente 450 ns em relação ao sinal de comando da retenção do circuito "sample-hold".

Como o conversor é de 12 bits, enquanto a memória de dados, microprocessador, controlador de DMA, etc. são orientados por bytes, é necessária alguma providência para manipulação eficiente dos 12 bits gerados no processo de conversão A/D. Uma possibilidade, que tem o mérito da simplicidade, é a conversão dos 12 bits em 2 bytes pelo acréscimo de 4 zeros, mas o preço que se paga numa implementação deste tipo é a redução na capacidade da memória por um fator de 3/4. No sistema desenvolvido foram utilizados dois esquemas alternativos.

No primeiro esquema, os 12 bits de saída disponíveis após a conversão são armazenados em "latches" (figura 3.8). Enquanto os 8 bits menos significativos são armazenados em U6, os 4 mais significativos ficam retidos em U7 e U8. Após a primeira conversão, os 8 bits menos significativos são enviados para o barramento de saída, iniciando em seguida uma nova conversão. Terminada esta segunda conversão, os 4 bits mais significativos da primeira conversão, retidos em U7 e os 4 bits mais significativos da segunda conversão, retidos em U8, são enviados

como um byte ao barramento de saída. Finalmente são enviados o 8 bits menos significativos da segunda conversão. Definindo:

$DB_1(i)$ = bit de ordem i ($i=0,1,\dots,11$) da 1ª conversão

$DB_2(i)$ = bit de ordem i ($i=0,1,\dots,11$) da 2ª conversão

chega-se à seguinte estrutura dos dados $D7, D6, \dots, D1, D0$ enviados à memória de dados:

D7	D6	D5	D4	D3	D2	D1	D0
.....
.....
.....
$DB_1(7)$	$DB_1(6)$	$DB_1(5)$	$DB_1(4)$	$DB_1(3)$	$DB_1(2)$	$DB_1(1)$	$DB_1(0)$
$DB_1(11)$	$DB_1(10)$	$DB_1(9)$	$DB_1(8)$	$DB_2(11)$	$DB_2(10)$	$DB_2(9)$	$DB_2(8)$
$DB_2(7)$	$DB_2(6)$	$DB_2(5)$	$DB_2(4)$	$DB_2(3)$	$DB_2(2)$	$DB_2(1)$	$DB_2(0)$
.....
.....

Neste 1º esquema, os dados $D7, D6, \dots, D1, D0$ são enviados ao barramento de saída com uma taxa igual a $3/2 F_a$, onde F_a é a taxa de conversão analógico-digital.

No 2º esquema, os 12 bits são convertidos em um byte (8 bits) através de uma tabela ("table look-up") implementada numa EPROM (figura 3.8-U9). Desta maneira, transforma-se o processo de conversão uniforme de 12 bits numa conversão não uniforme com 8 bits, segundo a lei A de compansão (compressão-expansão) utilizada nos sistemas de Modulação por Códigos de Pulsos da rede telefônica nacional. Esta técnica mantém relativamente constante a relação sinal/ruído de quantização para amostras de valores numa gama bastante grande, introduzindo uma certa penalização em relação à codificação linear para sinais fortes. Como para estes últimos o ruído de quantização não é muito significativo, especialmente para sinais de voz, dadas as características de mascaramento do ouvido humano, chega-se à conclusão que este esquema apresenta o grande atrativo da economia de memória com desempenho bastante razoável. Como ponto negativo, além da degradação da relação sinal/ruído deve ser citada a necessidade de um processamento adicional dos dados antes de tentar qualquer processamento digital com característica linear, como filtragem, por exemplo.

3.3.4 - Conversão D/A e Reconstrução do Sinal Analógico

A - Circuito de Conversão D/A

O circuito de conversão digital-analógico do SAD é formado por um conjunto de "latches", duas EPROM's e o conversor D/A propriamente dito, que é o AD-667 (Analog Devices). Os "latches" são usados para permitirem a montagem das palavras de 12 bits para o AD-667, pois as palavras fornecidas pela memória são de apenas 8 bits. Esta opção é utilizada quando a conversão é linear, ou seja, os 12 bits são aproveitados integralmente. Para o caso em que se trabalha com compansão de 8 bits, os 12 bits fornecidos ao conversor D/A são gerados a partir das EPROM'S, que fazem a decodificação dos 8 bits (que foram codificados no circuito de conversão A/D). A figura 3.9 apresenta o circuito detalhado da conversão D/A.

Para formação das palavras de 12 bits na conversão D/A são necessárias três leituras de memória, de modo a se obter duas palavras

de 12 bits. Na primeira leitura, os 8 bits, que correspondem aos bits menos significativos da primeira amostra, são armazenados em U5 na transição positiva do relógio CLK1. Na leitura seguinte, os 8 bits, cujos 4 bits mais significativos correspondem aos 4 bits mais significativos da primeira palavra de 12 bits e os 4 bits menos significativos correspondem aos 4 bits mais significativos da segunda palavra de 12 bits, são armazenados simultaneamente nas unidades U6 e U7, nas transições positivas dos sinais CLK2 e CLK3, respectivamente. Deste modo tem-se nos "latches" U5 e U6 a primeira palavra de 12 bits completa que, ao sinal simultâneo de OC1/ e OC2/, é liberada para a entrada do AD-667, que tem a conversão ativada com o sinal CS. No período seguinte a unidade U5 armazena os 8 bits menos significativos da segunda palavra de 12 bits, na transição de CLK1. Assim, nos "latches" U5 e U7, tem-se a segunda palavra de 12 bits completa, que é liberada para o AD-667 com a ativação simultânea dos sinais OC1/ e OC3/. A partir deste ponto o processo passa a se repetir.

Quando os dados provenientes da conversão A/D de 12 bits estiverem codificados em 8 bits, é necessário um circuito que faça a operação inversa. Este circuito foi implementado utilizando-se duas Eprom's 2764 (U8 e U9, figura 3.9), que têm a função de expandir a palavra de 8 bits (D0 a D7) para uma palavra de 12 bits (DB0 a DB11). Os dados de 8 bits provenientes da memória de dados atuam como endereços simultaneamente nas duas memórias EPROM's. Da primeira EPROM saem os 8 bits menos significativos e da segunda os 4 bits mais significativos.

Na configuração adotada no SAD, o AD 667 usa os 12 bits (DB0 a DB11) provenientes do conjunto de "latches" (U5, U6 e U7) para gerar o sinal analógico. Estes bits ficam armazenados no "latch" interno do conversor, que é controlado pelo sinal A3 (pino 12). Somente quando A3 está em nível baixo, o valor digital armazenado é convertido para analógico e assim permanece até receber um novo dado digital.

B - Reconstrução do Sinal Analógico

O sinal de saída do conversor D/A apresenta uma queda de resposta em frequência de acordo com uma função "sampling", do tipo $\text{sen}(\text{PI.F}/\text{Fa})/(\text{PI.F}/\text{Fa})$, onde Fa é a frequência com que o sinal foi inicialmente amostrado para conversão A/D. Para anular este efeito, foram elaborados filtros de reconstrução que fazem simultaneamente a

compensação $\frac{\sin(\pi F/F_a)}{(\pi F/F_a)}$ para as frequências de amostragem de 8 kHz, 20 kHz e 40 kHz. Estes filtros são topologicamente parecidos com os filtros de 3,4, 8 e 16 kHz, respectivamente, utilizados para seleção da faixa do sinal (item 3.3.2). Como opção existe também um canal passado, que não atribui nenhuma compensação.

Os filtros estão estruturados da mesma forma que os filtros usados para seleção de uma faixa de frequência para aquisição (figura 3.4). A resposta em frequência dos filtros de reconstrução são apresentadas na figura 3.10 (A, B e C). As curvas de número 1 representam as respostas em frequência dos filtros passa-baixas. As curvas de número 2 representam as respostas dos filtros multiplicada pela função $\frac{\sin(\pi F/F_a)}{(\pi F/F_a)}$ nas frequências de amostragem indicadas.

3.3.5 - Reprodução de Sinais de Áudio e Circuito de Alimentação

A - Sistema de Reprodução de Sinais de Áudio

Para realização de testes subjetivos de processamentos de sinais de áudio e voz, foi desenvolvido um amplificador de áudio com potência suficiente para alimentar uma caixa acústica ou um fone de ouvido. Na figura 3.11 tem-se um diagrama simplificado do sistema de reprodução de sinais de áudio, onde também está incluído um circuito de monitoração do nível do sinal, empregando um indicador de painel convencional do tipo VU ("Volume Units").

O sinal aplicado ao amplificador de áudio e ao circuito do VU sofre inicialmente um processo de seleção através de SW1 e SW2. As fontes de entrada, com exceção de ENTR, têm a amplitude normalizada entre os níveis ± 1 V.

SW1 é uma chave manual de 5 posições com as seguintes possibilidades de seleção:

- 1) ENTR/M - Sinal analógico de entrada do SAD, sem qualquer processamento (amplificação, filtragem ou remoção do nível CC)
- 2) AMPL/M - Sinal na saída do amplificador de entrada.

- 3) AD/M - Sinal na saída dos filtros e entrada do conversor analógico-digital.
- 4) DA/M - Sinal na saída do filtros associados ao conversor digital-analógico.
- 5) AUTO - Transferência da seleção à chave MOS SW2.

As quatro primeiras posições permitem ao operador a seleção manual da fonte de sinal a ser monitorada, facilitando a tarefa de estabelecimento dos níveis de amplificação, detecção de irregularidades, etc. Na quinta posição, a seleção é delegada à chave MOS SW2, que funciona comandada por "software", de acordo com a operação que estiver sendo executada no momento. Por exemplo, durante uma gravação, será selecionada a entrada AD; numa reprodução, a entrada DA, etc. A seleção automática é feita a partir do circuito de controle das funções analógicas.

O sinal selecionado é enviado ao circuito indicador de nível de sinal (VU) e ao controle de volume (A2) conectados na entrada do seguidor A1. O sinal proveniente do controle de volume, de baixa impedância, é enviado ao circuito de controle de tom (A3). Da saída do controle de tom, o sinal segue para o amplificador de potência. O sinal de saída pode ser enviado a uma caixa acústica com impedância de 8 ohms ou a um fone de ouvido.

B - Fonte de Alimentação

Os vários circuitos utilizados no SAD possuem alimentações de diferentes valores, o que exige uma fonte de tensão de níveis variados.

A fonte do SAD, apresentada na figura 3.12, segue um esquema convencional. O transformador utilizado fornece as tensões de 8 V e 15+15 V (valores eficazes), que são utilizadas para a geração das tensões contínuas de +5 V, ± 20 V, ± 15 V e ± 12 V. As tensões de ± 20 V não sofrem regulação e são utilizadas exclusivamente para alimentação do estágio de saída de áudio, que drena alta intensidade de corrente, com grandes variações, mas que por outro lado é pouco sensível às flutuações da alimentação. As demais tensões são reguladas por circuitos reguladores integrados, que permitem bom desempenho com economia de espaço.

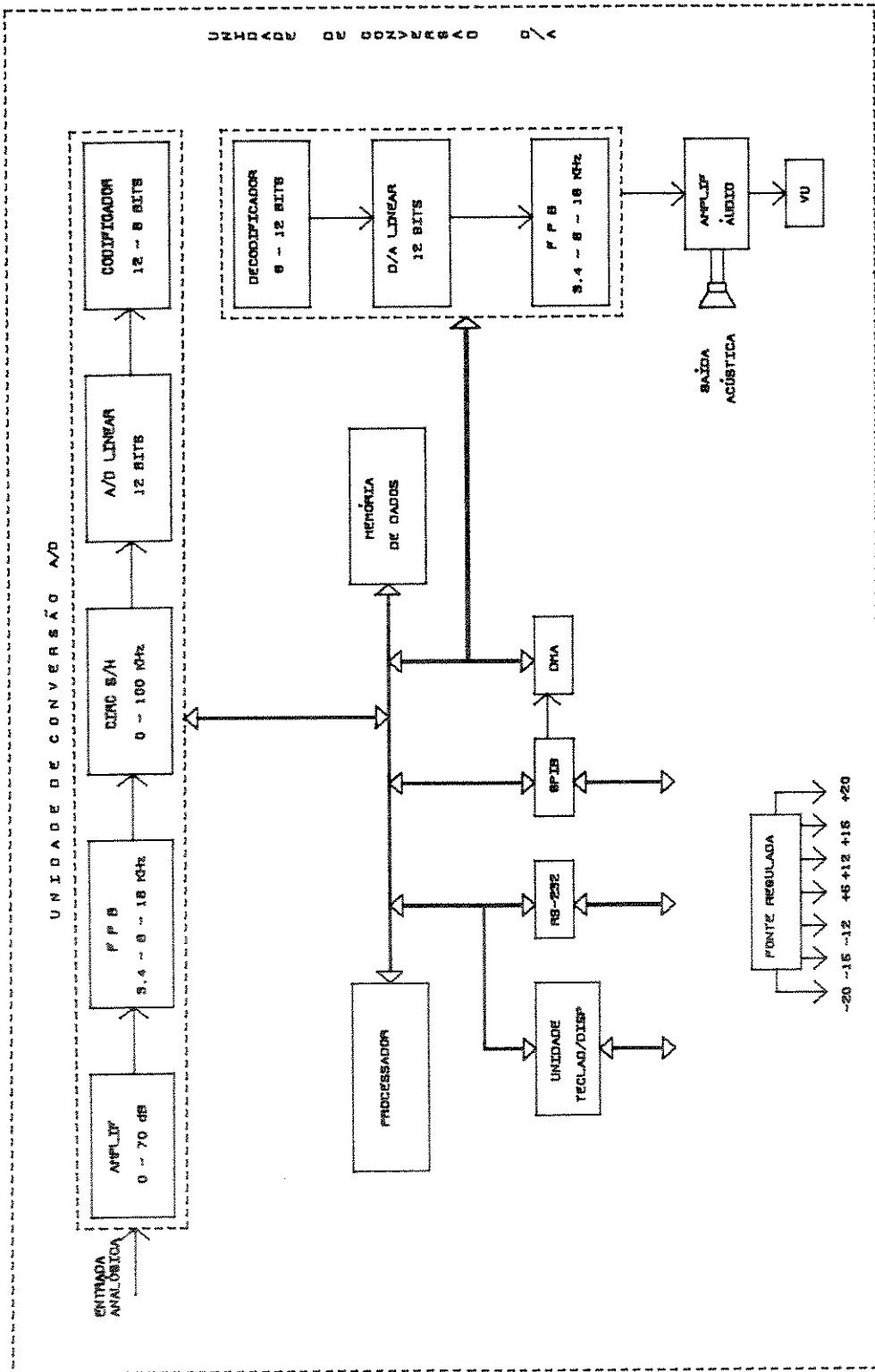


FIGURA 3.1 - DIAGRAMA DE BLOCOS DO SISTEMA DESENVOLVIDO (SAD)

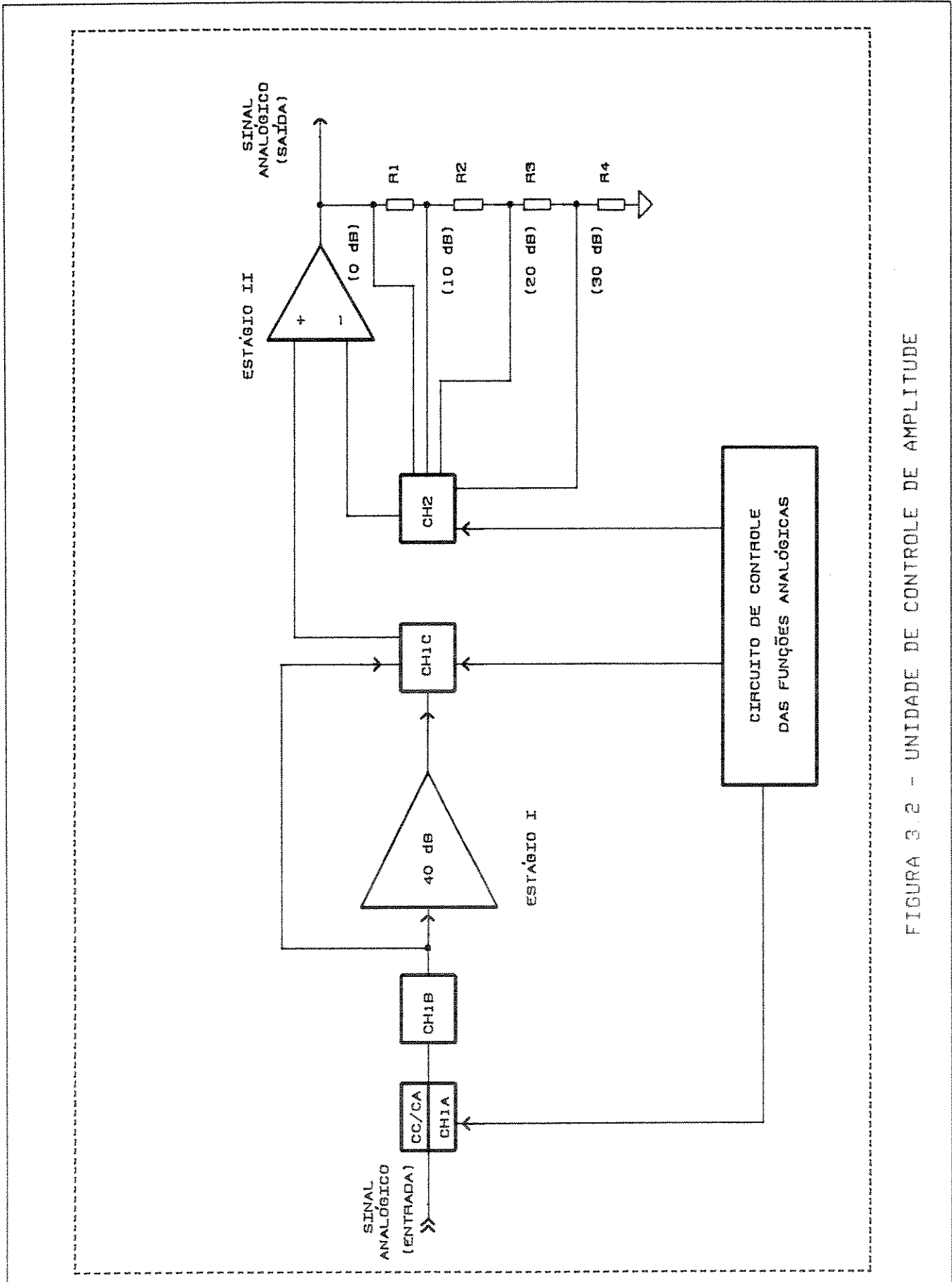


FIGURA 3.2 - UNIDADE DE CONTROLE DE AMPLITUDE

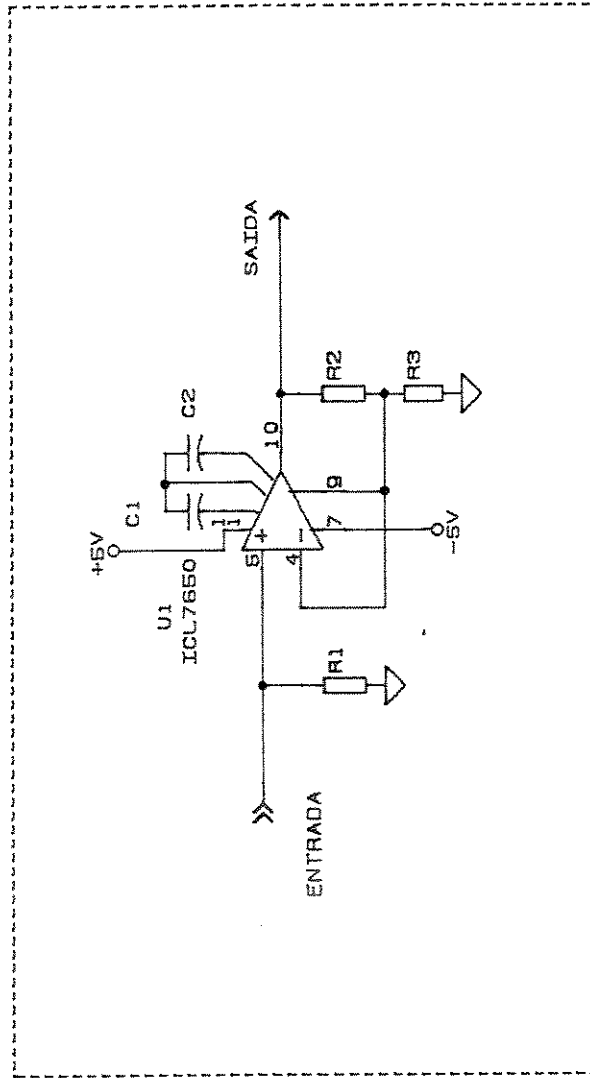


FIGURA 3 3 - CÉLULA AMPLIFICADORA BÁSICA

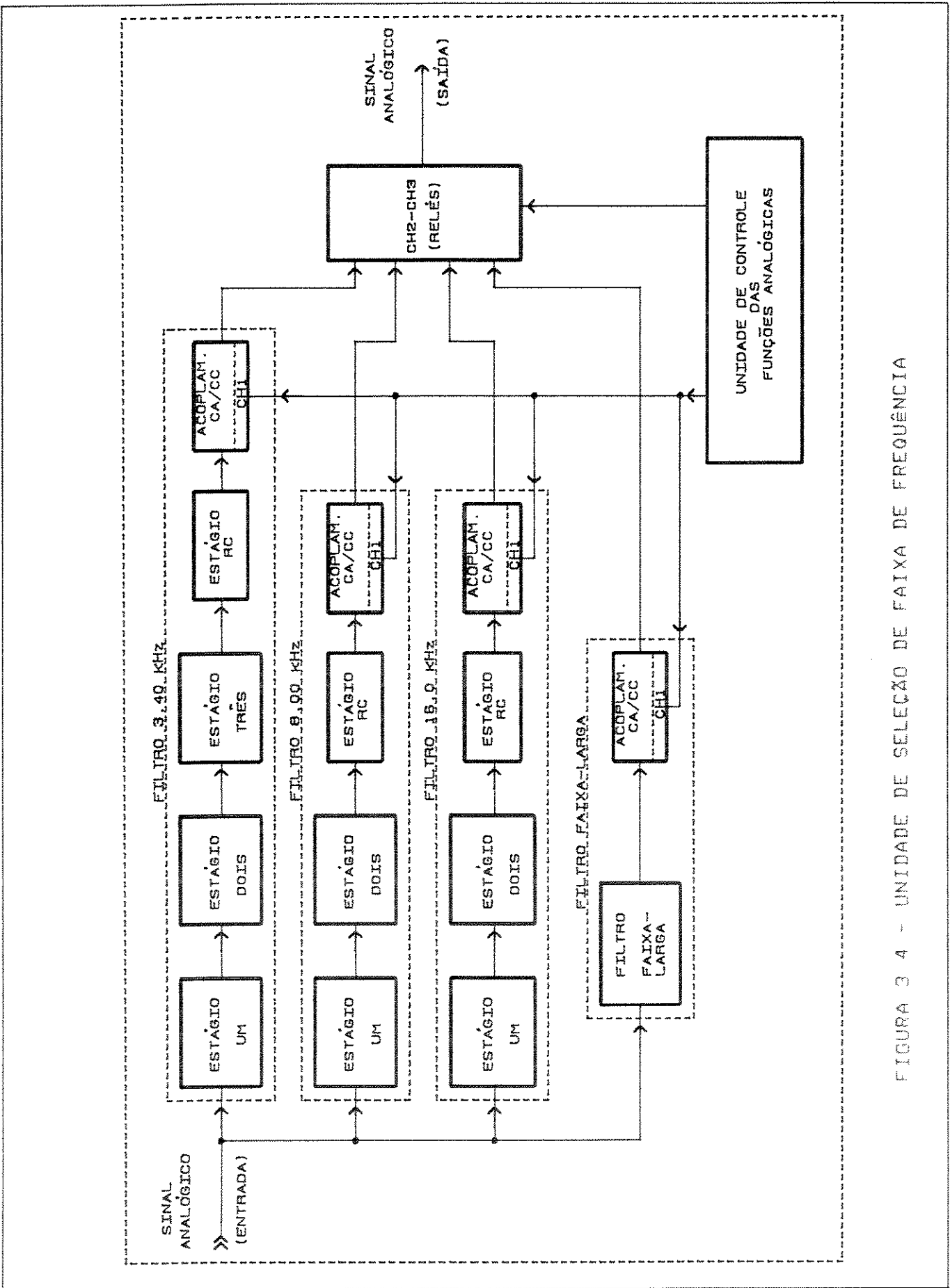


FIGURA 3 4 - UNIDADE DE SELEÇÃO DE FAIXA DE FREQUÊNCIA

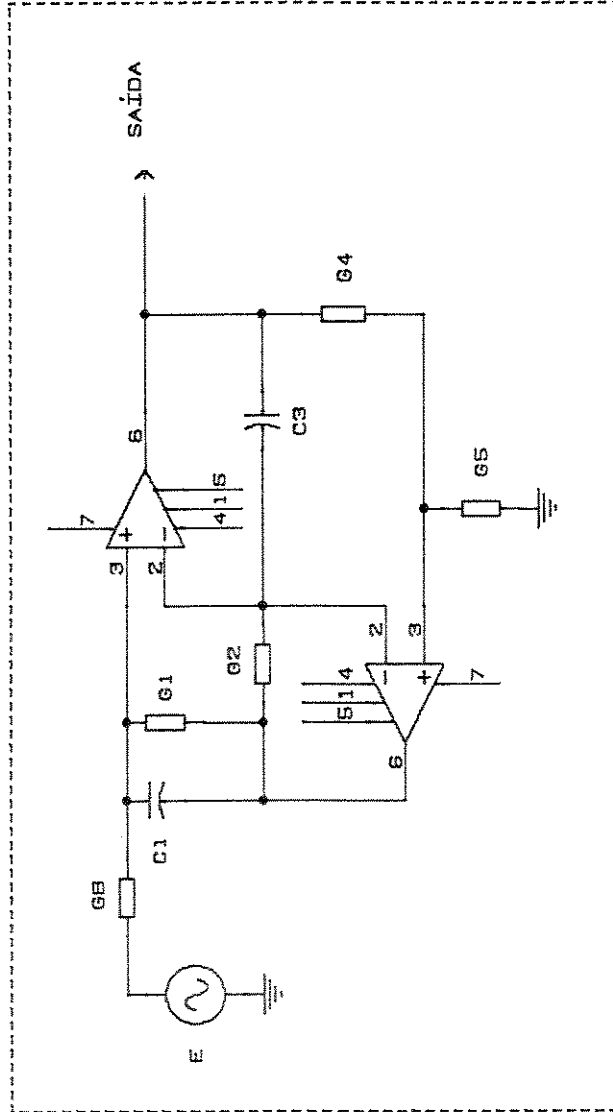
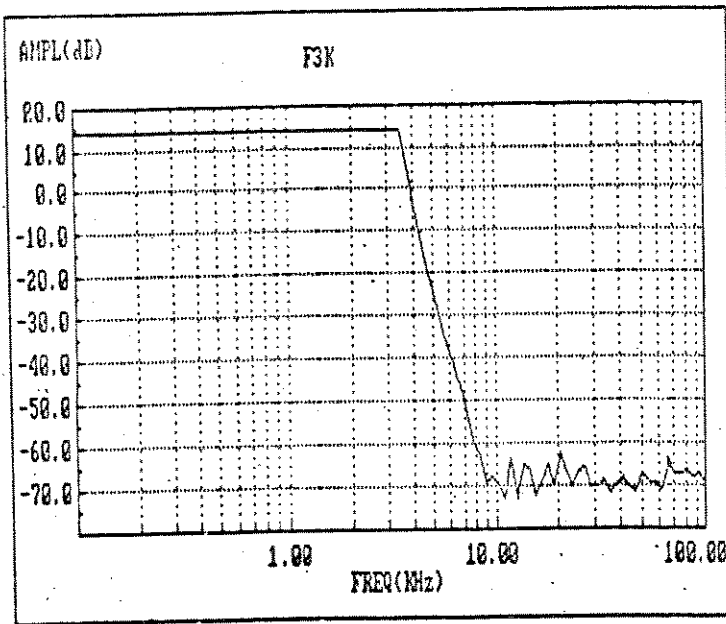
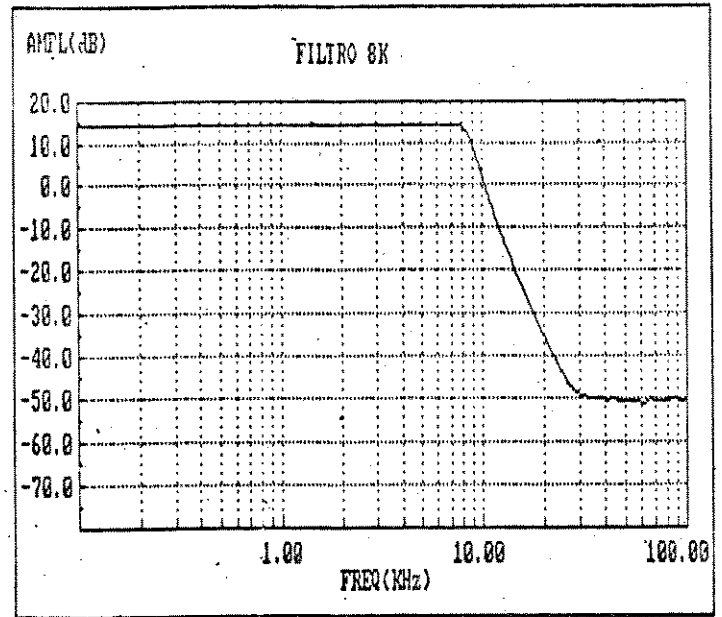


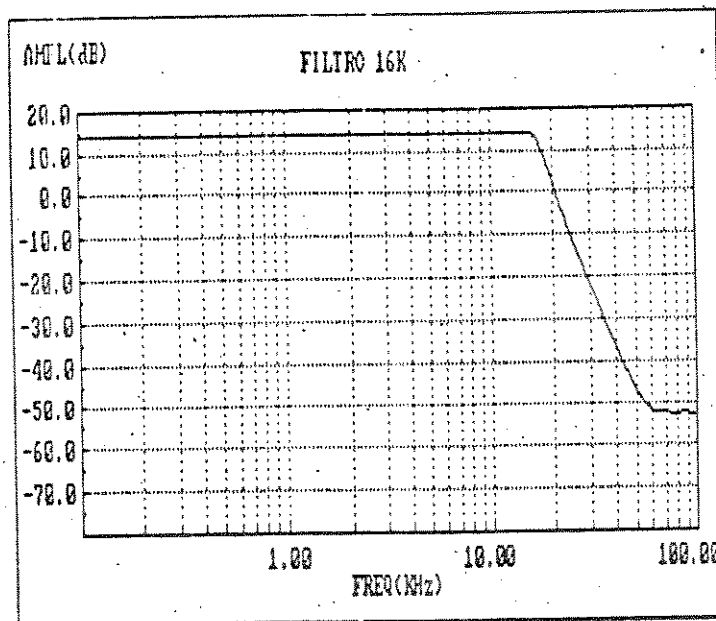
FIGURA 3 5 - CÉLULA ATIVA BÁSICA DOS FILTROS



(A)



(B)



(C)

FIGURA 3 6 - (A) RESPOSTA EM FREQUÊNCIA DO FILTRO DE 3,4 kHz
 (B) RESPOSTA EM FREQUÊNCIA DO FILTRO DE 8 kHz
 (C) RESPOSTA EM FREQUÊNCIA DO FILTRO DE 16 kHz

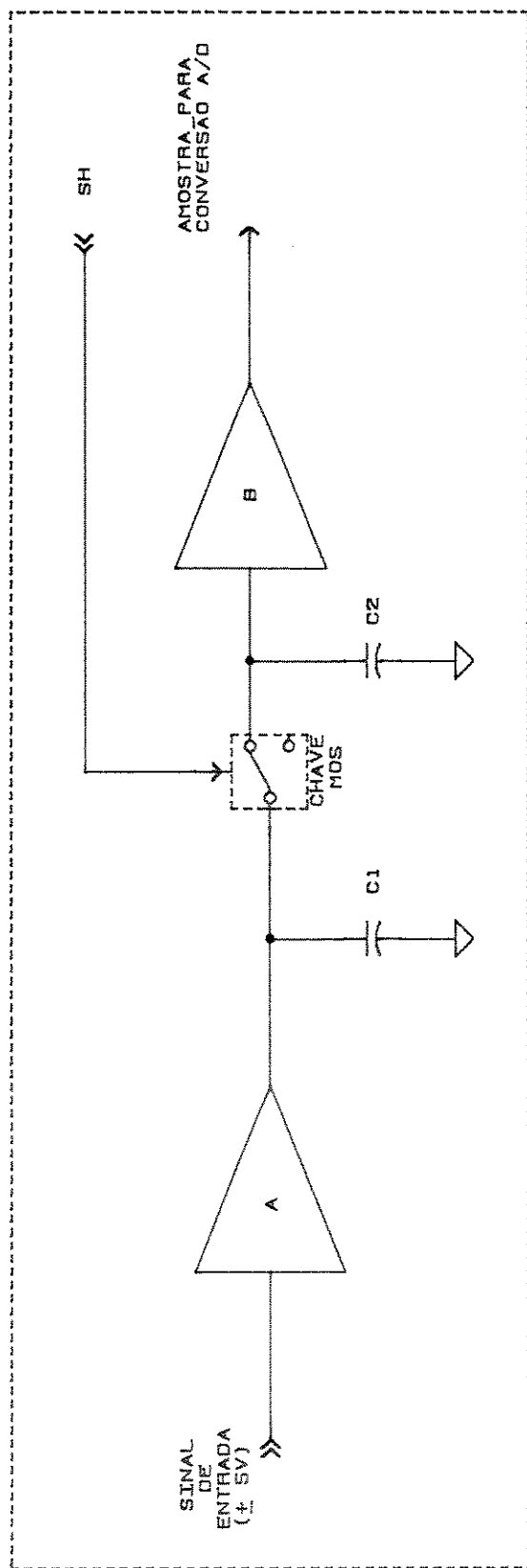


FIGURA 3 7 - DIAGRAMA SIMPLIFICADO DO CIRCUITO "SAMPLE-HOLD"

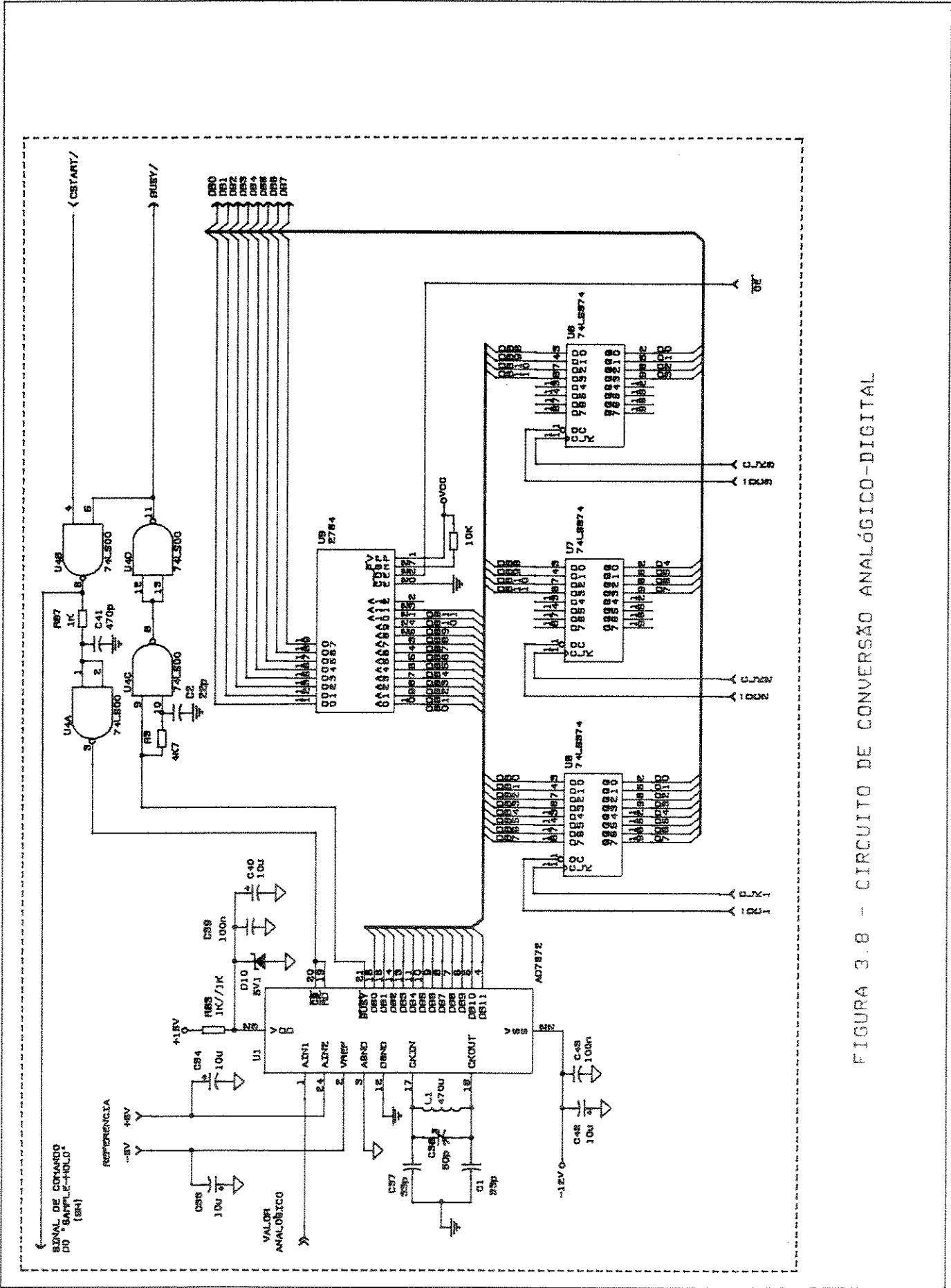


FIGURA 3.8 - CIRCUITO DE CONVERSÃO ANALÓGICO-DIGITAL

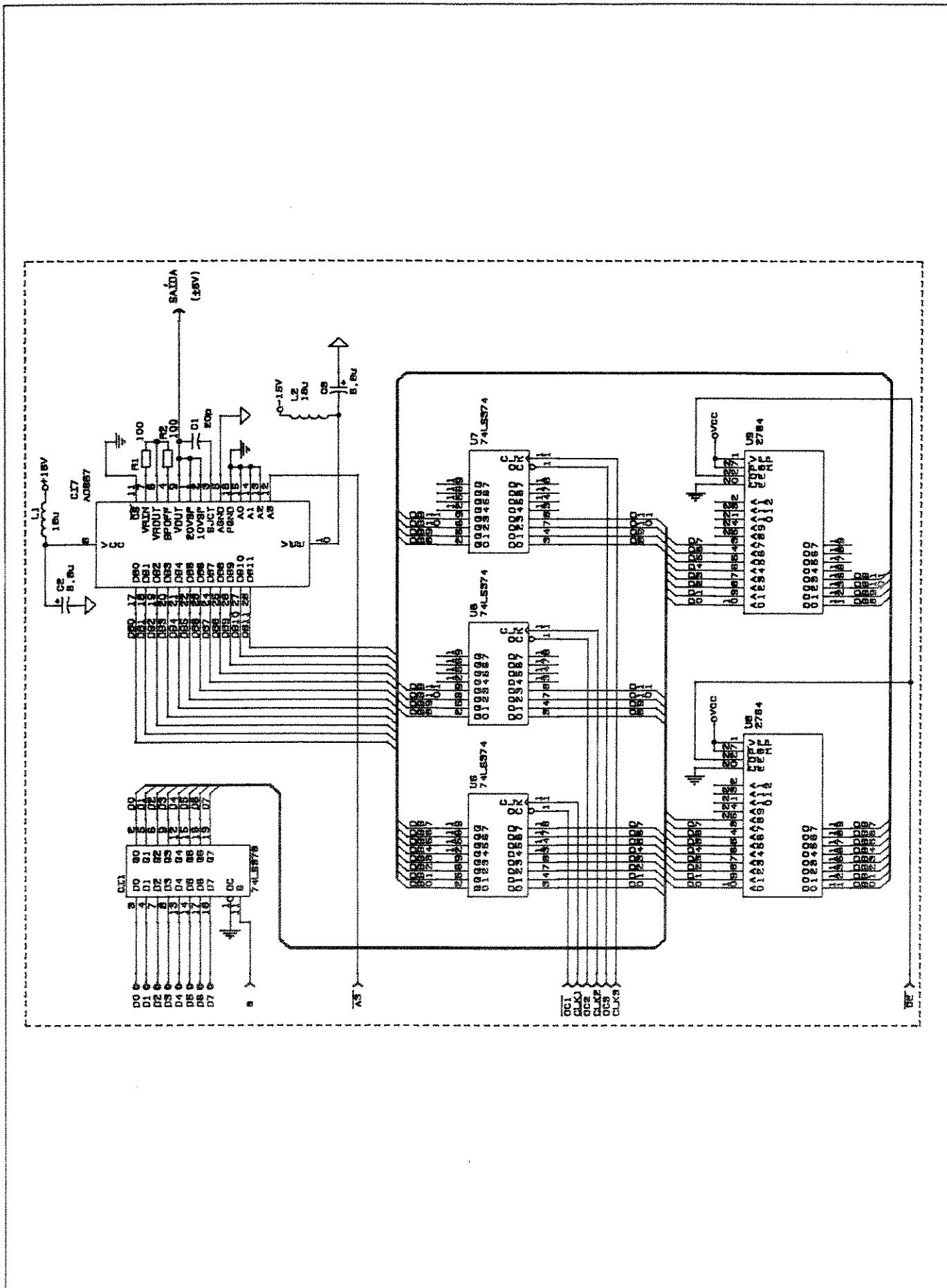


FIGURA 3.9 - CIRCUITO DE CONVERSÃO DIGITAL-ANALÓGICO

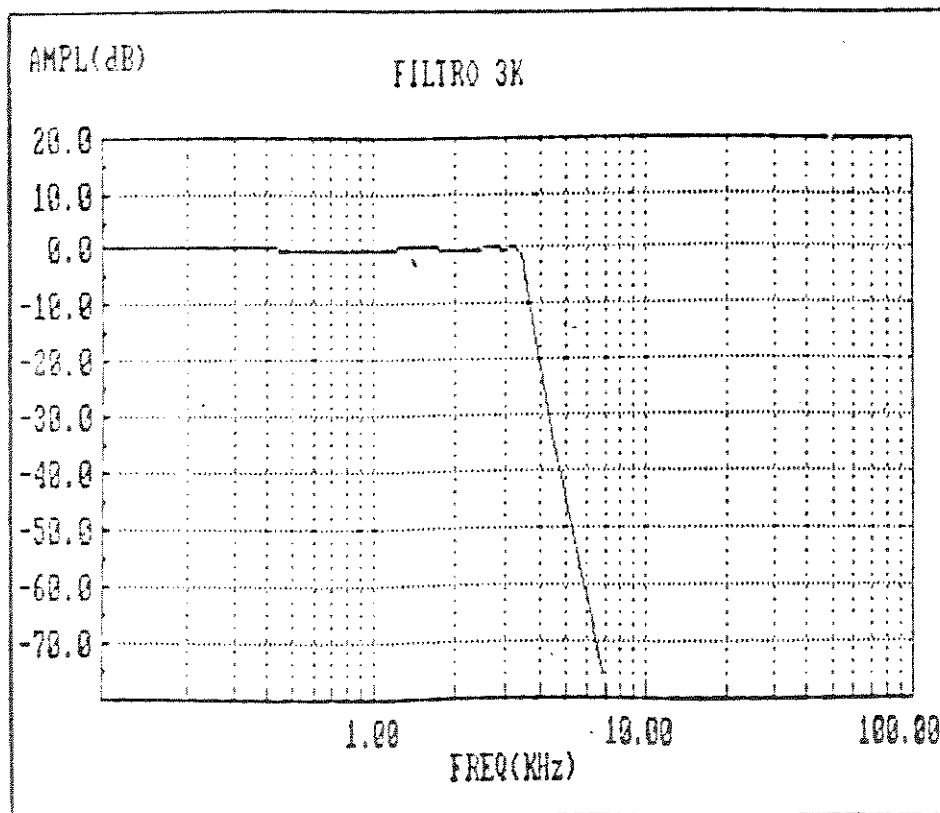
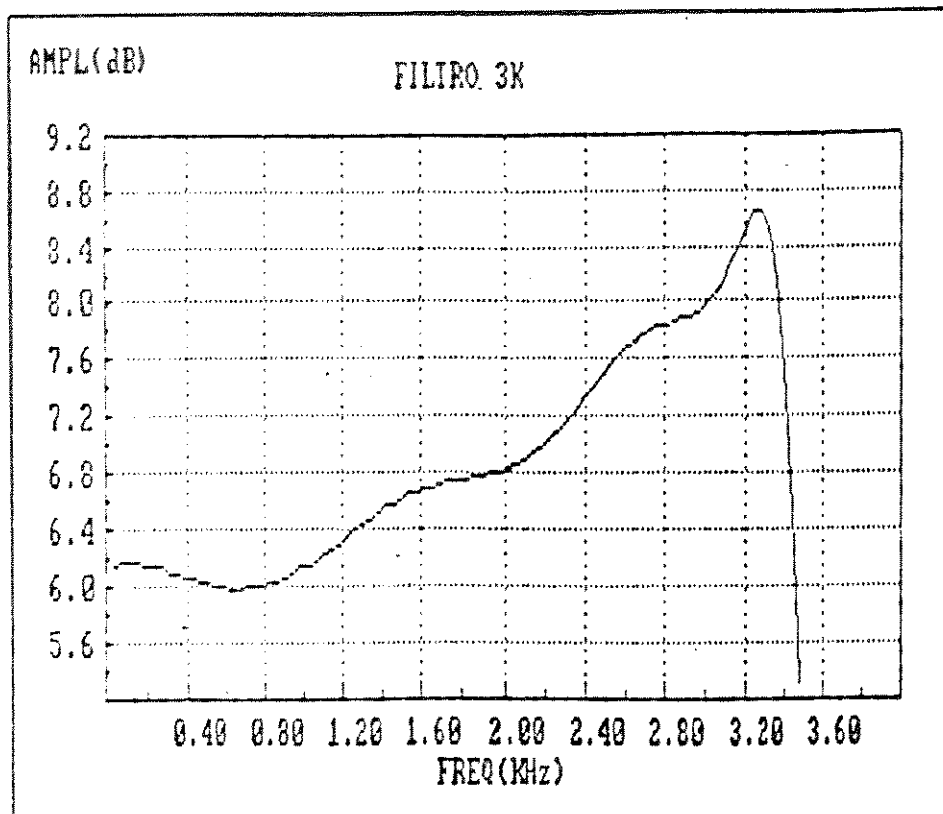


FIGURA 3 10 - (A) RESPOSTA EM FREQUÊNCIA DO FILTRO DE RECONSTRUÇÃO PARA AMOSTRAGEM EM 8 kHz

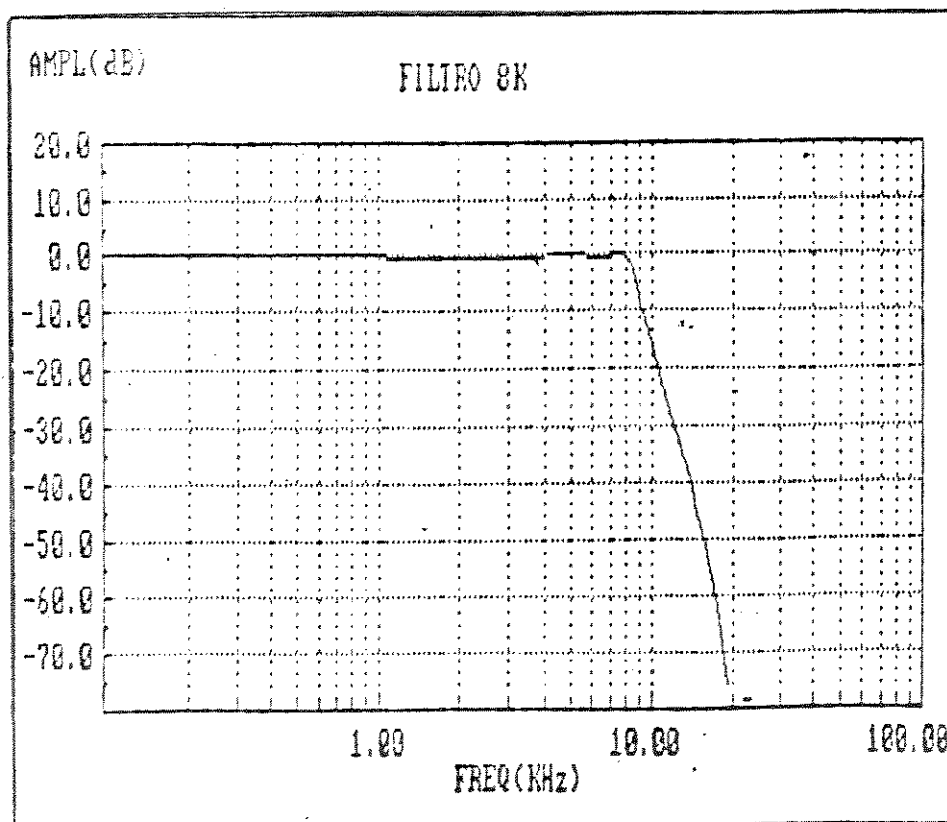
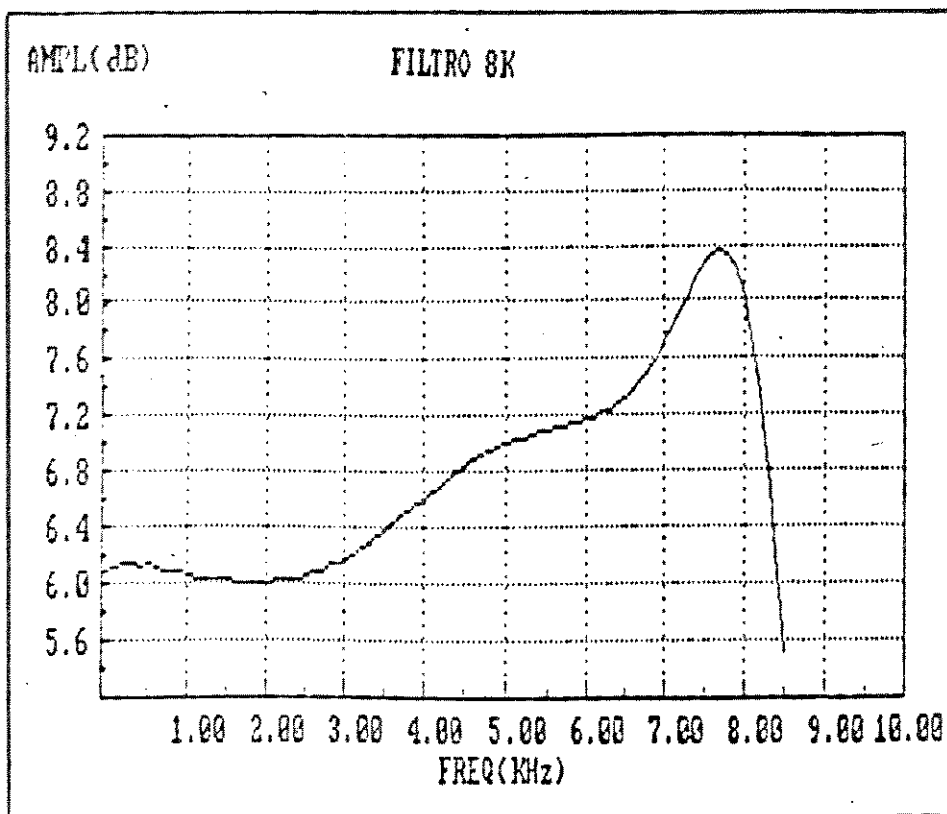


FIGURA 3 10 - (B) RESPOSTA EM FREQUÊNCIA DO FILTRO DE RECONSTRUÇÃO PARA AMOSTRAGEM EM 20 kHz

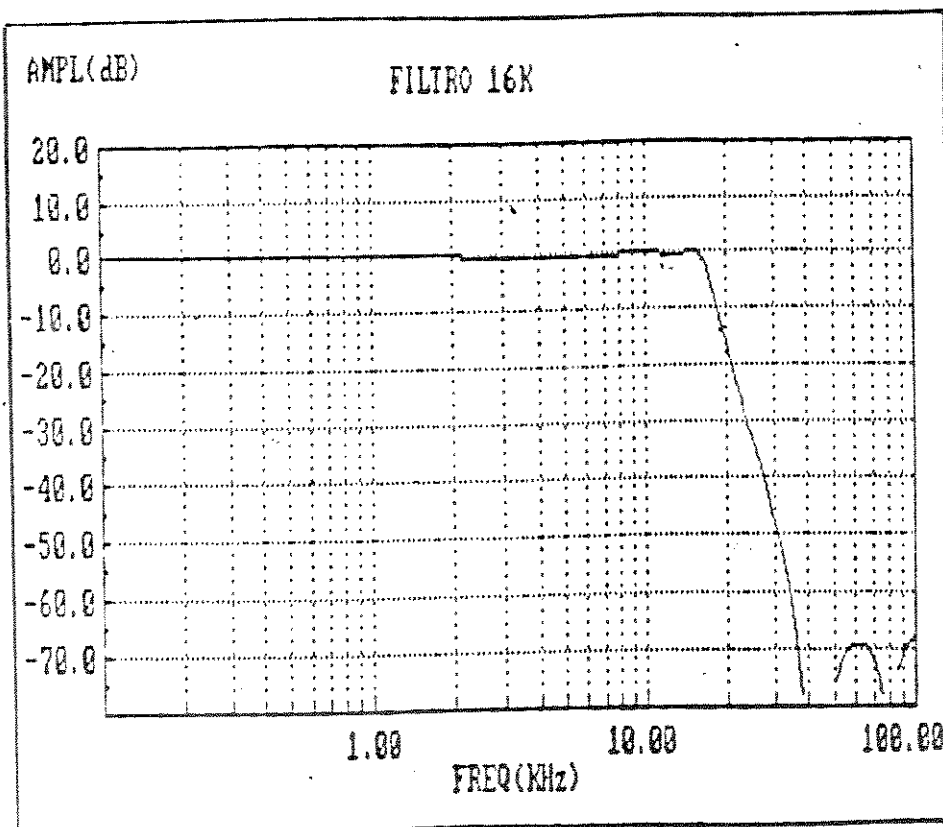
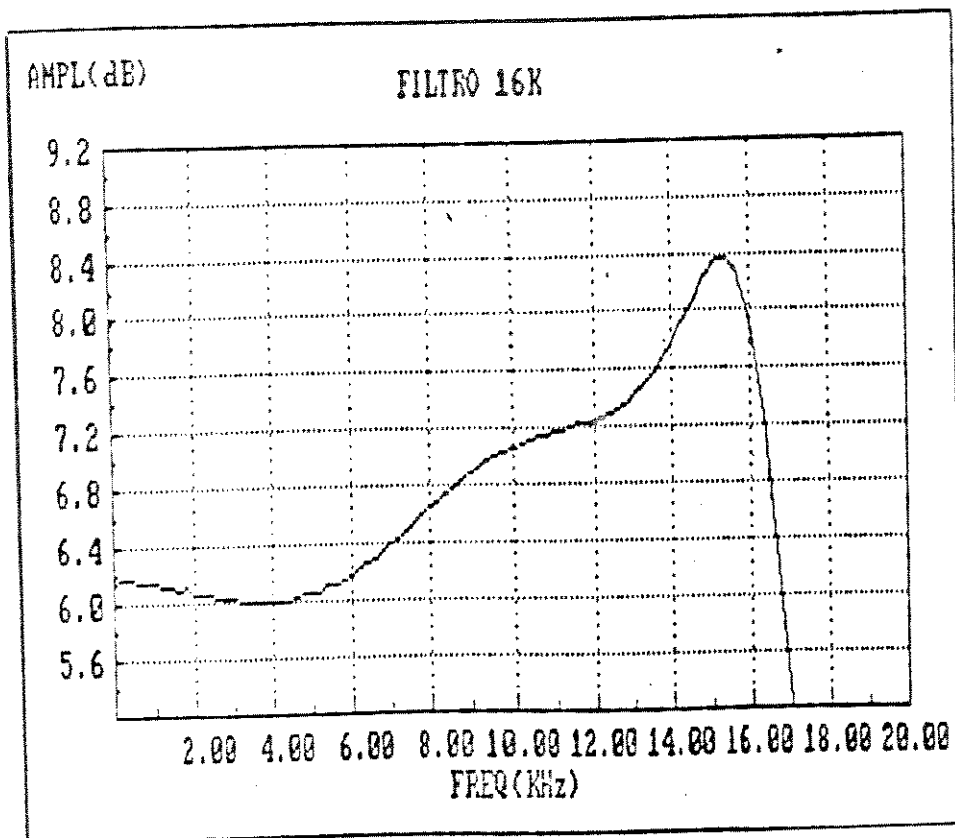


FIGURA 3.10 - (C) RESPOSTA EM FREQUÊNCIA DO FILTRO DE RECONSTRUÇÃO PARA AMOSTRAGEM EM 40 kHz

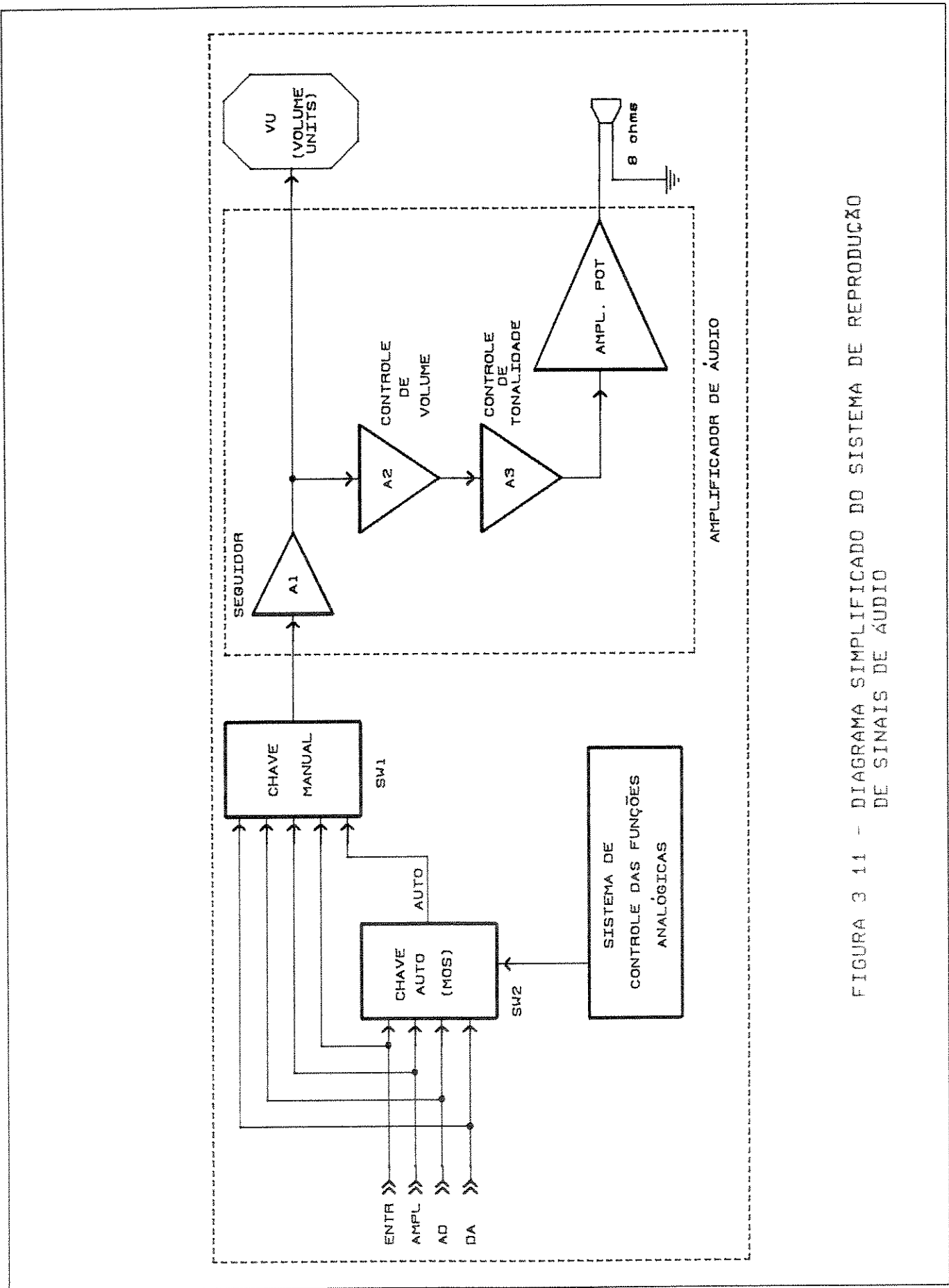


FIGURA 3 11 - DIAGRAMA SIMPLIFICADO DO SISTEMA DE REPRODUÇÃO DE SINAIS DE ÁUDIO

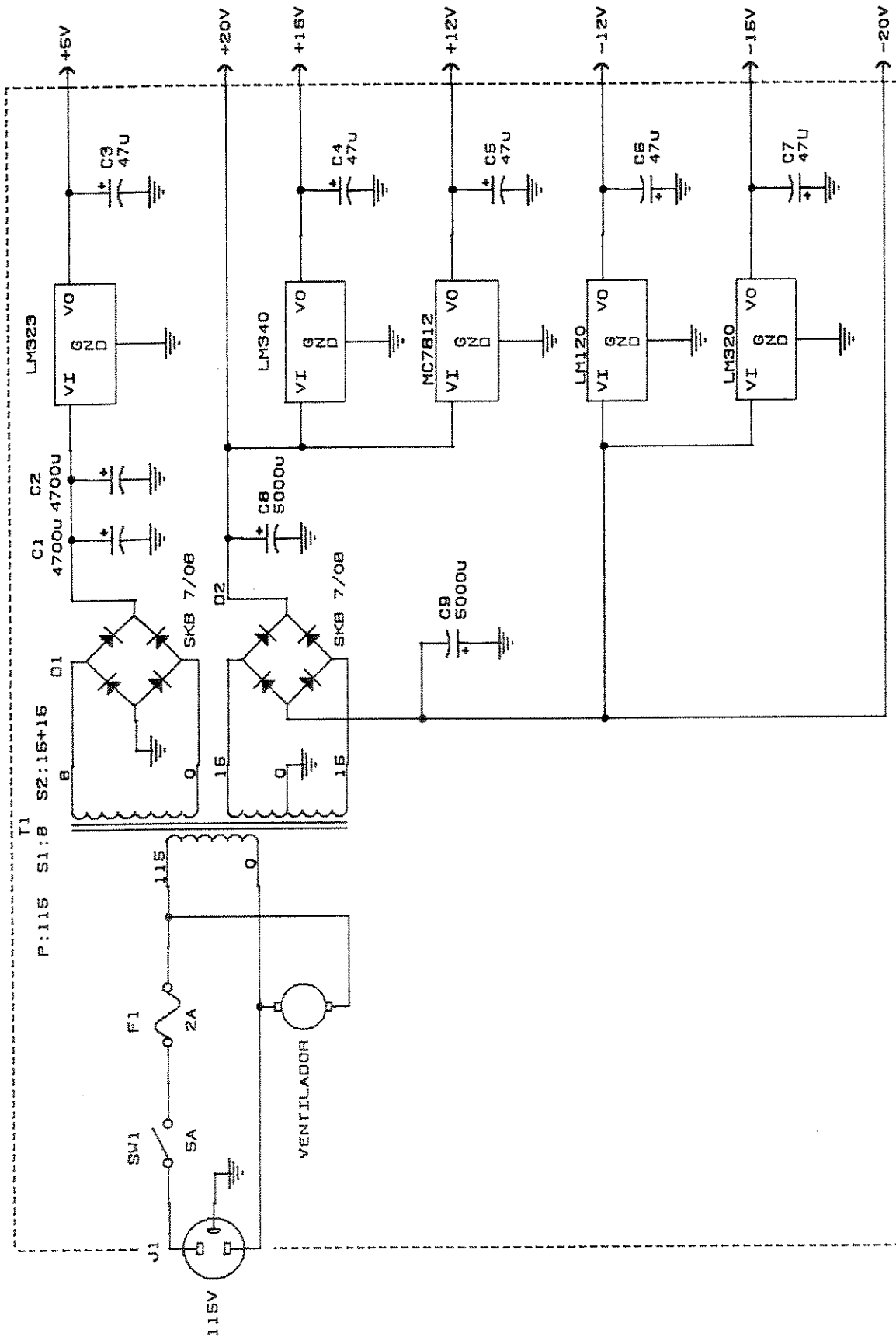


FIGURA 3 12 - FONTE DE ALIMENTAÇÃO DO SAD

CAPÍTULO 4 - UNIDADES DE CONTROLE E MANIPULAÇÃO DE DADOS

No tipo de arquitetura adotado para o SAD, optou-se por um sistema de controle com microprocessador. A escolha do processador adequado foi baseada nas características definidas previamente, destacando-se principalmente a grande capacidade de endereçamento e facilidade de interligação com outros sistemas. Dentro de uma avaliação que envolveu custo e facilidade de obtenção de componentes, poderiam ser utilizados processadores de 8 ou 16 bits.

Os microprocessadores de 8 bits, como por exemplo o Z-80 e o 8085, possuem conjuntos de instruções suficientes para atender às necessidades do SAD. No entanto, as limitações do "hardware", principalmente em relação a capacidade de endereçamento (no modo direto, apenas 64 kbytes de memória), os tornam desaconselháveis. Por outro lado, a baixa velocidade de processamento (Por exemplo, Z-80 e 8080 operam com 2 MHz, 8080A e 8085A com 3 MHz.) poderia limitar expansões futuras no sistema.

Dentro da família dos microprocessadores de 16 bits existe uma variedade de possibilidades e a opção final foi pelo microprocessador 8088. Fatores importantes na escolha deste microprocessador foram as facilidades de aquisição de componentes periféricos e de desenvolvimento de "software". O microprocessador 8088 atende à todos os requisitos de "hardware" e "software" exigidos para o SAD e tem como grande atrativo o fato de ser o processador usado na maioria dos microcomputadores do tipo PC/XT fabricados no Brasil. Além de facilitar de maneira significativa o desenvolvimento de software, esta escolha também simplifica bastante a interconexão do SAD com este tipo de computador.

Como elemento importante do sistema, implementou-se um sistema de DMA, dada a rapidez com que devem ser efetuadas as transferências de dados nas conversões A/D e D/A, conforme discutido no capítulo 2. Com transferências por DMA, a taxa de amostragem pode chegar a 100 kHz, o que é suficiente para se estudar uma grande variedade de sinais. Tipicamente estes sinais podem ser de áudio, voz, biológicos, sismológicos, etc.

Uma outra característica importante no sistema desenvolvido é sua capacidade relativamente alta de armazenamento de dados. Como discutido no capítulo 2, quando se trabalha com altas taxas de amostragem, e particularmente no SAD que gera amostras de 12 bits, a capacidade e velocidade de armazenamento passam a ser pontos muito importantes. Fez-se a opção de armazenamento dos dados em memória semicondutora, com capacidade inicial de 2 Mbytes, expansível a 16 Mbytes. 2 Mbytes é suficiente para armazenar, por exemplo, até 33 segundos de áudio, a uma taxa de 40 kamostras/s.

O controle das conversões A/D e D/A e das funções do sistema, que é basicamente a geração de ondas de controle adequadas, é feito através de um circuito que utiliza uma interface paralela e um "TIMER" programável.

4.1 - UNIDADE DO PROCESSADOR

Esta unidade representa uma das partes fundamentais do SAD. Foi desenvolvida de forma compacta e apresenta uma configuração típica de microcomputador. Com capacidade para operar independentemente de outras unidades existentes no SAD, a unidade de processador pode ser vista como dois blocos básicos: CPU e Sistema de Comunicação e Entrada/Saída (E/S).

O bloco da CPU é constituído pela própria CPU 8088, memória EPROM de 16 kbytes, memória RAM de 8 kbytes, Controlador de Interrupções e circuitos auxiliares para formação das vias de dados, controle e endereços. O sistema de E/S é composto por um conjunto Teclado/Display, com controlador dedicado, Temporizador Programável, Interface Serial e circuitos adicionais para formação do canal de comunicação RS-232. A figura 4.1 apresenta um diagrama de blocos completo da unidade do processador.

4.1.1 - Unidade de Controle e Processamento (CPU)

O processador 8088 foi concebido de forma a ser compatível com sistemas desenvolvidos com CPU's de 8 bits, como 8080 e 8085. Desta forma, ele possui características tanto de um microprocessador de 16 bits quanto de 8 bits. Algumas dessas características são: Arquitetura interna de 16 bits, Interface de barramento de dados de 8 bits, Capacidade de endereçamento direto de 1 Mbytes de memória, etc.

Com a capacidade de endereçamento do 8088, não é possível acessar diretamente os 2 Mbytes de memória (expansível a 16 Mbytes) especificada para o SAD. Para contornar este problema, implementou-se um circuito que gera oito linhas de endereços adicionais usando um registrador de oito bits, que é visto pelo processador como um dispositivo de E/S. Esta solução permite ao SAD manipular memórias de até 16 Mbytes.

Para formar as vias de dados do sistema (endereços, dados e controle), fez-se uma divisão em dois grupos. Um, que inclui as oito linhas de endereços adicionais, define as vias usadas por todo o sistema. O outro define as vias usadas internamente na placa do

processador. A figura 4.2(A) apresenta o circuito completo da formação das vias do SAD. Pode-se destacar neste circuito os sinais DEN/ e ALE, usados na demultiplexação das linhas de dados e endereços (A0/D0 - A7/D7), os sinais MEMW/, MEMR/ e M/IO, que geram alguns dos sinais de controle, e o circuito que fornece o relógio do sistema, o sinal de inicialização ("RESET") e o sinal "READY" (usado para que o 8088 possa operar com dispositivos lentos). Este circuito trabalha com um cristal de 15 MHz, fornecendo uma frequência de operação de 5 MHz para o 8088 e todos os periféricos.

Na unidade do processador também estão incorporados alguns periféricos e circuitos adicionais. Na figura 4.2(b) é mostrado o circuito de endereçamento dos diversos dispositivos da unidade do processador, destacando-se os sinais P0/-P7, CSRAM/ e CSROM/. Na figura 4.2(c) estão os circuitos de alguns periféricos importantes, que completam uma configuração mínima de operação. No circuito estão incluídos:

EPROM - Uma unidade de armazenamento não-volátil é fundamental a qualquer sistema baseado em microprocessador. É na EPROM (Erasable Programmable Read Only Memory) que fica armazenado o programa monitor que possibilita as operações do SAD. A capacidade desta memória é de 16 kbytes, expansível a 64 kbytes, o bastante para armazenar um programa básico de operação. A memória EPROM utilizada é a 27128, organizada em 16384 palavras de 8 bits.

RAM - Uma memória RAM (Random Access Memory) permite que programas possam ser transferidos e executados no SAD. A unidade utilizada, 6264L, é do tipo estática e tem capacidade de 8192 palavras de 8 bits, que pode ser expandida para até 64 kbytes.

CONTROLADOR DE INTERRUPÇÕES - A implementação de serviços de interrupção num sistema baseado em microprocessador é muito interessante, principalmente pelo aumento da capacidade de manipulação de dispositivos de E/S. Com o objetivo de tornar o SAD eficiente, implementou-se um circuito para atendimento de serviços de interrupção baseado no controlador de interrupções programável 8259-A. Com isto, o sistema é capaz de trabalhar com até sete níveis de interrupção, com prioridades que podem ser estabelecidas a critério do usuário.

4.1.2 - Unidades de Comunicação e Entrada/Saída

O sistema de aquisição de dados desenvolvido pode interagir com o usuário através de um conjunto teclado/display, usado para entrar dados e executar funções manualmente. Além disso pode ser conectado com outros sistemas por meio de duas interfaces, que podem ser usadas tanto para comunicação como para operação no modo remoto: RS-232, incorporada diretamente na placa do processador, e GPIB (general purpose interface bus), construída de forma opcional.

A - Sistema Teclado/Display

O teclado é composto por 32 teclas, divididas em dois grupos 4x4. O primeiro grupo constitui um teclado hexadecimal, que possibilita a entrada de dados no sistema e o segundo forma um conjunto de teclas de comandos pré-programados, que possibilita ao usuário executar as várias funções do sistema.

O display possui 16 unidades de sete segmentos, também divididos em dois grupos, que permitem a visualização de endereços, dados, comandos, etc.

A figura 4.2(d) apresenta o circuito completo do conjunto teclado/display. Foi utilizado o controlador programável para teclado/display 8279. A importância no uso de um controlador dedicado está na economia do tempo que seria necessário para que a CPU executasse esta tarefa por software. O CI 8279 opera com um relógio interno de 100 kHz, de modo a garantir um tempo de varredura para o teclado de 5,1 ms e um tempo de "debounce" de 10,3 ms.

B - Interface Serial RS - 232

As interfaces de comunicação são fundamentais nos sistemas de computação em geral. Particularmente, a interface serial RS-232 é tida como padrão na grande maioria dos sistemas que usam microprocessador. A interface RS-232 implementada opera a uma taxa de até 19,2 kbits/s,

permitindo, além da transferência de dados, que o SAD opere em modo remoto, recebendo comandos de outro computador.

A figura 4.2(e) apresenta o circuito da interface serial RS-232. O circuito é baseada na interface de comunicação programável 8251-A, que permite a conversão de dados paralelos em seriais e vice-versa.

As saídas de transmissão e controle do CI 8251-A, que estão em níveis TTL (0 - 5 V), são convertidas para o padrão RS-232 (± 12 V) através do conversor de nível 1488. Na recepção acontece o contrário, com o padrão RS-232 sendo convertido para TTL através do conversor 1489. A frequência que serve de base para determinar a taxa de transmissão é fornecida pelo temporizador programável 8253-5.

Os parâmetros da comunicação serial, como taxa de bits, comprimento da palavra, número de "stop-bits", tipo de paridade, etc., que definem o modo de operação da RS-232, são especificados por "software".

C - Interface GPIB (General Purpose Interface Bus) - Padrão IEEE 488

Uma interface GPIB no SAD, em princípio, executa as mesmas funções da serial RS-232. A diferença básica está no formato e velocidade de transferência dos dados. Os dados são transferidos de forma paralela e a taxa de transferência pode ser superior a 300 kbytes/s.

O circuito desenvolvido e implementado, mas que ainda não foi posto em operação, é apresentado na figura 4.2(f). A interface é baseada no 8291A TALKER/LISTENER (INTEL), que é responsável pela interligação entre o microprocessador 8088 e o padrão IEEE-488, e conectada às vias através dos transceptores GPIB 8293 (INTEL), que provêm todas as especificações elétricas do padrão IEE-488.

As transferências de dados no SAD através da interface GPIB são feitas por DMA. Para conexão de um determinado microcomputador com o SAD é necessário que o mesmo possua, além de uma interface GPIB, um controlador GPIB.

4.2 - UNIDADE DE CONTROLE DE DMA

Como discutido no capítulo 2, as transferências de dados só conseguem atingir taxas elevadas quando se usa a técnica de DMA. A implementação de DMA no SAD trouxe os seguintes resultados: transferências rápidas de dados nas conversões A/D e D/A, facilidade nas operações de "refresh" e condições de se utilizar todo o potencial da interface GPIB.

A base do sistema de DMA do SAD é o controlador 8237-2, que possui uma capacidade de operação com quatro dispositivos simultaneamente, pode endereçar até 64 kbytes de memória e possibilita transferências a uma taxa de até 1,6 Mbytes/s. Mas, como o processador 8088, o controlador de DMA 8237-2 não consegue operar com a memória de 2 Mbytes incorporada. A solução adotada neste caso foi o desenvolvimento de um sistema de endereçamento e contagem de transferências que substitui os existentes no controlador. Este sistema auxiliar pode operar com memórias de até 16 Mbytes.

4.2.1 - O Controlador de DMA 8237-2

A - Características Gerais

- Quatro canais de serviço independentes;
- Controle individual das linhas para requisição de DMA;
- Capacidade de auto-inicialização independente para cada canal;
- Inicialização de blocos de memória;
- Transferências de memória para memória;
- Possibilidade de incremento ou decremento de endereço;
- Relógio de 5 MHz;
- Capacidade de expansão para qualquer número de canais;
- Interrupção de transferência por "hardware";
- Requisição de DMA por "software" e por "hardware";
- Controle de polaridade independente para os sinais de requisição e aceitação de DMA;
- Transferências por blocos de até 64 Kbytes.

B - Operação de DMA

O controlador de DMA 8237-2 trabalha basicamente em dois ciclos: (a) inativo e (b) ativo. O primeiro é assim denominado quando o controlador não é requisitado para serviço. Neste ciclo, as linhas DREQ (DMA Request) e CS (Chip Select) são analisadas a cada ciclo de relógio. Quando o sinal CS for para nível baixo, o controlador entra na condição de programa e a CPU pode ler ou fazer alterações nos seus registros internos. O controlador entra no ciclo ativo no instante em que algum canal solicita serviço através da linha DREQ. Quando isto ocorre, a linha HRQ (Hold Request), ligada diretamente à CPU, é ativada pelo controlador. A CPU responde com o sinal HLDA (Hold Acknowledge), liberando as vias de dados, controle e endereços. Os serviços de DMA podem ser realizados de três modos:

1 - Modo de Transferência Única

Neste modo o dispositivo é programado para fazer apenas uma transferência a cada pedido de DMA. A linha DREQ deve ser ativada e assim permanecer até que o controlador responda com o sinal DACK (DMA Acknowledge). Este sinal permanece ativo até que a transferência seja efetivada. No final da operação, o sinal HRQ é desativado e as vias de dados, controle e endereços retornam ao controle da CPU.

2 - Modo de Transferência por Bloco

Se algum dispositivo dispõe de um bloco de dados para transferir para memória ou, ao contrário, deve ler um determinado bloco de dados da memória, o controlador pode efetuar a operação sem necessidade de vários pedidos de DMA. Neste caso o dispositivo ativa a linha DREQ uma única vez. A transferência se inicia quando o controlador envia o sinal DACK, que permanece ativo até que todo o bloco seja transferido ou um sinal EOP (End of Process) seja detectado. Este sinal pode ser gerado por hardware ou software.

3 - Transferência por Demanda

Neste modo, um dispositivo que deseja ter acesso à memória pode ativar a linha DREQ e assim permanecer até que os serviços de DMA tenham terminado. Quando o sinal DREQ é ativado, o controlador solicita o controle dos barramentos do sistema por meio do sinal HRQ e inicia as transferências desejadas. As transferências são feitas por etapas, possibilitando a CPU utilizar eventualmente os barramentos durante este período, quando a operação é interrompida, reiniciando depois sem causar prejuízos. As transferências só terminam quando o sinal DREQ é desativado, ou ocorre alguma interrupção (tipo EOP).

As operações de transferência, quando o controlador está no ciclo ativo, podem ser de três tipos: Leitura, Escrita e Verificação. Nas transferências de leitura, os dados são movidos da memória para um dispositivo de E/S, com as linhas MEMR/ e IOW/ sendo ativadas. Na escrita, os dados são transferidos do dispositivo de E/S para memória, com ativação das linhas MEMW/ e IOR/. A terceira e última opção caracteriza pseudo-transferências. Neste caso, o controlador 8237-2 opera tanto em leitura como em escrita, gerando endereços, respondendo a EOP, etc. No entanto, seus sinais de controle permanecem inativos.

4.2.2 - Circuito de DMA do SAD

Numa operação de DMA todos os sinais de controle, como MEMW, MEMR, IOW e IOR, e de endereços, que devem ser usados para tornar possível a transferência, são gerados pelo controlador.

No caso do SAD, fez-se necessário a implementação de um circuito auxiliar para gerar endereços e contar transferências, de forma a possibilitar operações de DMA com a memória de dados de 2 Mbytes.

Na figura 4.3(A) é apresentado o circuito completo da unidade de controle de DMA do SAD. Como o CI 8237-2 apresenta algumas das linhas de endereços multiplexadas com as linhas de dados, fez-se uma separação das vias utilizando-se transceptores de dados, que também permitem que as linhas de endereços usadas para manipulação do controlador (A0, A1, A2 e A3) operem tanto como entrada quanto como saída. O restante do

circuito é responsável por um reforço dos sinais de controle, alguns bidirecionais (como IDR/ e IOW/).

O circuito de geração de endereços e contagem de transferências, apresentado na figura 4.3(B) em conjunto com a lógica de endereçamento, é baseado no contador programável de 4 bits 74LS193. Para gerar endereços, os CI's 74LS193 foram ligados em cascata, formando um contador crescente de 24 bits, que possibilita o endereçamento de memórias de até 16 Mbytes. Os contadores estão divididos em 3 grupos de dois, formando dispositivos de E/S com 8 bits de dados. Cada grupo tem suas saídas ligadas a entradas de "latches". As saídas destes "latches", ligadas ao barramento de endereços do sistema, são controladas pelo sinal DACK correspondente a cada canal, que é também o sinal de relógio dos contadores. Quando, por exemplo, ocorre algum pedido de DMA, o sinal de reconhecimento DACK libera as linhas de endereços retidas nos "latches" no intervalo de tempo em que o dado deve ser transferido. No final da operação, quando o sinal DACK muda de nível, os contadores de endereços são incrementados. Assim, o dado é armazenado na posição de memória apontada pelo endereço incrementado na operação anterior.

O circuito auxiliar opera com os canais 1, 2 e 3 do controlador 8237-2. O canal 0 tem aplicação dedicada às operações de "refresh" da memória de dados e utiliza apenas 8 linhas de endereços, geradas pelo próprio controlador. Pode-se observar no circuito da figura 4.3(A) que todas as linhas de endereços provenientes do controlador são desativadas quando um dos canais 1, 2 ou 3 está em operação.

A implementação do contador de transferências, também mostrado na figura 4.3(B), foi feita de forma similar ao circuito de geração de endereços. Os CI's 74LS193 foram interligados de modo a formarem um contador decrescente de 24 bits. Neste caso, suas saídas não são utilizadas e o único sinal de interesse é o TC1, que sai do contador menos significativo e que indica o final da contagem. Este sinal é usado para bloquear os pedidos de DMA, finalizando as operações de transferência. Ele gera também uma interrupção para a CPU, indicando que as operações programadas terminaram (figura 4.3(A)).

4.2.3 - Operação de DMA no SAD

A menos da capacidade expandida de endereçamento, o circuito de DMA do SAD opera de forma semelhante a um circuito típico de acesso direto à memória que empregue o controlador 8237-2. O tempo de execução de um ciclo de DMA é de 6 ciclos de relógio (ver figura 4.4).

Devido à utilização do canal 0 para as operações de "refresh", o sistema de DMA do SAD pode atender a alguma solicitação com um tempo mínimo de 16 ciclos de relógio. Para dois pedidos consecutivos, o atendimento só é garantido se o intervalo de tempo mínimo entre os pedidos for de 22 ciclos de relógio (ver APÊNDICE A).

Com o desenvolvimento de circuitos auxiliares no sistema de DMA do SAD, criaram-se algumas características particulares que devem ser consideradas quando da programação do controlador 8237-2. Basicamente deve-se observar que:

- 1 - As linhas de DREQ e DACK são ativas quando em nível baixo;
- 2 - O canal 0 não pode ser usado para transferências, pois é dedicado exclusivamente às operações de "refresh"
- 3 - Na utilização do canal 1, deve-se usar auto-inicialização e apenas o modo de transferência única
- 4 - O número de bytes a serem transferidos e o endereço inicial da memória devem ser fornecidos previamente aos contadores auxiliares do controlador de DMA.

Com a implementação de um sistema de DMA no SAD, as taxas de transferência ficaram limitadas pelo conversor A/D ou D/A utilizado. Na figura 4.4 é apresentado um diagrama que mostra as operações de DMA no 8237-2 e de "hold" no 8088. Sabendo-se que o ciclo de operação das unidades de memória dinâmica é da ordem de 350 ns e que o relógio utilizado é de 5 MHz, a taxa máxima de transferência permitida para o SAD é de 220 kbytes/s (ver APÊNDICE A). Este valor é amplamente suficiente para operações de aquisição no SAD, já que o conversor A/D utilizado pode gerar no máximo 150 kbytes/s.

4.3 - UNIDADE DE ARMAZENAMENTO DE DADOS

Uma das especificações fundamentais em um sistema de aquisição de dados para operar com sinais acima de algumas dezenas de kHz é sua capacidade de armazenamento. Como discutido no capítulo 2, o tipo de dispositivo capaz de atender a este requisito, sem impor grandes restrições na velocidade de transferência de dados, é a memória semicondutora, que neste caso deve ser do tipo RAM.

As memórias do tipo RAM são disponíveis em dois modelos básicos: RAM estática e dinâmica. As RAM's estáticas são complexas do ponto de vista de fabricação e normalmente são usadas quando pequenas capacidades são requeridas (geralmente abaixo de 64 kbytes). A característica principal deste tipo de memória é seu baixo tempo de acesso, sendo por este motivo muito usada nos sistemas para processamento em tempo real. Quando se requer grandes quantidades de armazenamento geralmente se usa as RAM's dinâmicas. Além de possuírem maior densidade de armazenamento, consomem menos e são mais baratas. Ao contrário das RAM's estáticas, estas memórias possuem um tempo de acesso relativamente grande, mas que não as inviabilizam na grande maioria dos casos. Uma outra característica das RAM's dinâmicas é a necessidade de "refresh", que pode levar, em alguns casos, a complicações adicionais no "hardware" do sistema.

A memória do SAD é formada por unidades dinâmicas de 256 kbits, do tipo 41256, com capacidade de armazenamento de 2 Mbytes, expansível a 16 Mbytes. Uma memória muito grande pode ser onerosa, ao mesmo tempo em que dificulta as operações de "refresh". Por outro lado, uma memória de pequena capacidade pode limitar as aplicações do sistema. O valor de 2 Mbytes foi escolhido de modo a satisfazer as necessidades do SAD, sem elevar de modo excessivo o seu custo, a ponto de inviabilizá-lo. Com esta capacidade pode-se armazenar até 13 segundos de um sinal na faixa máxima especificada.

4.3.1 - Sistema de Memória e Endereçamento

O sistema de memória do SAD é composto pelas memórias de programa RAM e EPROM, incorporadas diretamente na placa do processador, e pela memória RAM de dados, que é construída de forma independente, como mostrado na figura 4.5. As unidades PRAM (RAM de programa) e EPROM só podem ser acessadas pelo microprocessador. Já a unidade de memória de dados pode ser acessada tanto pelo microprocessador quanto pelo controlador de DMA.

Para efeito de endereçamento pela CPU, a memória de dados é dividida logicamente em 256 segmentos de 64 kbytes. Esta forma de endereçamento é definida a partir de uma janela de 64 kbytes dentro do espaço de 1 Mbytes do 8088, selecionada pelos sinais A16, A17, A18 e A19 que definem o segmento E0000 (A16=0, A17=1, A18=1 e A19=1). Os 256 segmentos de 64 kbytes da memória de dados só podem ser acessados pela CPU um por vez, sendo o segmento específico definido pelo conteúdo do registrador de 8 bits que gera os sinais AB20 - AB27 (figura 4.2(A)). O mapa da memória de dados visto pela CPU é mostrado na figura 4.6, onde o símbolo XY indica o conteúdo do registrador. Particularmente, os sinais AB25 - AB27 selecionam páginas de 2 Mbytes (MD0 - MD7, figura 4.5). Dentro da página de 2 Mbytes, os sinais AB22 - AB24 definem um bloco de 256 kbytes e, neste bloco, os segmentos de 64 kbytes são selecionados pelos sinais AB20 - AB21.

Nas operações de DMA, a memória de dados é vista como um único bloco de 2 Mbytes, com as 24 linhas de endereços provenientes do circuito gerador de endereços sendo usadas de forma direta.

4.3.2 - Circuito de Endereçamento

As unidades de memória dinâmica (tipo 41256) são endereçadas de forma matricial e possuem os endereços de linhas e colunas multiplexados. A célula de armazenamento básica é formada por uma matriz de 512 x 512, gerando uma capacidade de armazenamento de 262.144 bits. Para acesso à memória é necessário gerar sinais de endereços tanto para as linhas quanto para as colunas. O controle das linhas é feito através de sinais RAS (Row Address Strobe) e das colunas por sinais CAS (Column Address Strobe).

Na forma como está organizada a memória do SAD, a cada bloco de 256 Kbytes está associado um sinal RAS e um CAS. Para acessar 2 MBytes são necessários os sinais RAS0 - RAS7 e os sinais CAS0 - CAS7. Na figura 4.7 tem-se o circuito completo da unidade de memória de dados do SAD, destacando-se o circuito de endereçamento. Um dos sinais mais importantes é o 2HLDAB, que é responsável pela separação dos sinais de endereços provenientes da CPU e do controlador de DMA.

4.3.3 - Sistema de "REFRESH"

Como qualquer unidade de memória dinâmica, a RAM 41456 precisa ter suas linhas de endereços ativadas periodicamente, de modo que os dados permaneçam inalterados.

Para as operações de "refresh" do SAD, optou-se por um sistema que faz uso do controlador de DMA. Para tal, utiliza-se de modo fixo um gerador de pulsos ligado ao canal 0 do controlador de DMA, operando numa frequência de aproximadamente 64 kHz, que garante pedidos de DMA a cada 15,6 ms, tempo máximo estabelecido para o intervalo de um ciclo completo de "refresh" (4 ms/256 linhas). A unidade de memória 41256 possibilita diferentes formas de se fazer as operações de "refresh". Optou-se por um esquema que faz uso apenas dos sinais RAS, garantindo uma alta impedância no barramento de dados no momento da operação, desde que os sinais CAS estejam desativados. No circuito da figura 4.7 pode-se notar um sinal denominado REFRESH/, que garante a desativação dos sinais CAS e também alta impedância nas saídas de dados das memórias.

As operações de "refresh" por DMA podem parecer à primeira vista um desperdício de tempo da CPU. Com um pedido de serviço de DMA a cada 15,6 ms e sabendo-se que a execução do serviço leva 1,2 ms (ver tempo TODISP na figura 4.4), uma operação de "refresh" no SAD consome apenas 7,5 % do tempo global disponível para a CPU (são 6 ciclos de relógio destinados a operação de "refresh" a cada 80 destinados a CPU). Portanto, as operações de "refresh" via DMA penalizam relativamente pouco a operação da CPU.

4.4 - UNIDADE DE CONTROLE DE CONVERSÕES A/D, D/A E DAS FUNÇÕES ANALÓGICAS DO SAD

Como apresentado no capítulo 3, o SAD possui uma variedade de opções para realização de conversões A/D e D/A. Todas as opções oferecidas podem ser manipuladas por "software".

A seleção adequada de unidades necessárias para as conversões A/D e D/A é feita por uma Interface Paralela Programável 8255. A interface 8255 possui três "ports" de 8 bits que podem ser usados como entrada ou saída. Particularmente para as funções do SAD, os três "ports" são usados apenas como saída. O uso desta interface garante uma maior flexibilidade nas operações de configuração das funções do SAD.

Foi visto também, que uma operação de conversão A/D ou D/A envolve sempre o uso de vários sinais de controle, principalmente quando se trabalha com as amostras representadas em 12 bits. A geração dos sinais necessários para tal operação é feito através de circuitos lógicos convencionais em conjunto com um "TIMER" PROGRAMÁVEL 8253.

4.4.1 - Circuito de Seleção das Funções Analógicas e de Geração de Taxas de Amostragem Programáveis.

Na figura 4.8 é mostrado um diagrama simplificado do circuito que controla as funções do SAD (os blocos lógicos estão descritos no item 4.4.2). A conexão ao sistema da interface 8255 e do "timer" 8253, na categoria de dispositivos de E/S, é feito através do barramento de dados e de uma lógica adequada de endereçamento.

Para atuar nas chaves eletrônicas que selecionam as opções analógicas do SAD são utilizados os bits provenientes dos "ports" A e B da interface 8255, com as seguintes funções:

- PA0-PA2 - Controlam o Ganho do Amplificador de Entrada.
- PA3 - Define o Acoplamento (CA-CC) do Amplif. de Entrada.
- PA6-PA7 - Selecionam o Filtro de Entrada do Conversor A/D.
- PB0 - Define o Acoplamento (CA-CC) do Filtro do Conversor A/D.

- PB1** - Selecciona Conversão Linear ou Compansão (12/8 bits).
PB2 - Controla Início/Fim da Conversão A/D.

Outras saídas da interface também são usadas para controles na conversão D/A e reprodução de sinais analógicos:

- PA4-PA5** - Seleccionam o Sinal de Entrada do Amplificador de Áudio.
PB3-PB4 - Seleccionam Filtro de Saída do Conversor D/A.
PB5 - Controla Início/Fim da Conversão D/A.

As taxas de amostragem no SAD também podem ser programadas, graças ao uso do "TIMER" programável 8253. Este dispositivo possui três contadores independentes de 16 bits (Cont0, Cont1 e Cont2) e pode trabalhar com um relógio de até 2 MHz. A obtenção da frequência de saída se dá com a ligação de dois destes contadores (Cont0 e Cont1) em cascata. A saída do Cont0 é utilizada como relógio de Cont1. O relógio de Cont0 é fornecido pela unidade do processador e é de 1,250 MHz. A taxa desejada é obtida com a programação adequada dos contadores no modo binário e como gerador de taxas. A taxa de amostragem na saída de Cont1 é dada por:

$$\text{Frequência de Amostragem} = (1250 \text{ kHz})/N1.N2,$$

onde N1 e N2 são os valores programados em Cont0 e Cont1, respectivamente.

O valor da frequência que se pode obter com esta configuração de circuito varia de 0,0745 Hz a 312,5 kHz. Se for utilizado um relógio de 2 MHz, é possível obter variações de frequência de 0,119 Hz até 500 kHz.

4.4.2 - Lógica de Geração de Sinais de Controle para as Conversões Analógicos-Digitais

O circuito de conversão A/D apresentado no capítulo 3 (figura 3.8), necessita de uma seqüência de sinais para gerar corretamente uma determinada amostra, fazer a formatação adequada dos bytes de dados, gerar sinais de requisição de DMA para transferência e liberar o dado no momento certo para armazenamento. Estes sinais são gerados como mostrado na figura 4.9: as descrições dos principais sinais são dadas a seguir.

- FREQ.AMOST** - Sinal gerado pelo "TIMER" programável 8253, que determina a taxa de amostragem;
- BIT-S** - Sinal usado para ativar o sinal CSTART
- CSTART/** - Sinal com a mesma freqüência do sinal FREQ.AMOST. É usado para disparar as conversões A/D;
- BUSY/** - Sinal proveniente do conversor A/D, que indica a geração de uma amostra de 12 bits;
- CLK 1** - Sinal que armazena o byte menos significativo de cada amostra;
- CLK 2** - Sinal que armazena os 4bits mais significativos das amostras A1, A3, A5, ...;
- CLK 3** - Sinal que armazena os 4 bits mais significativos das amostras A2, A4, A6, ...;
- DMAR/** - Sinal que requisita serviços de DMA ao canal 1 do sistema de DMA para transferência dos bytes armazenados pelos sinais CLK 1, CLK 2 e CLK 3;
- DMAA/** - Sinal que indica a aceitação dos pedidos de DMA;

- OC1/ - Responsável pela liberação do byte armazenado pelo sinal CLK 1;
- OC2/-OC3/ - Responsável pela liberação do byte armazenado pelos sinais CLK 1 e CLK 2;
- OE/ - Quando a opção do sistema é operar com 8 bits, este sinal libera os dados provenientes da memória EPROM;

A geração destes sinais de controle é realizado pelo circuito da figura 4.10. A operação do circuito tem início quando o bit PB2 proveniente da interface 8255 (figura 4.8) é ativado, gerando o sinal BIT-S. Este sinal ativa também o sinal CSTART/, que dispara a conversão A/D. Quando a amostra está pronta, o conversor leva o sinal BUSY/ ao nível alto e a partir dele obtém-se os sinais CLK 1, CLK 2 e CLK 3, usados para armazenar dados provenientes do conversor A/D (figura 3.8). O sinal BUSY/ também dispara o circuito de requisição de DMA, que gera o sinal DMAR/. O sinal de aceitação do pedido de DMA, indicado por DMAA/, quando recebido, gera os sinais OC1/, OC2/-OC3/ e OE/. Se o tipo de conversão que estiver sendo adotado for linear de 12 bits, o sinal OE/ é desativado (bit PB1 da interface 8255). No entanto, se a conversão for do tipo compansão 8 bits, os sinais OC1/ e OC2/-OC3/ é que são bloqueados. Neste caso o sinal de requisição de DMA passa a ser simplesmente o sinal BUSY/.

Independentemente do tipo de conversão, todos os sinais gerados na conversão A/D são desabilitados no instante em que for realizada uma conversão D/A. Além disto, os pedidos de DMA são gerados por um circuito em coletor aberto para possibilitar a conexão com outros dispositivos que utilizam o canal 1 de DMA do SAD.

4.4.3 - Lógica de Geração de Sinais de Controle para as Conversões Digitais-Analógicas

Da mesma forma que nas conversões A/D, para se efetuar corretamente as conversões D/A é necessária uma seqüência adequada de sinais de controle, que possibilitem leituras de dados da memória, a transferência desses dados para o conversor D/A e a conversão D/A propriamente. Os sinais envolvidos neste controle são mostrados no diagrama da figura 4.11, e os principais são (ver figura 3.9):

- OSCILADOR \emptyset** - Sinal gerado pelo "TIMER" 8253 (figura 4.8). Determina a taxa de conversão D/A;
- BIT-S1** - Sinal gerado pelo bit PB5 da interface 8255, para liberar o sinal OSCILADOR1;
- OSCILADOR1** - Sinal usado na geração dos sinais DREQ/ e A3/, e também para liberar os 8 bits menos significativos das amostras para conversão D/A;
- DREQ/** - Sinal usado para requisitar serviços de DMA;
- DACK/** - Sinal que indica aceitação de pedido de DMA;
- CLK1** - Sinal usado para armazenar o byte menos significativo lido da memória para formação da amostra de 12 bits para conversão D/A;
- CLK2/CLK3** - Sinais usados para dividir o byte mais significativo lido da memória em dois grupos de 4 bits;
- OC2/** - Libera os 4 bits mais significativos das amostras A1, A3, A5, ..., para a conversão D/A;
- OC3/** - Libera os 4 bits mais significativos das amostras A2, A4, A6, ..., para a conversão D/A;

A3/

- Sinal que dispara as conversões D/A.

Os sinais descritos são gerados pelo circuito da figura 4.12. O processo tem início quando o sinal BIT-S1 (controlado pelo bit PB5 da interface 8255) é ativado. Com isto, obtém-se o sinal OSCILADOR1, disparado pela primeira subida do sinal OSCILADOR0. O sinal OSCILADOR1 controla a seqüência de operações no processo de conversão D/A. A cada subida de OSCILADOR1 é feito um pedido de acesso à memória através do sinal DREQ, ao mesmo tempo em que a descida gera um pulso negativo A3/ para iniciar uma nova conversão D/A.

Como são necessários três acessos de leitura à memória de dados para cada duas conversões, os pedidos de DMA gerados por OSCILADOR1 devem ser somados a um outro sinal de requisição de serviço de DMA. O controle do pedido adicional é realizado através de uma máquina de três estados, onde a transição entre estados se dá no momento da chegada de um pulso DACK/. No estado ZERO (S0), o pedido é realizado pelo sinal OSCILADOR1; na transição do estado UM (S1) para DOIS (S2), é gerado o pedido intermediário e na transição de S1 para S2, novamente o pedido é realizado por OSCILADOR1. Na transição do estado S2 para S0, não há pedido de DMA, completando o ciclo de três leituras para duas conversões. Com uma combinação adequada dos sinais na saída da máquina de três estados, obtém-se os sinais CLK1, CLK2, CLK3, OC2/ e OC3/. As subidas de CLK1 são geradas com algum atraso nas transições S2/S0 e S0/S1. As subidas de CLK2 e CLK3 são geradas na transição S1/S2.

As ondas OC2/ e OC3/ são obtidas nos estados S2 e S0, respectivamente. Serão exatamente iguais a Y2 e Y0 (saídas do decodificador 74LS138) quando se estiver trabalhando na condição normal, sem compensação. Neste caso, o bit 12/8 fornecido pela interface 8255 será igual a 0. No caso de se utilizar compensação, OC1/, OC2/ e OC3/ serão colocadas no nível lógico 1, desativando os "latches" (figura 3.9) e eliminando os pulsos na linha DREQ gerados a partir da máquina de estados. O sinal G, gerado por uma combinação dos sinais DACK e MEMR, controla os "latches" que retêm os dados provenientes da memória de dados.

No caso da conversão ser do tipo compensação - 8 bits, o sinal OE/, que habilita as saídas das EPROM's, é gerado pelo próprio bit de controle do tipo de conversão (bit PB1, figura 4.8).

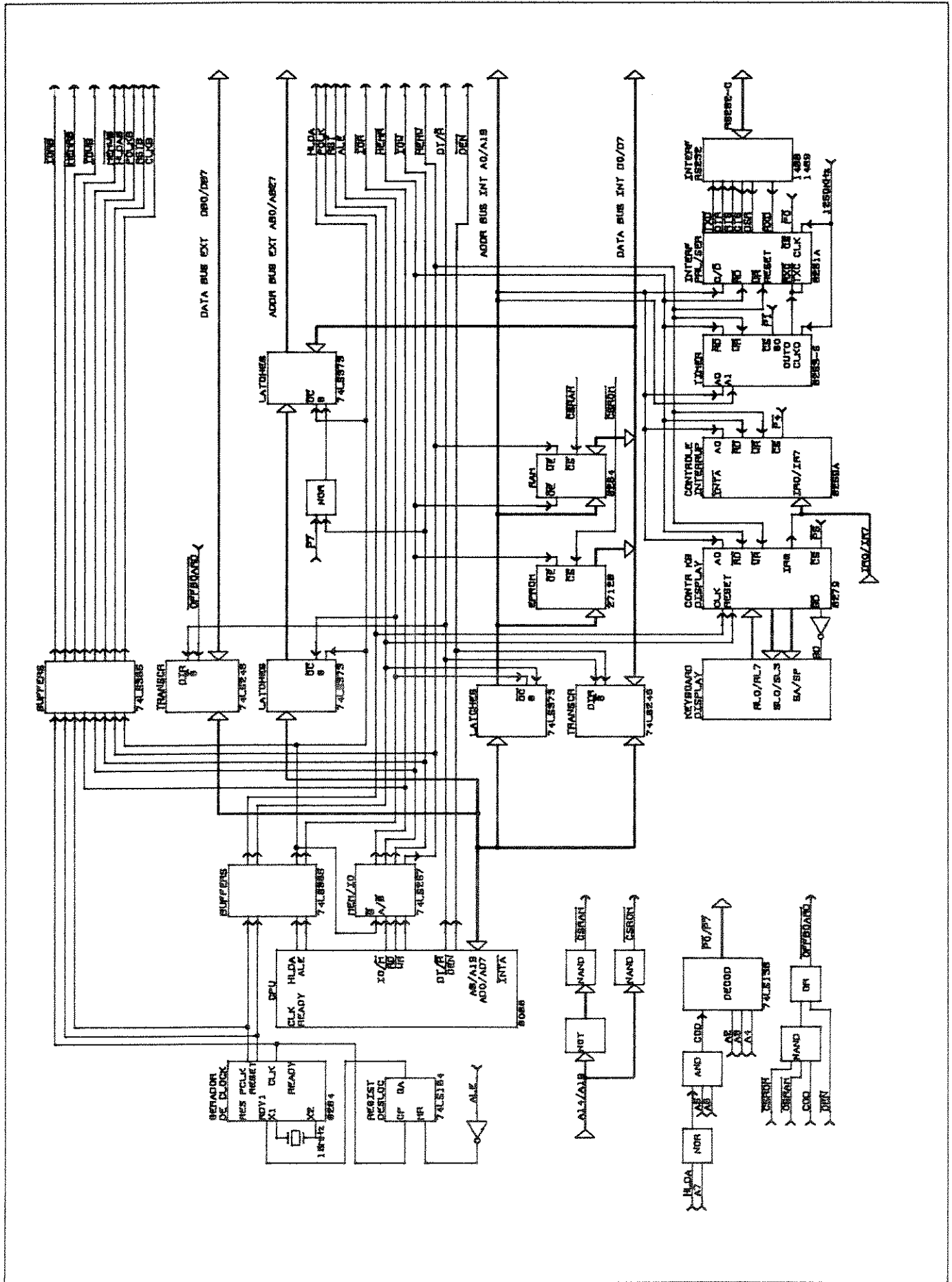


FIGURA 4 1 - DIAGRAMA DE BLOCOS DA UNIDADE DO PROCESSADOR

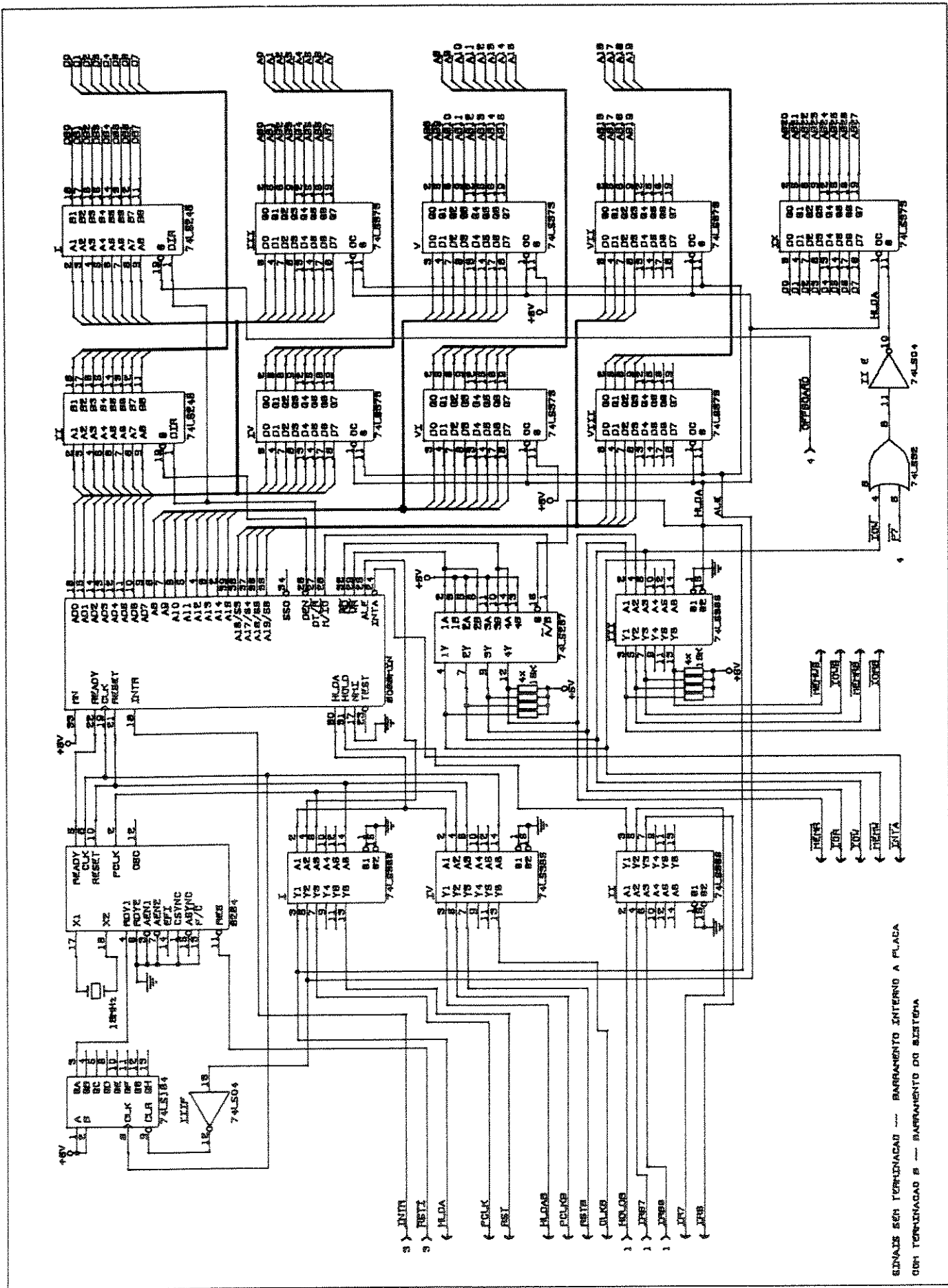


FIGURA 4.2 (a) - UNIDADE DA CPU

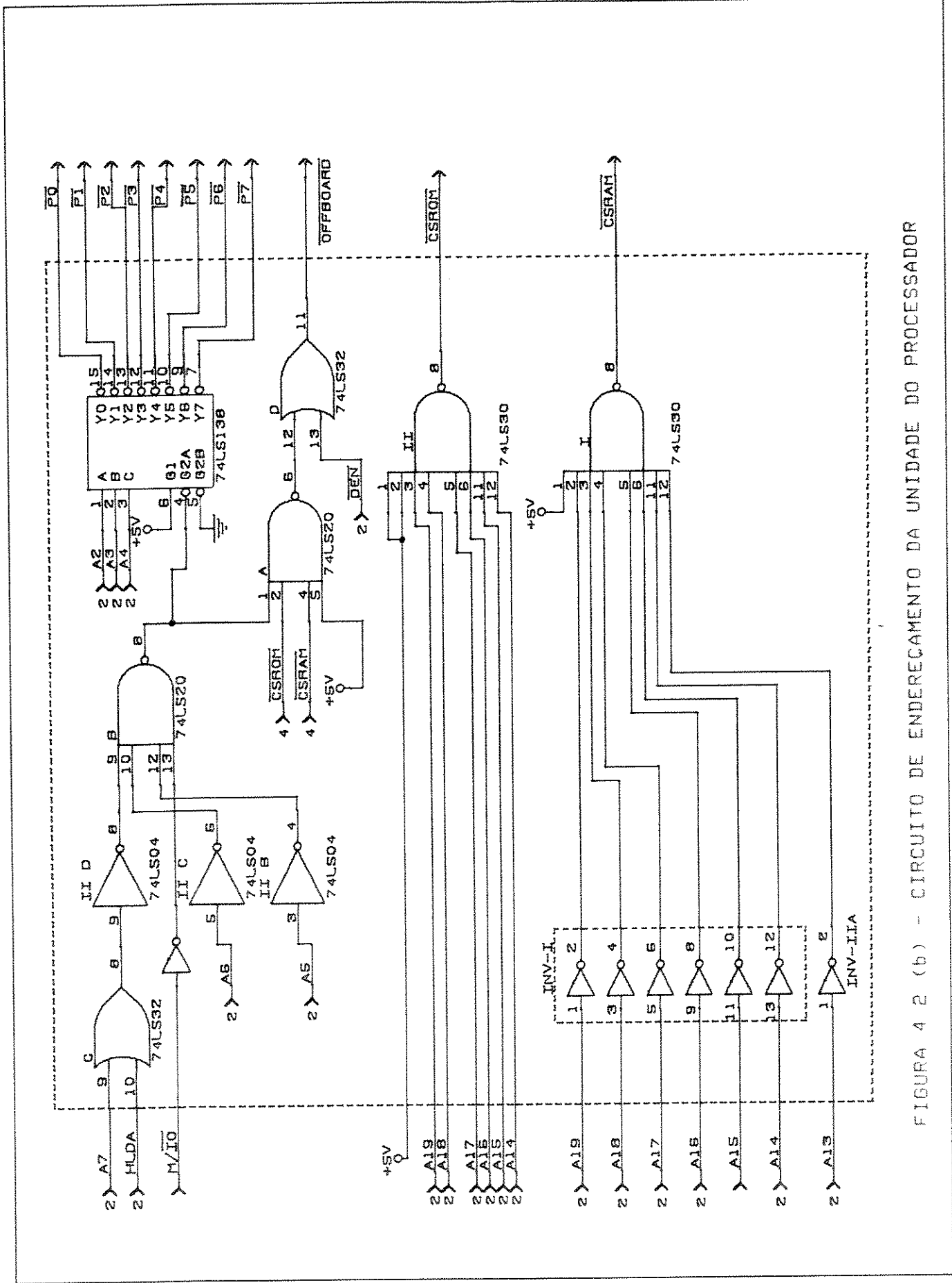


FIGURA 4 2 (b) - CIRCUITO DE ENDEREÇAMENTO DA UNIDADE DO PROCESSADOR

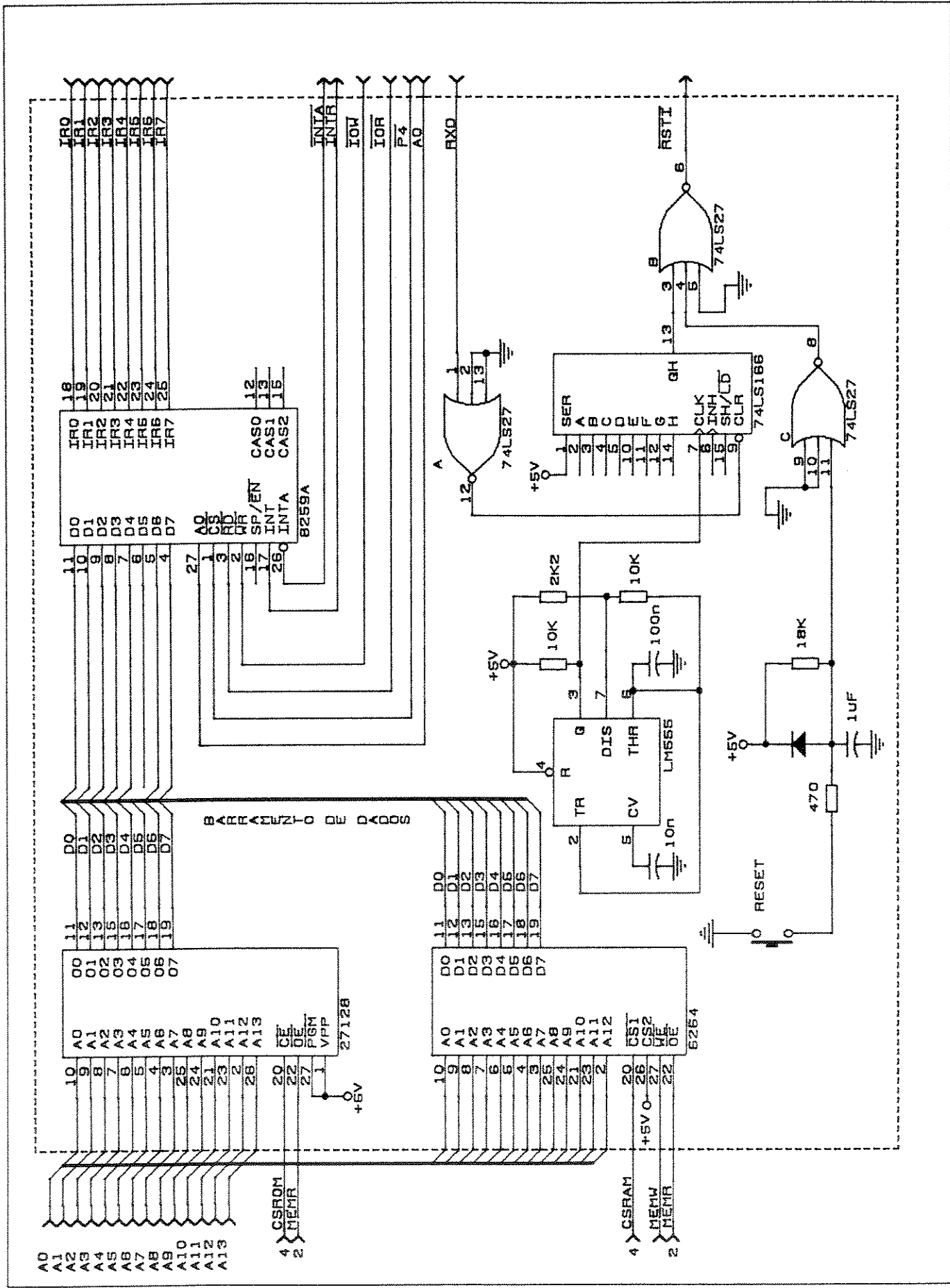


FIGURA 4.2 (c) - PERIFÉRICOS E CIRCUITO DE INICIALIZAÇÃO

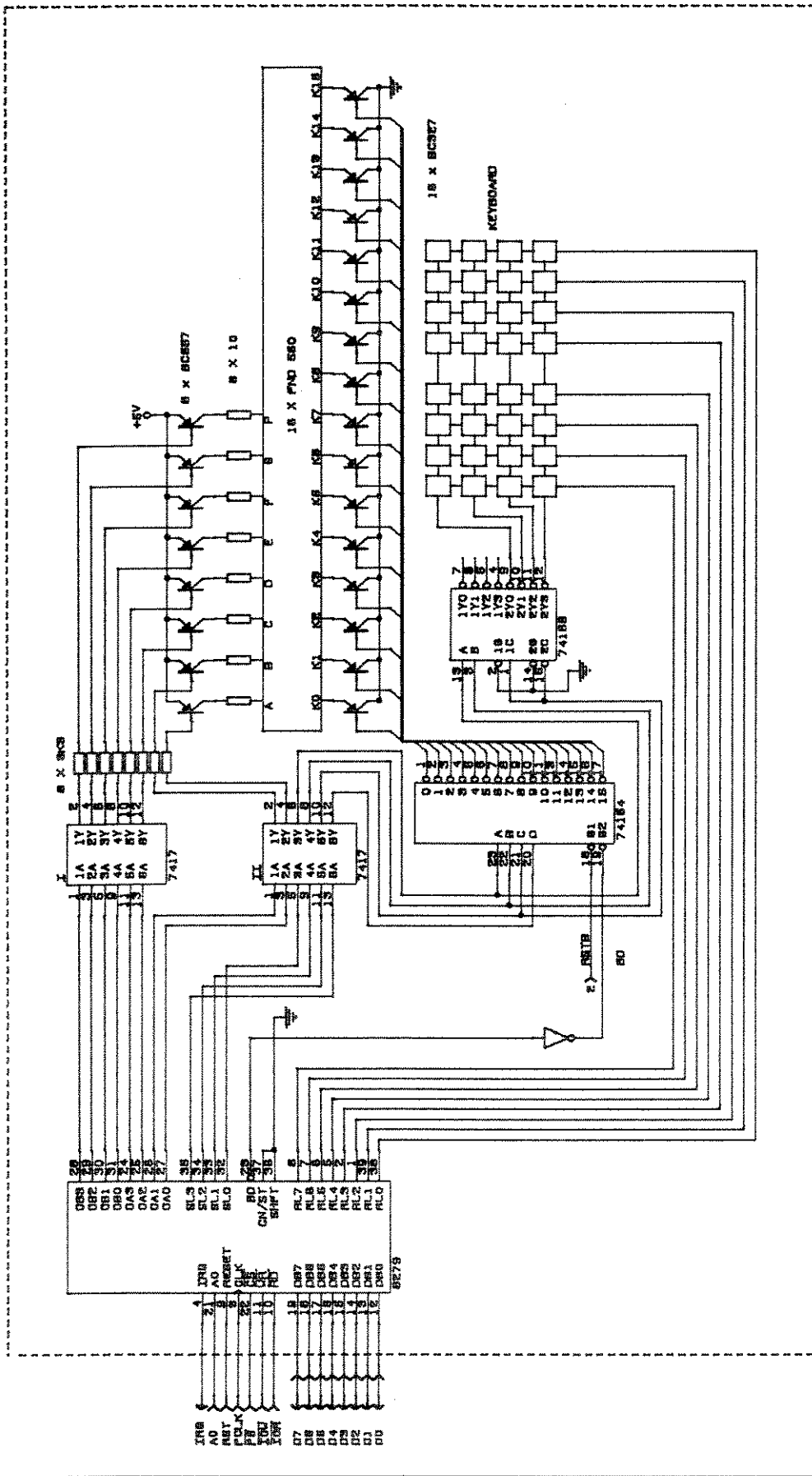


FIGURA 4 2 (d) - CIRCUITO DE CONTROLE DA UNIDADE TECLADO/DISPLAY

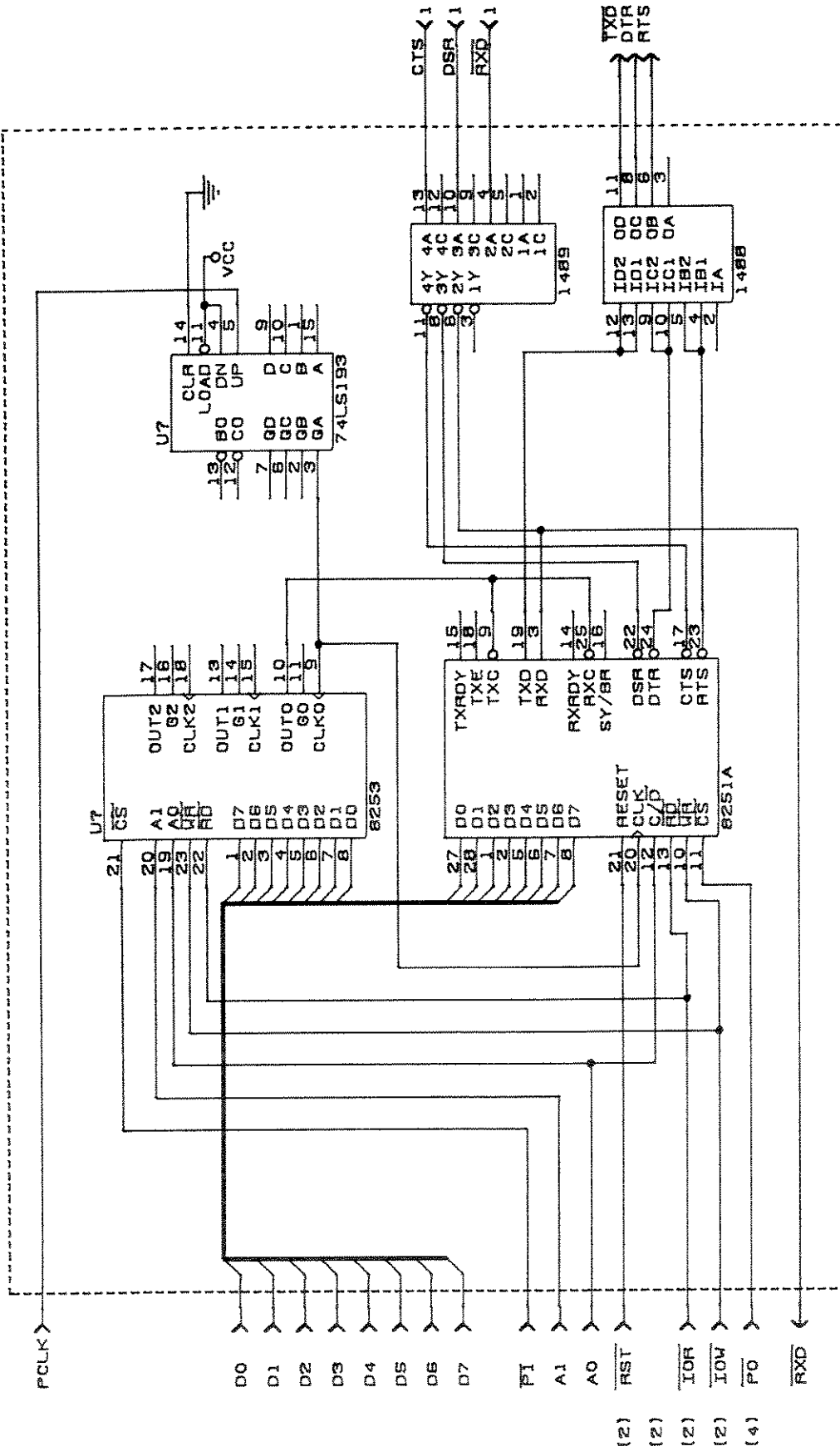


FIGURA 4 2 (e) - INTERFACE SERIAL RS-232

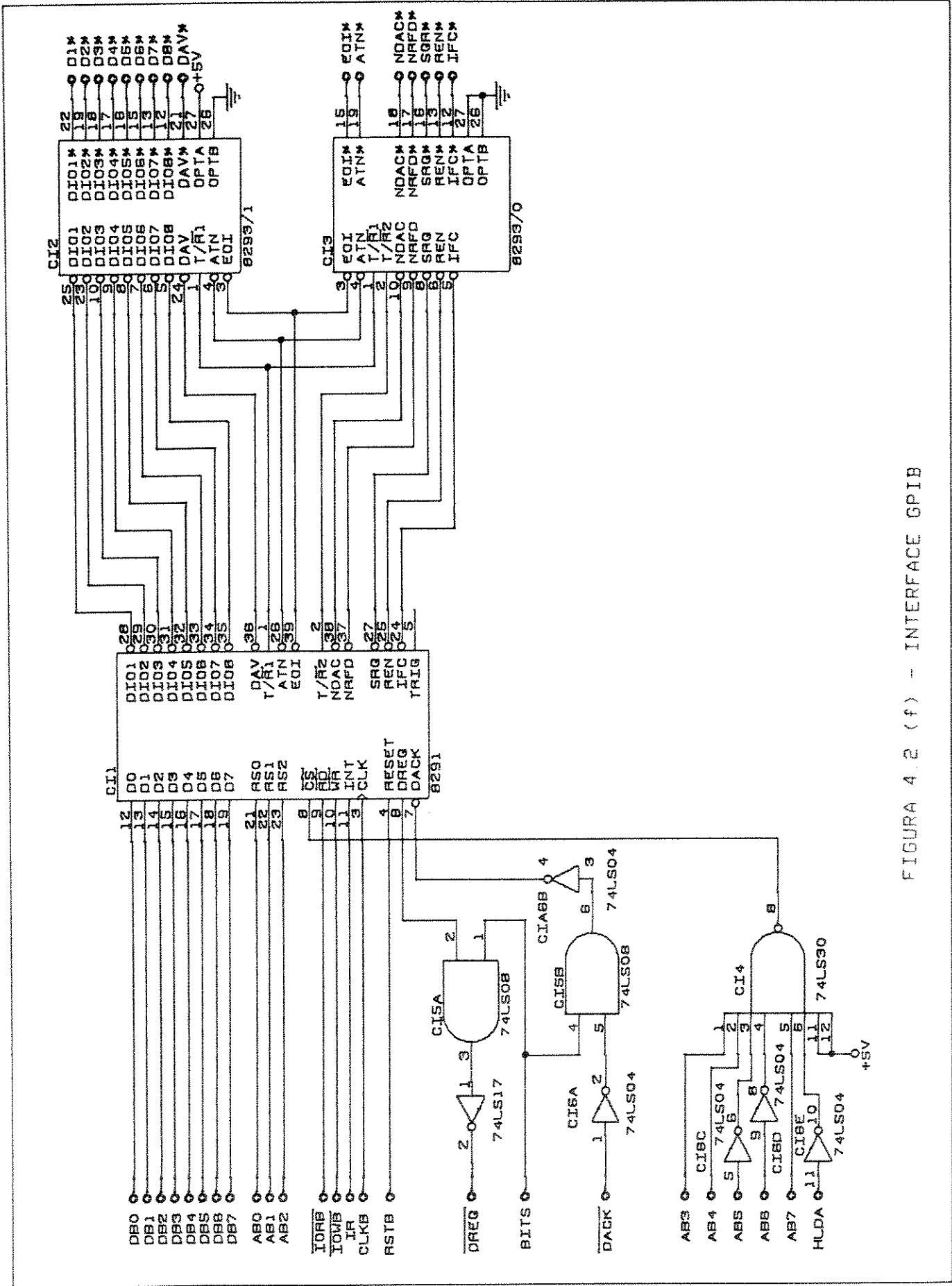


FIGURA 4.2 (f) - INTERFACE GPIB

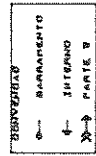
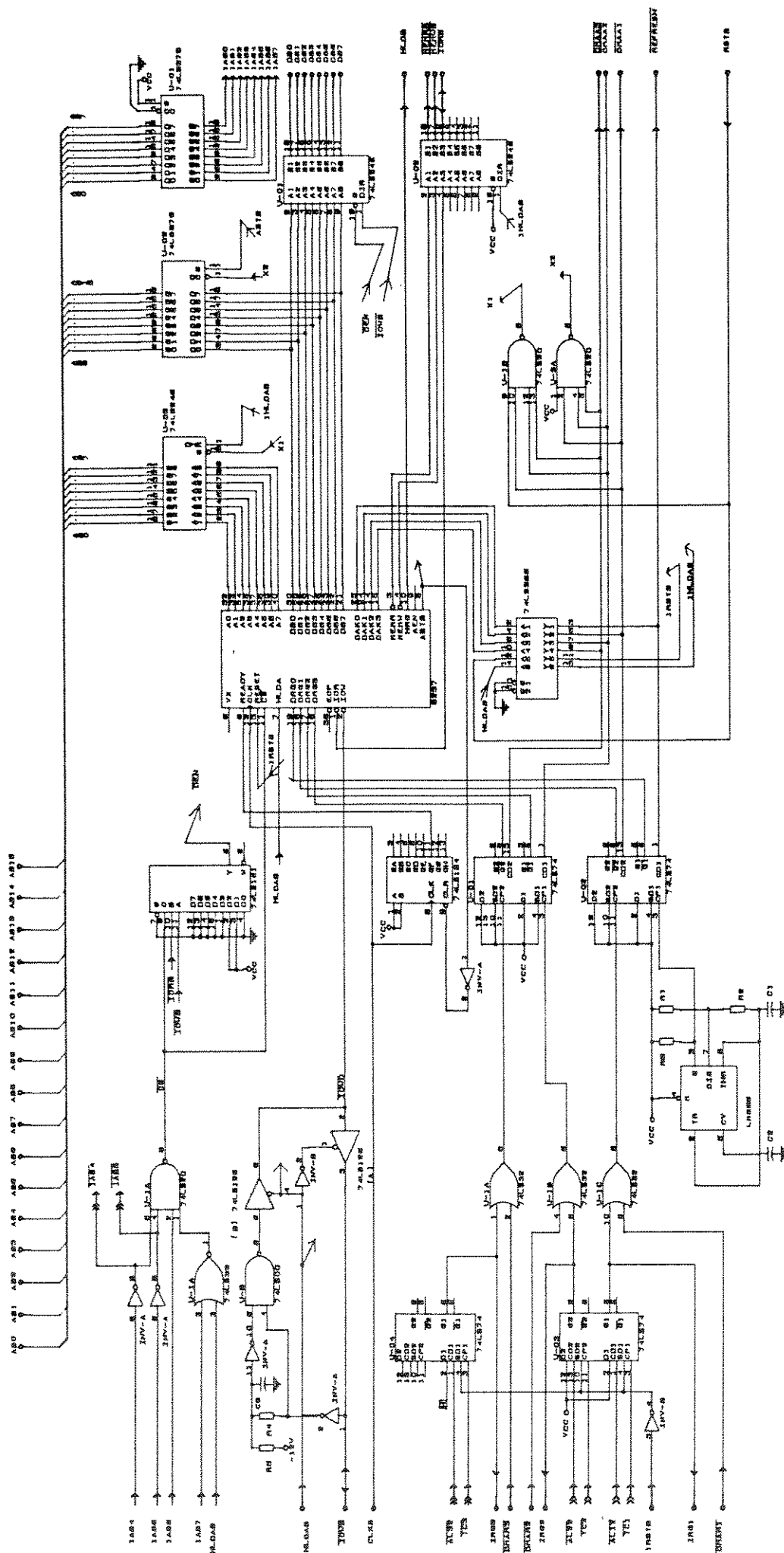


FIGURA 4.3 (a) - CIRCUITO DE CONTROLE DE DMA

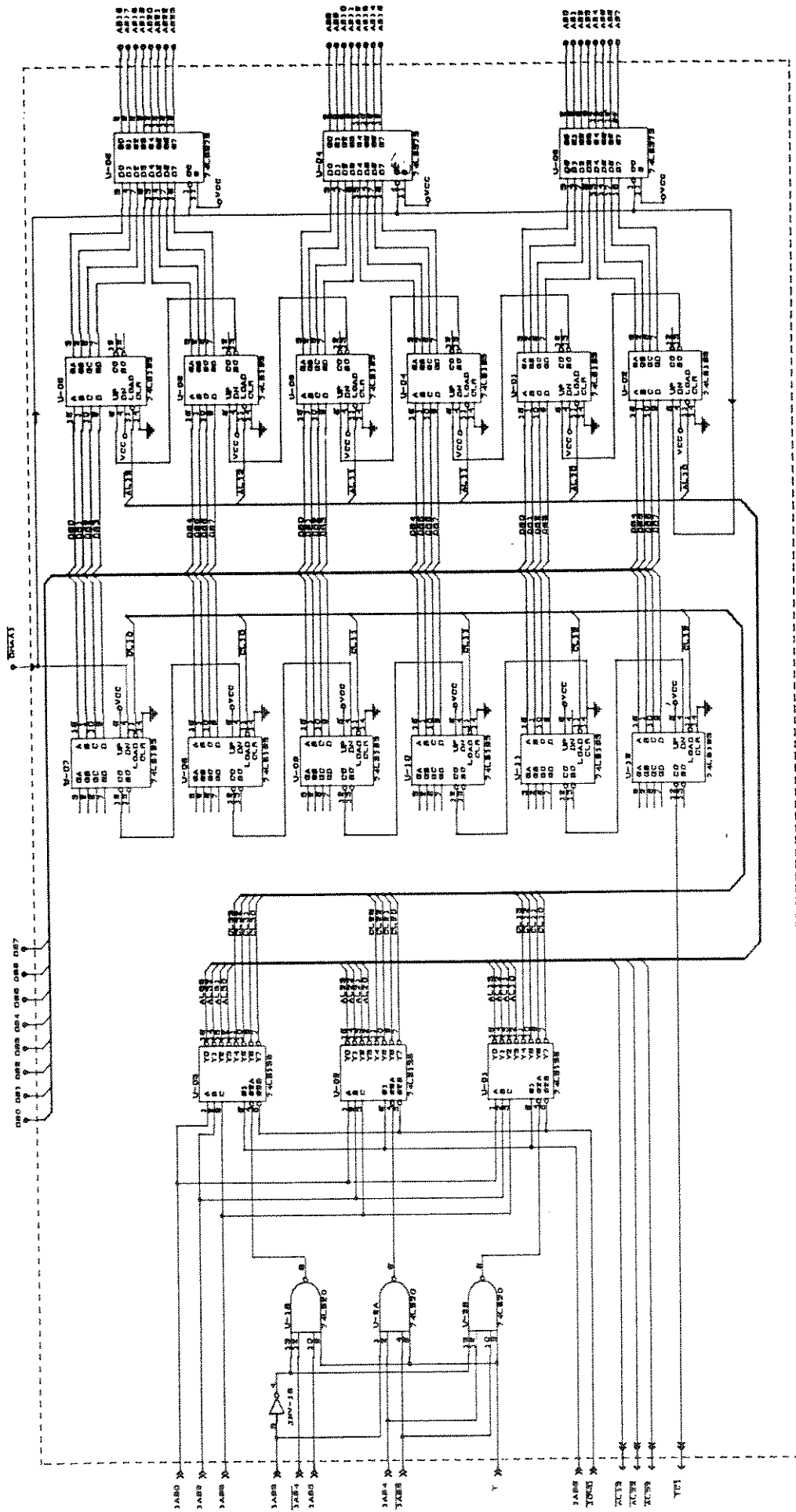
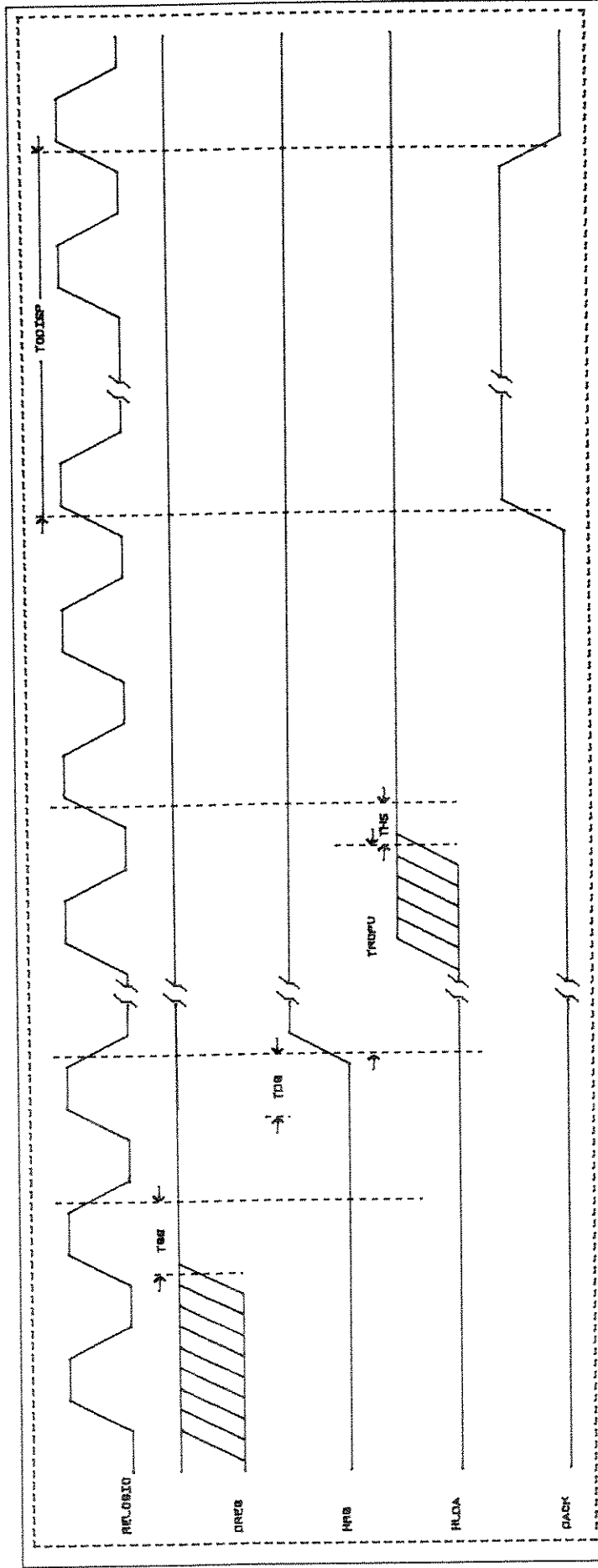
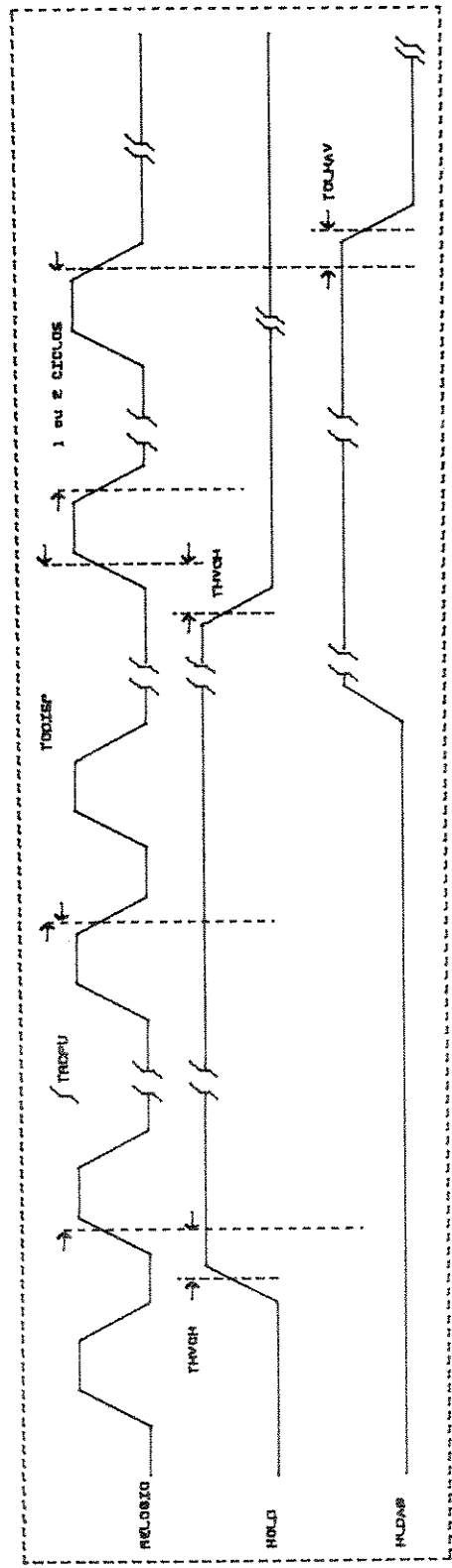


FIGURA 4.3 (b) - CIRCUITO DE GERAÇÃO DE ENDEREÇOS E CONTAGEM DE TRANSFERÊNCIAS



(A)



(B)

TDRSU - Tempo de resposta da CPU (medida de 2 ciclos de Relógio)
 TDRPU - Tempo de operação do dispositivo de DMA (8 ciclos de Relógio)
 THS - Tempo de "Setup" de HRS (mínimo de 0 ns)
 THVSH - Tempo de "Setup" de HLDV (mínimo de 75 ns)
 THVSH - Tempo de atraso para HRS válido (mínimo de 100 ns)
 THVSH - Tempo de "Setup" de HOLD (mínimo de 25 ns)
 TOLHAV - Tempo de atraso para HLDV válido (mínimo de 180 ns)

FIGURA 4.4 - OPERAÇÃO DE DMA DO 8237-2 E DE "HOLD" DO 8088

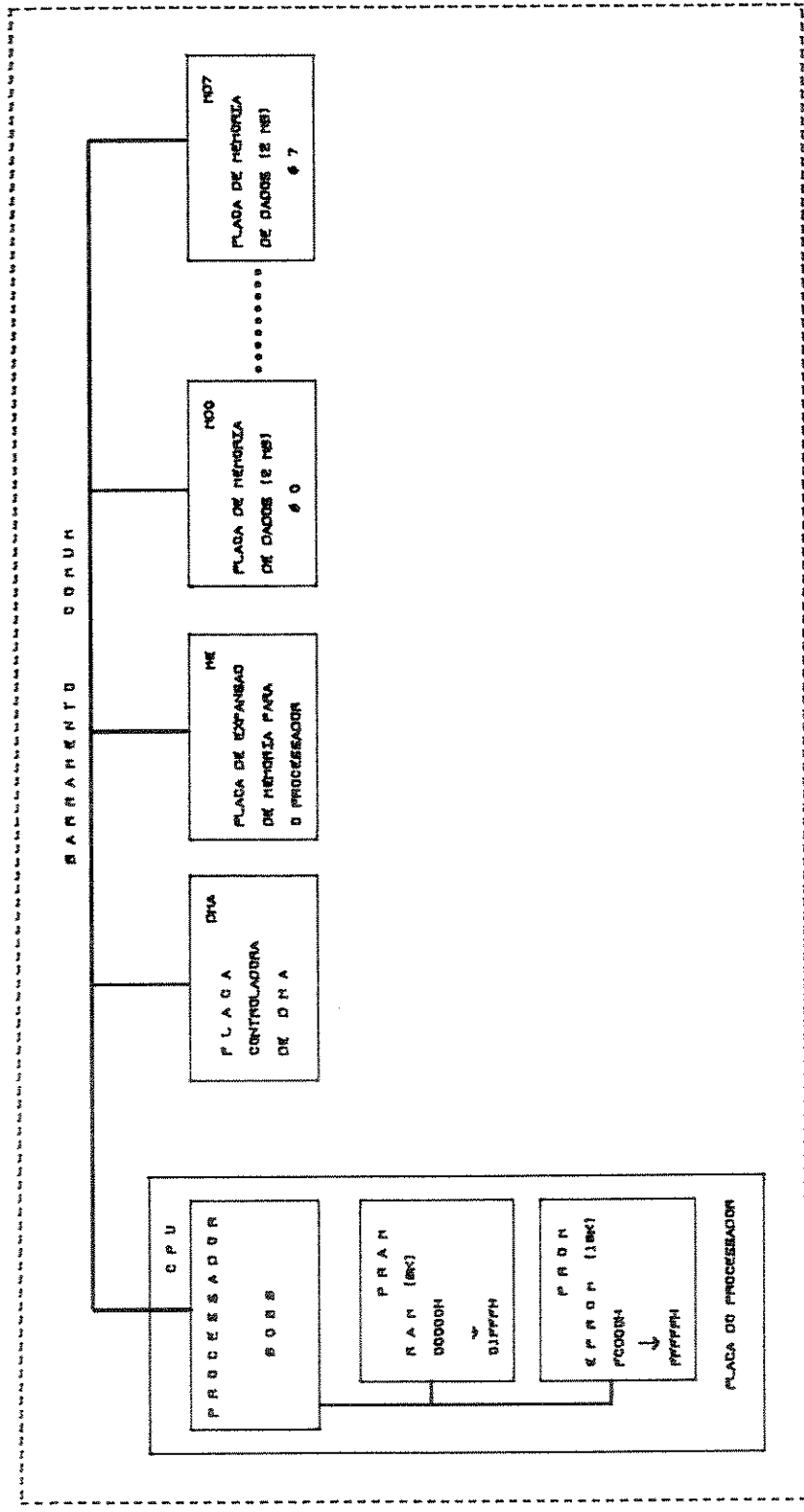


FIGURA 4 5 - ORGANIZAÇÃO DA MEMÓRIA DO SAD

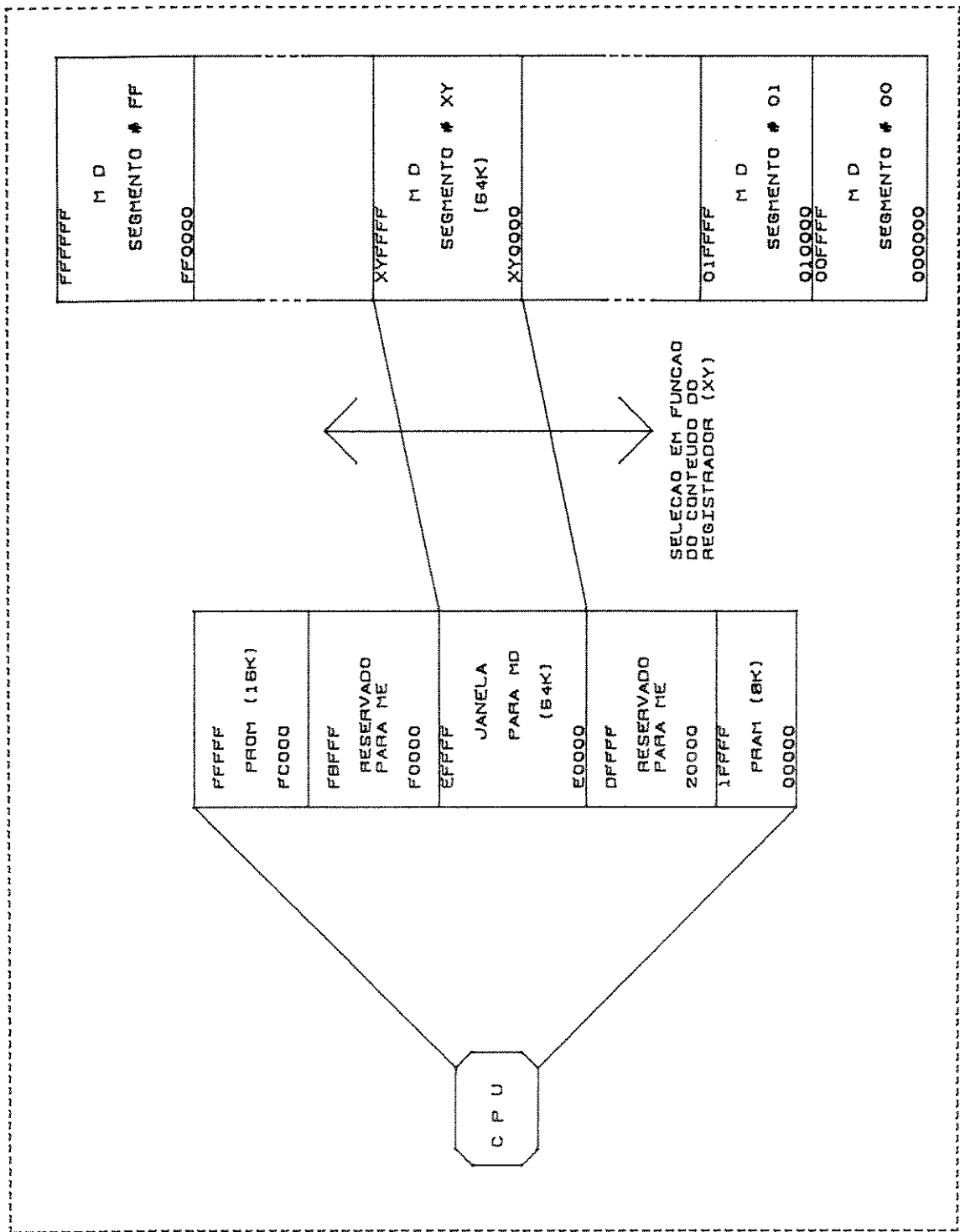


FIGURA 4.6 - SISTEMA DE ENDEÇAMENTO DA MEMÓRIA

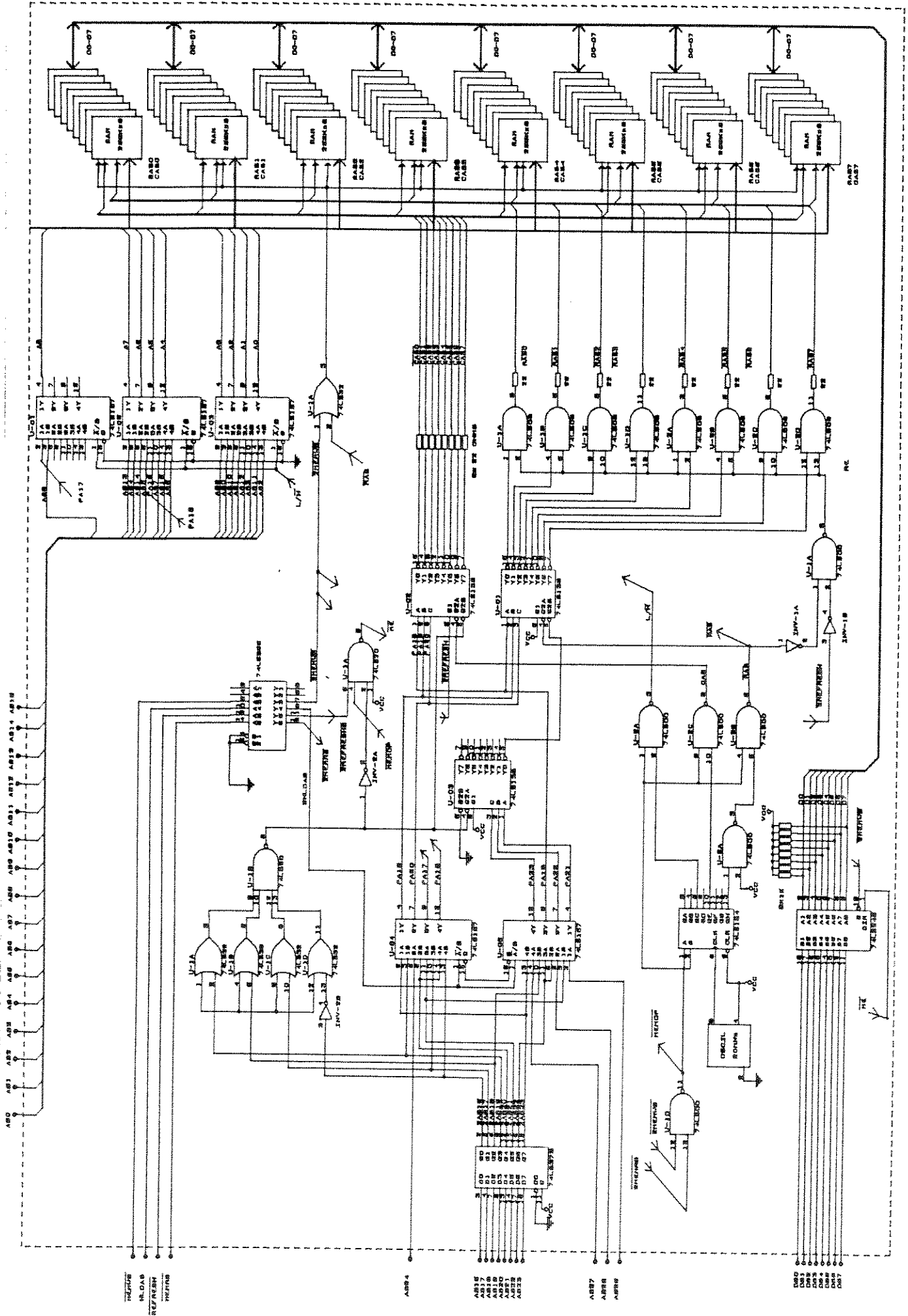


FIGURA 4 7 - CIRCUITO DE ENDEREÇAMENTO DA MEMÓRIA DE DADOS

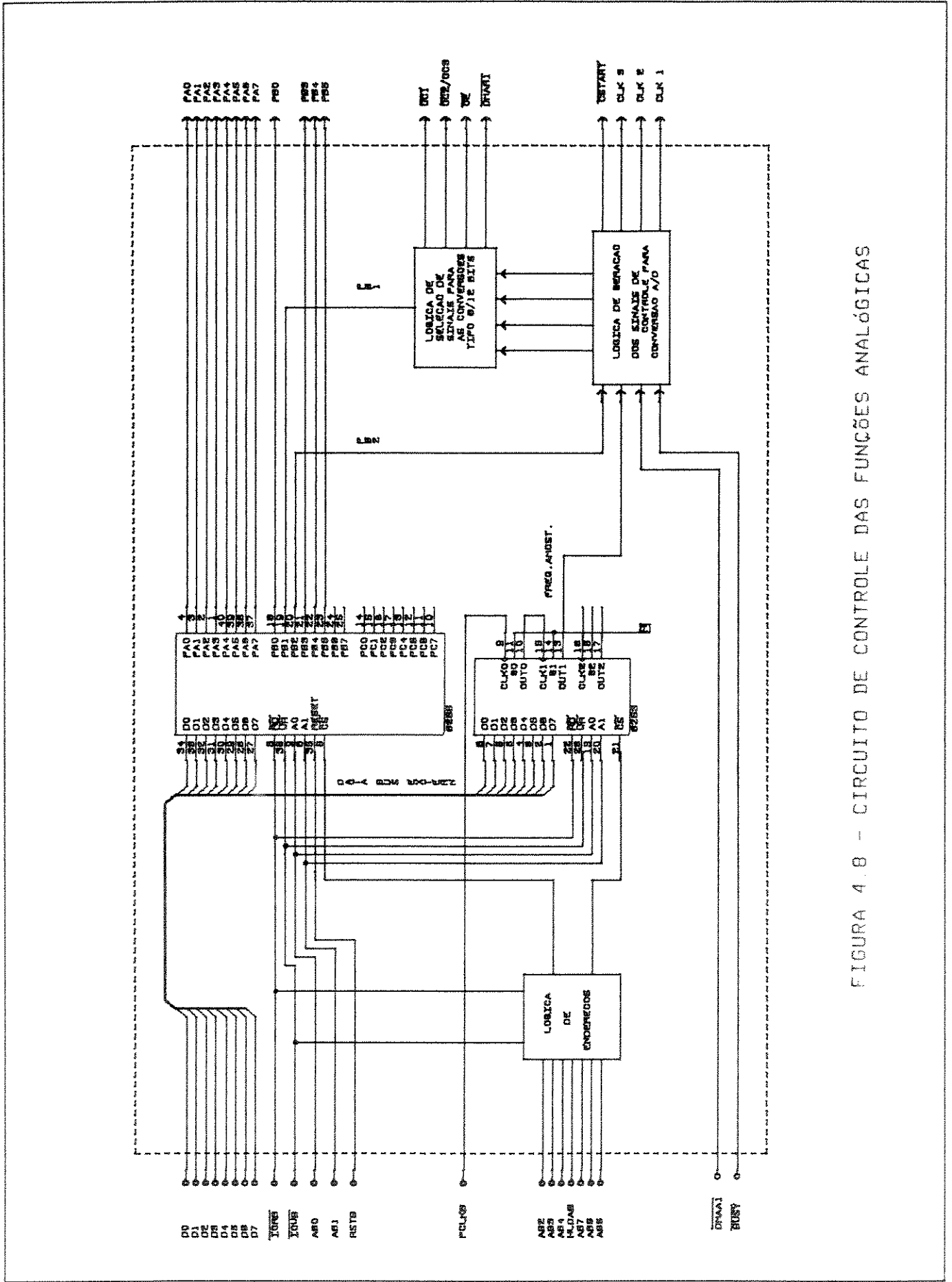


FIGURA 4.8 - CIRCUITO DE CONTROLE DAS FUNÇÕES ANALÓGICAS

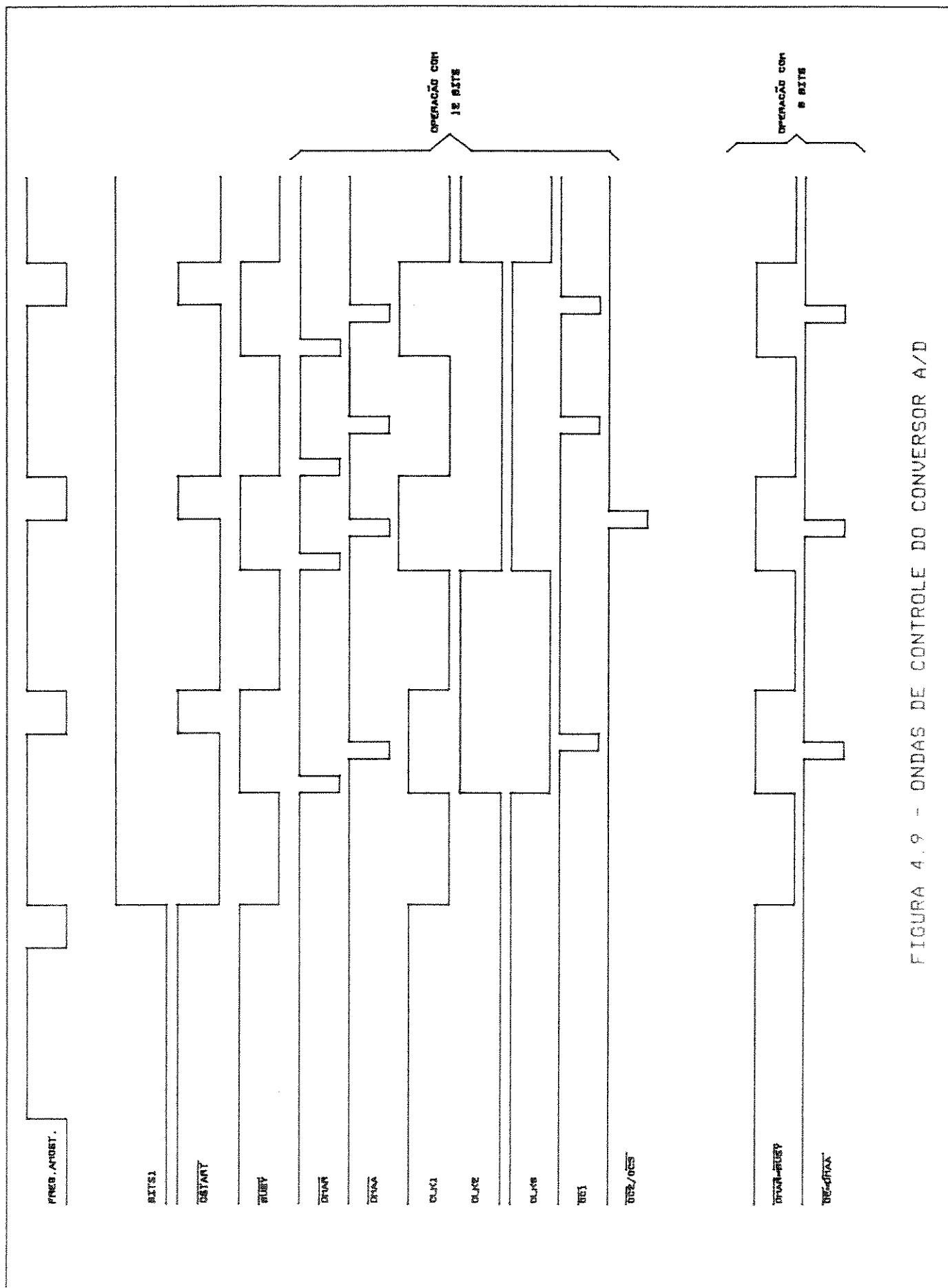


FIGURA 4.9 - ONDAS DE CONTROLE DO CONVERSOR A/D

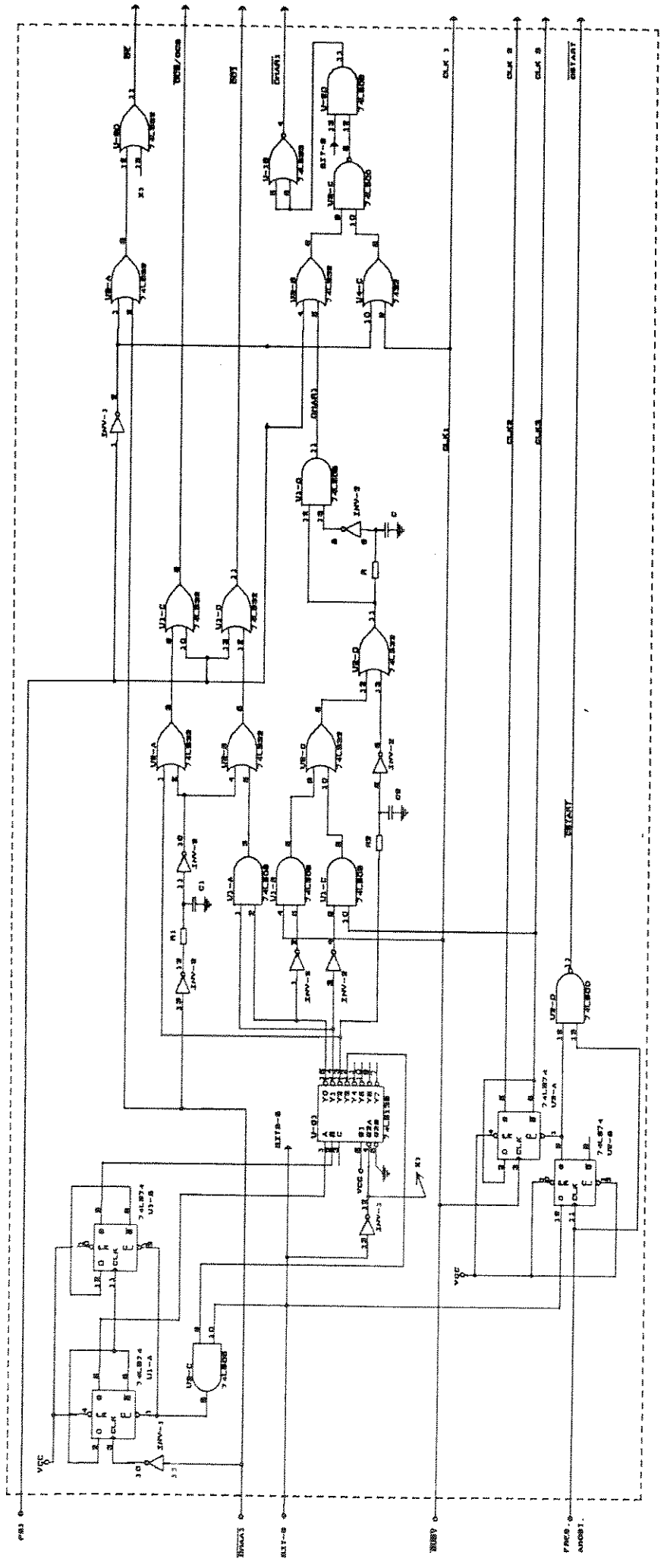


FIGURA 4.10 - CIRCUITO DE GERAÇÃO DOS SINAIS DE CONTROLE DO CONVERSOR A/D

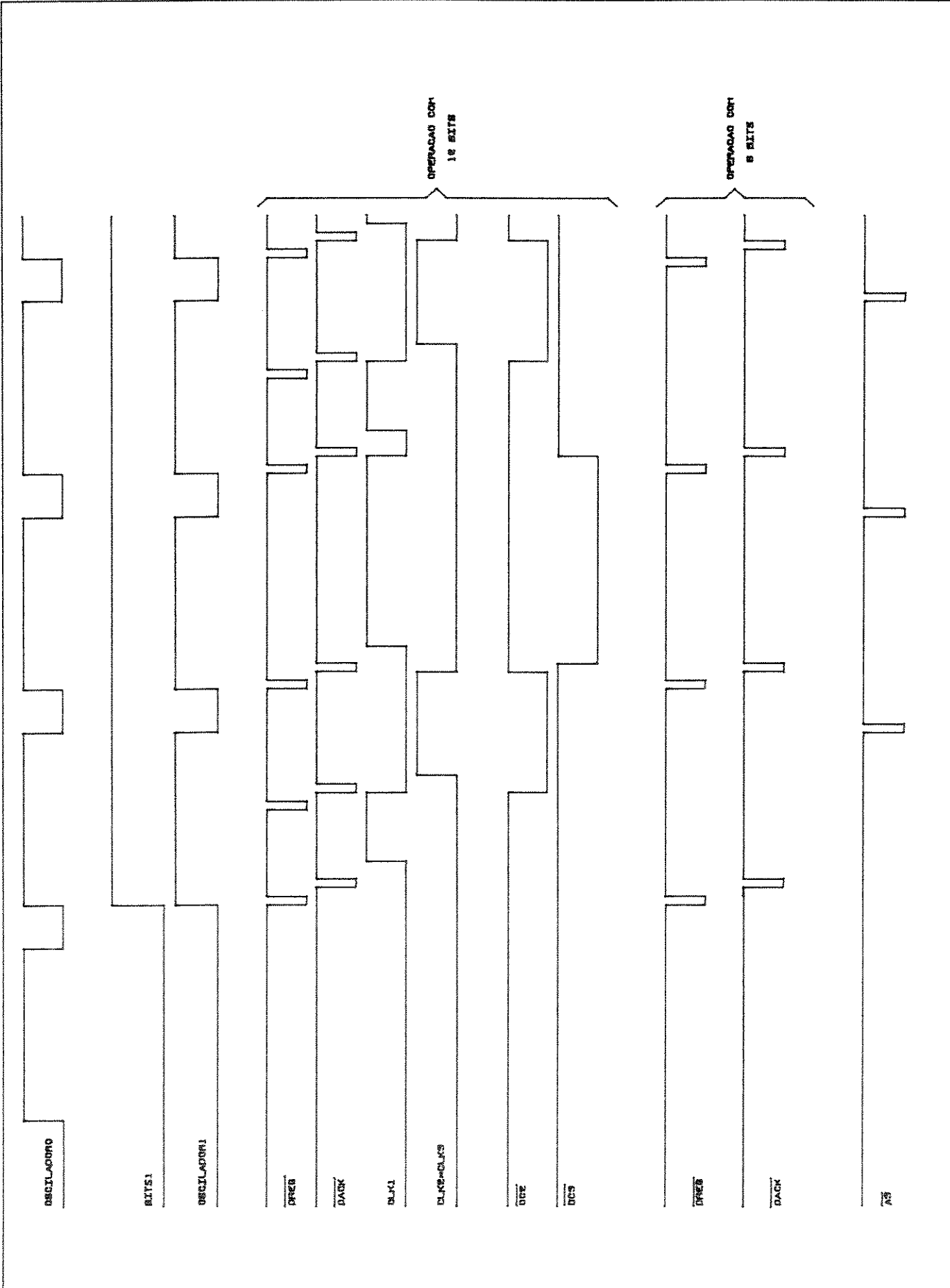


FIGURA 4.11 - ONDAS DE CONTROLE DO CONVERSOR D/A

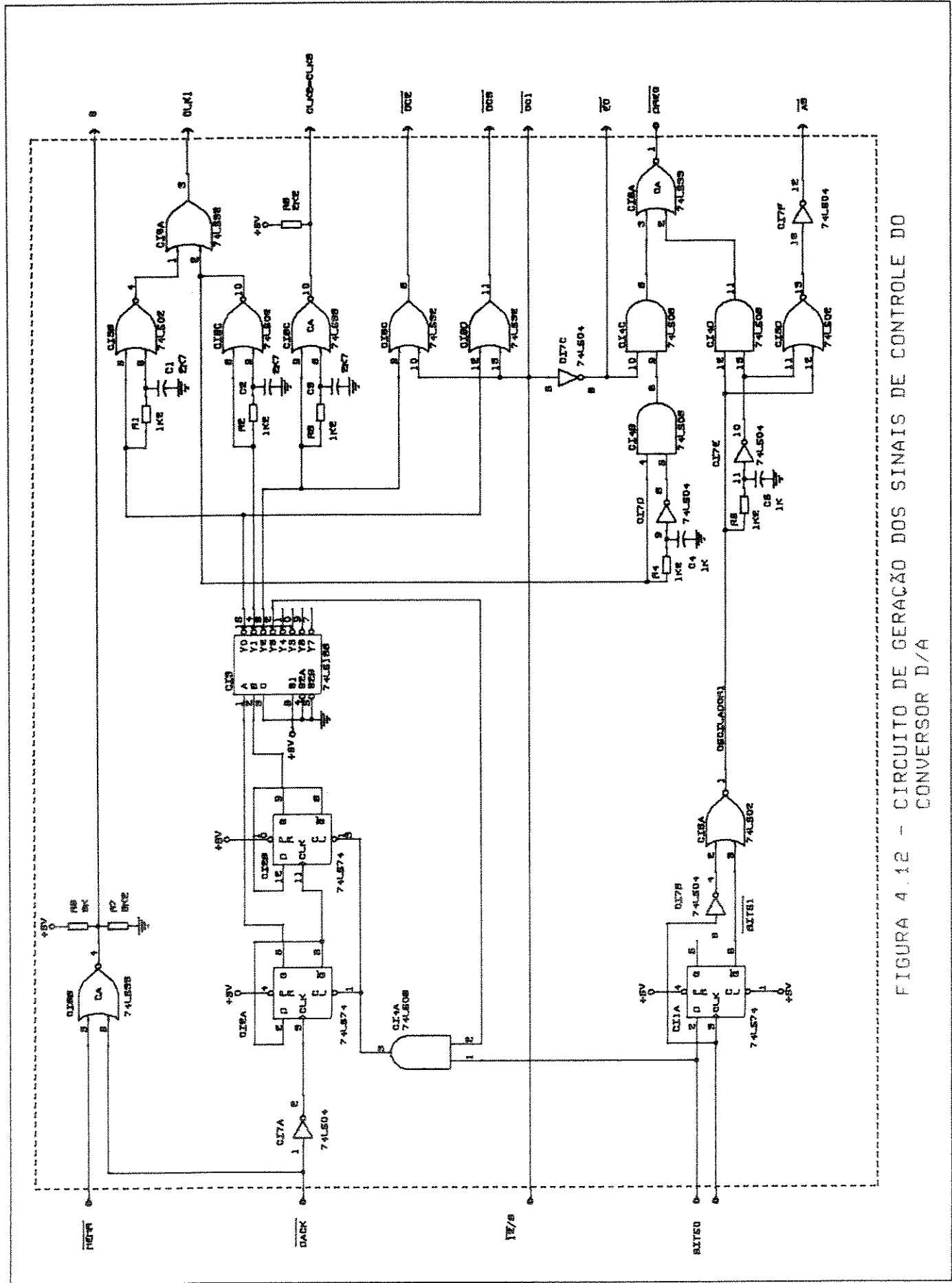


FIGURA 4.12 - CIRCUITO DE GERAÇÃO DOS SINAIS DE CONTROLE DO CONVERSOR D/A

CAPÍTULO 5 - "SOFTWARE" DE OPERAÇÃO DO SAD

No desenvolvimento de sistemas de aquisição de dados, uma das principais preocupações é a definição de um "hardware" que atenda às especificações para uma determinada aplicação. No entanto, quando se trabalha com sistemas do tipo auto-suficiente, com controle baseado em microprocessador, tão importante quanto o "hardware" passa a ser o "software", que efetivamente faz o sistema.

A operação do SAD está condicionada a dois programas, que o fazem operar de modo bastante flexível: SBSAD ("Software" Básico do SAD) e SCOMSAD ("Software" de Comunicação para o SAD). O SBSAD é o programa monitor que fica armazenado permanentemente na EPROM do sistema. É responsável pela inicialização do "hardware" e pela operação do SAD através de uma comunicação serial, via RS-232. O SCOMSAD é um programa que deve ser executado num computador tipo IBM-PC, de modo que todas as funções do SAD possam ser controladas remotamente.

5.1 - O "SOFTWARE" MONITOR SBSAD

O programa monitor do SAD (ANEXO B) foi desenvolvido em linguagem Assembly num microcomputador tipo PC-XT. A preocupação inicial foi fazer um programa que controlasse corretamente as funções básicas do sistema e que permitisse a realização de mudanças nas suas funções, sempre que necessário.

Desenvolveu-se um programa que tem como principal característica a maneira como as suas sub-rotinas podem ser acessadas. As sub-rotinas estão divididas em dois blocos: principais e de uso geral. As primeiras são as que realizam as funções principais do sistema, como as de manipulação de memória, de execução de programas, de manipulação de registros do microprocessador, etc. As sub-rotinas de uso geral executam diferentes funções e podem ser acessadas por qualquer outra sub-rotina. Como exemplo, tem-se as sub-rotinas de manipulação dos dispositivos de "hardware", as de leitura e escrita de dados na interface serial, etc.

O acesso às sub-rotinas principais é feito por códigos armazenados numa tabela, que fornecem os endereços das sub-rotinas. Esses códigos representam comandos que podem ser recebidos através da

interface serial RS-232. Quando um código de comando é recebido, o sistema busca na tabela o endereço associado com a sub-rotina desejada, executando-a no passo seguinte.

Os blocos mostrados na figura 5.1 representam as seguintes funções do "software" monitor:

1 - Início

Quando o sistema é ligado, automaticamente o registrador IP (apontador de programa) do microprocessador irá apontar para a posição de memória FFFF0. Nesta posição, uma instrução de desvio incondicional desloca IP para o início da rotina principal denominada start, que inicializa o "hardware". Devido à flexibilidade do programa desenvolvido, o endereço da sub-rotina start vai depender do número de funções presentes.

2 - Inicialização do Hardware

Neste bloco são inicializados os vários dispositivos programáveis que compõem o "hardware" do sistema, dentre os quais pode-se destacar:

TIMER 8253 (I)- Usado na determinação da taxa de transmissão/recepção da interface serial RS-232. Seu contador 0 é programado com 08 (decimal), para atuar como gerador de taxa, fornecendo uma frequência de 156 kHz. Este valor pré-estabelecido permite comunicação a uma taxa de 2400 ou 9600 bits/s, que depende da programação da interface serial 8251. Outros valores de taxas de comunicação podem ser obtidos com uma reprogramação adequada do TIMER.

TIMER 8253 (II) - Usado no circuito de geração programável de taxas de amostragem. Na configuração inicial é estabelecida sua operação como gerador de taxas, com seu contador 0 sendo programado com 02 (decimal) e o contador 1 com 15 (decimal), fornecendo uma frequência de 41,6 kHz. Este valor depende do relógio do contador 0, que no SAD é de 1250 kHz.

CONTROLADOR DE DMA 8237 - Usado nas operações de "refresh" e nas conversões A/D e D/A. A condição de operação inicial é estabelecida apenas para o canal 0, que é usado para "refresh" da memória de dados. De acordo com a configuração de "hardware", tem-se:

- Prioridade fixa nos pedidos de DMA;
- Linhas DREQ e DACK ativas quando baixas;
- Transferência com leitura de dados na memória;
- Programação para auto-inicialização;
- Transferência no modo único;
- Decremento de endereços;
- Contador de endereços com o valor 256 (decimal).

INTERFACE PARALELA 8255 - Usada na configuração das funções analógicas e no controle das operações de conversão A/D e D/A. A interface é inicializada para operar seus "ports" A, B e C para serem apenas saída. Os "ports" A e B são programados com 26 e 19 (hexadecimal), respectivamente. Estes valores estabelecem a seguinte configuração inicial para o SAD:

- Ganho de 0 dB;
- Filtro de Entrada de 16 kHz;
- Saída de áudio ativada para gravação;
- Tipo de conversão 12 bits;
- Acoplamento direto (CC);
- Filtro de saída de 16 kHz.

INTERFACE SERIAL 8251 - Usada na interface de comunicação serial RS-232. No início é estabelecida uma configuração de comunicação no seguinte formato:

Taxa de comunicação de 9600 bits/s;
Dados para transmissão/recepção de 8 bits;
Comunicação com 2 "stop" bits e sem paridade;
Comunicação assíncrona.

3 - Bloco de Comunicação

Após a inicialização dos dispositivos programáveis, dá-se início a um "loop" de leitura e escrita na interface serial RS-232. Inicialmente é enviado o código ASCII do caractere "*" (2Ah) para a porta serial, indicando ao computador externo que o SAD está em operação e pronto para estabelecer a comunicação. No passo seguinte, o sistema lê a porta serial em busca de comandos. Se não existir comando, o código 2A é novamente enviado, estabelecendo o que aqui foi denominado de "Loop" de Busca de Comando.

4 - Bloco de Busca de Comando e Execução das Funções

O computador externo conectado ao SAD pode solicitar serviços. Para tal, deve enviar o byte correspondente ao código da função a ser selecionada, que corresponde ao endereço inicial da sub-rotina principal que deve ser executada. A execução desta sub-rotina não é, no entanto, um passo único. Dentro da sub-rotina principal várias outras podem ser acessadas para completar a função desejada, sendo denominadas de sub-rotinas de uso geral. Assim, por exemplo, quando o computador externo quer ler um bloco de memória do SAD, deve enviar o código de acesso à sub-rotina principal READ (ver item 5.1.1). Esta, por sua vez, irá acessar alguns sub-rotinas de uso geral, como as que irão receber os endereços inicial e final do bloco de dados, a que irá enviar o byte de dados pela RS-232, etc.

5.1.1 - SUB-ROTINAS PRINCIPAIS

A operação inicial do SAD exigiu o desenvolvimento de várias sub-rotinas importantes, de modo a permitir, principalmente, o acompanhamento dos testes de "hardware". Posteriormente foram desenvolvidas as sub-rotinas que permitem o controle das operações de aquisição de dados propriamente ditas. A seguir é dado um resumo das funções das principais sub-rotinas desenvolvidas e os códigos correspondentes para acessá-las. Vale ressaltar que um erro no uso destas sub-rotinas pode levar a um bloqueio na comunicação SAD/COMPUTADOR.

- WRITE** - Sub-rotina que possibilita acesso às memórias do SAD para escrita. Quando um computador externo quer escrever dados na memória do sistema, deve enviar o comando de acesso a esta sub-rotina e, em seguida, o tamanho do bloco de memória que deve ser alterado, que é definido com o envio dos endereços inicial e final. Finalmente deve ser enviado o bloco de dados.
Código de acesso: Valor ASCII de W ou w.
- READ** - Sub-rotina usada para leitura das memórias do SAD. Do mesmo modo que na sub-rotina WRITE, após o acesso a esta sub-rotina, o dispositivo deve indicar o bloco de memória que deverá ser lido, enviando os endereços inicial e final. Em seguida o dispositivo deve estar pronto para receber os dados do SAD.
Código de acesso: Valor ASCII de R ou r.
- FILL** - Usada para escrever um determinado dado num bloco de memória do SAD. Da mesma forma que nas sub-rotinas anteriores, o dispositivo deverá indicar o tamanho do bloco de memória com os endereços inicial e final, enviando em seguida o dado a ser armazenado. A diferença desta sub-rotina para a WRITE, é que todo o bloco é preenchido com o mesmo dado.
Código de acesso: Valor ASCII de F ou f.

- EXECUTE** - Sub-rotina utilizada para executar um programa armazenado na memória de programa do SAD. No momento de acesso a esta sub-rotina, o programa objeto já deve estar carregado, o que pode ser feito com o uso da sub-rotina WRITE, e o segmento da memória definido. Para execução, o registro IP da CPU deve estar correto.
Código de acesso: Valor ASCII de G ou g.
- SETSEG** - Usada para mudar o segmento de memória de trabalho do SAD, o que deve ser feito de acordo com o sistema de endereçamento do processador 8088. Na inicialização é estabelecido um segmento, que depende da quantidade de memória disponível na RAM de programa do SAD.
Código de acesso: Valor ASCII de S ou s.
- SEND_REGS**- Sub-rotina que possibilita a leitura dos registradores da CPU do SAD. Quando acionada, o conteúdo dos 14 registros do 8088, armazenados em pseudo-registradores, são transmitidos. A seqüência de transmissão é: AX, BX, CX, DX, SP, BP, SI, DI, IP, CS, DS, SS, ES e FL.
Código de acesso: Valor ASCII de X ou x.
- SET_REGS** - Usada para alterar os registradores da CPU do SAD. Ao ser acionada, o conteúdo dos pseudo-registradores é alterado de acordo com os valores enviados. Os registradores propriamente ditos são alterados só quando se usa a sub-rotina EXECUTE, isto é, imediatamente antes da execução de um programa armazenado.
Código de acesso: Valor ASCII de Y ou y.
- SET_BANCO**- Sub-rotina usada para alterar o conteúdo do registrador usado na expansão da capacidade de endereçamento do 8088. Ao ser acionada, um dos 32 bancos de 64 Kbytes da memória de dados pode ser selecionado, enviando-se o número do banco desejado, que pode variar de 0 a 1Eh.
Código de acesso: Valor ASCII de B ou b.

- GRAVAR** — Usada para iniciar a aquisição de um sinal analógico na entrada do SAD. Ao ser acionada, as operações de conversão A/D ou D/A em curso são interrompidas, é realizada uma reconfiguração das funções analógicas, de acordo com os valores obtidos pela função CONF_SAD, e dá-se início a conversão A/D.
Código de acesso: Valor ASCII de T ou t.
- REPRO** — Sub-rotina que estabelece a operação de conversão D/A. Do mesmo modo que na sub-rotina GRAVAR, as conversões em andamento são interrompidas, é realizada a reconfiguração do sistema e a conversão D/A é iniciada.
Código de acesso: Valor ASCII de U ou u.
- CONF_SAD** — Usada para alterar o conteúdo da interface paralela 8255, que configura as funções do SAD. Após acessá-la, o computador externo deve enviar os novos valores de configuração desejados. Primeiro deve-se enviar o conteúdo do "port" A e, na seqüência, o conteúdo do "port" B.
Código de acesso: Valor ASCII de L ou l.

5.1.2 - Sub-rotinas de Uso Geral

As sub-rotinas principais são acessadas diretamente através da interface serial RS-232. No entanto, para realização das operações, são necessárias várias sub-rotinas auxiliares.

As sub-rotinas de uso geral executam diversas funções e dentre as várias implementadas pode-se destacar (ver ANEXO II):

- OUTPUT** : Usada para enviar dados pela interface RS-232;
- INPUT** : Recebe dados através da interface RS-232;
- SEARCH** : Busca endereço correspondente ao código usado para acessar uma determinada sub-rotina principal;
- GET_START** : Recebe o endereço (2 bytes) que define o início de um bloco para transferência entre SAD- COMPUTADOR;
- GET_END** : Recebe o endereço (2 bytes) que define o fim de um bloco para transferência entre SAD- COMPUTADOR;

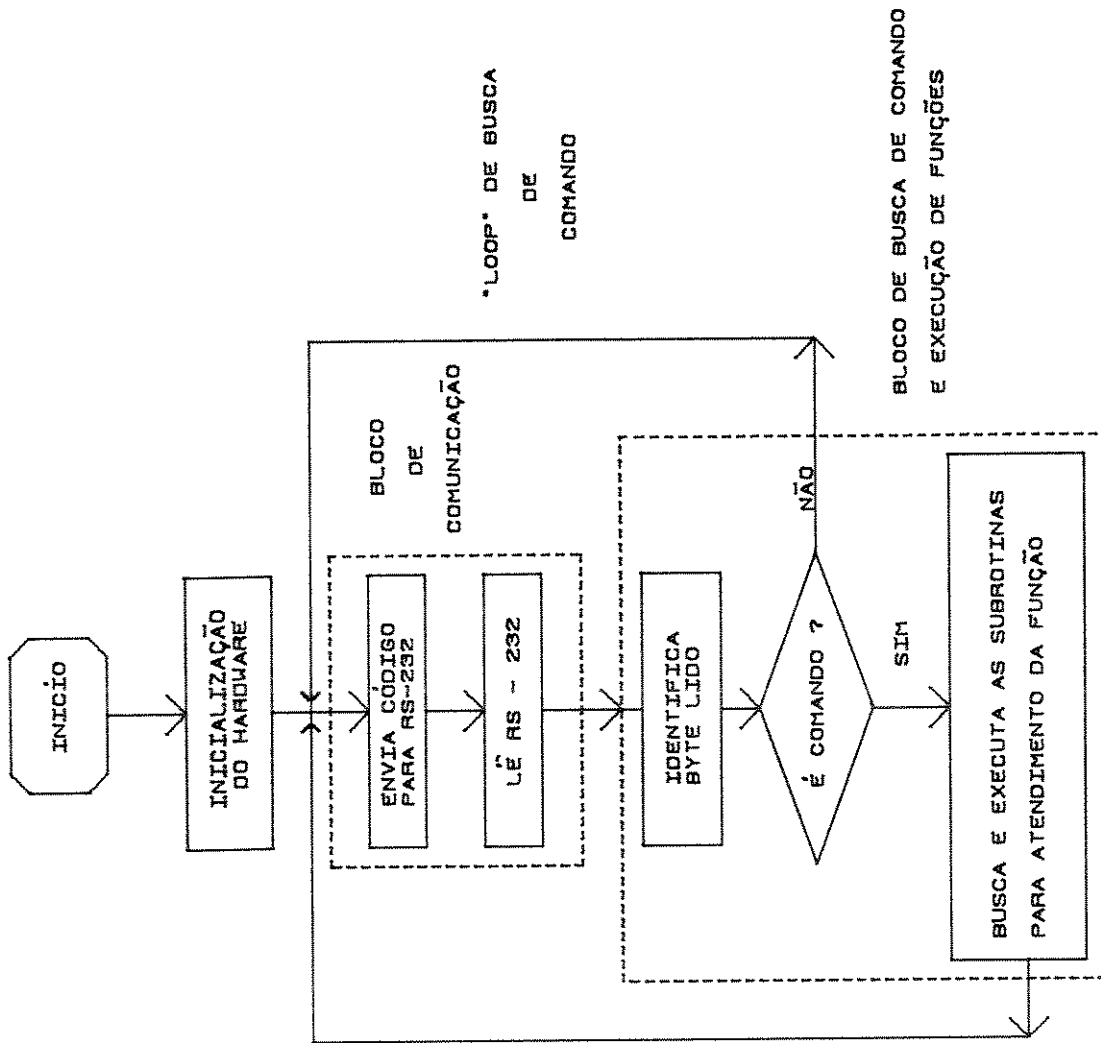


FIGURA 5.1 - ESTRUTURA DO "SOFTWARE" SBSAD

5.2 - O "SOFTWARE" DE COMUNICAÇÃO E OPERAÇÃO SCOMSAD

As operações realizadas no SAD se deram através de sua interligação com um microcomputador do tipo PC/XT, tornando-se necessário o desenvolvimento do "software" de comunicação e controle das suas funções no modo remoto, o SCOMSAD [19]. O programa foi elaborado em linguagem C, tomando-se por base todas as funções implementadas no programa monitor. O "software" SCOMSAD pode ser executado em qualquer microcomputador tipo IBM-PC.

O desenvolvimento do programa foi feito de tal forma que o PC/XT acoplado se transforma num terminal do SAD. As funções implementadas possibilitam as operações de conversão A/D e D/A, geração de arquivos binários no PC a partir dos dados digitalizados, transferências de arquivos do PC para o SAD, e vice-versa, e configuração das funções do SAD. Existe também uma série de comandos que permitem manipular o "hardware" do SAD, desde os registros da CPU até a memória de dados e de programa. O fluxograma da figura 5.2 (A) mostra como está estruturado o programa de um modo geral.

Na inicialização são definidas variáveis, matrizes e tabelas utilizadas de forma global. Na seqüência é estabelecida a comunicação com o SAD e o sistema entra no modo de operação. As funções de operação são divididas em dois blocos: Bloco de Controle das Funções do SAD e Bloco de Manipulação do "Hardware".

5.2.1 - Bloco de Controle das Funções do SAD

Neste bloco estão as funções que permitem a manipulação das operações de aquisição de dados. Na figura 5.2 (B) é apresentado um fluxograma que mostra a sua estruturação.

Inicialmente é definida uma configuração para a parte analógica do SAD. Os valores utilizados são baseados nos requisitos para aquisição de sinais de áudio. Para tal define-se: Filtros de 16 kHz, Taxa de Amostragem de 40 kHz, Acoplamento CC, Ganho de 0 dB, Saída de Áudio ativada para gravação e Conversão Linear de 12 bits.

Após estabelecer os valores iniciais para operação do SAD, é apresentada uma tela onde o usuário pode acompanhar ou alterar os valores previamente definidos. O usuário tem duas opções: sair do bloco de edição de tela e entrar no bloco de manipulação do "hardware" ou entrar na tabela de comandos que executam as funções de manipulação do

sistema. Se a opção for a primeira, o usuário pode manipular o "hardware" e inclusive retornar para a tela de operação, através de um comando denominado CONFIG. Se for a segunda, os parâmetros apresentados na tela são enviados ao SAD e o usuário pode executar as funções existentes. A seguir é dada uma descrição sumária de algumas funções implementadas, destacando-se também as sub-rotinas que são acionadas no programa monitor. A letra entre parênteses representa a tecla que o operador deve acionar para que o comando seja enviado ao SAD.

CONFIGURAR (C) - Função que permite modificar a configuração das unidades analógicas do SAD. Quando acionada, é apresentada uma tela com os valores da configuração atualizados, possibilitando também qualquer alteração. A comunicação com o SAD é feita através da sub-rotina CONF_SAD.

GRAVAR (G) - Função usada para iniciar aquisição de sinais analógicos. Quando acionada, é estabelecida a comunicação com o SAD através da sub-rotina GRAVA e são enviados os dados armazenados na etapa de configuração. Após isto é iniciada a conversão A/D.

REPRODUZIR (R) - Função que possibilita a reprodução de sinais digitalizados. Ao ser acionada é estabelecida a comunicação com o SAD através da sub-rotina REPRD. Do mesmo modo como na função GRAVAR, os dados da configuração são enviados. Na seqüência é dado início à conversão D/A.

CRIAR ARQUIVO (A) - Esta função cria um arquivo binário no PC com os dados gerados a partir da função gravar, que estão armazenados na memória de dados do SAD. Os arquivos são criados em blocos de 64 kbytes, podendo-se definir o número de blocos desejados e a posição do primeiro banco de 64 kbytes. A comunicação com o SAD é estabelecida através das sub-rotinas READ, que lê blocos de 64 kbytes e os transmite para o PC, e SET_BANCO, que comuta os bancos de 64 kbytes da memória de dados (para o caso de arquivos maiores que 64 kbytes). Ao acionar esta função, o usuário deve indicar o nome do arquivo a ser criado, a quantidade de bancos e o banco inicial.

CARREGAR ARQUIVO (E) - Função que transfere um arquivo binário do PC para o SAD. O arquivo deve estar no formato compatível com o padrão usado no SAD. Ao acioná-la o usuário precisa indicar apenas o nome do arquivo. Para auxiliar na formatação desenvolveu-se um programa que gera dados corretamente para o SAD a partir de um arquivo binário com dados de 16 bits. A comunicação é feita por meio das sub-rotinas WRITE e SET_BANCO, seguindo as mesmas operações descritas na função READ.

5.2.2 - Bloco de Manipulação do "Hardware" do SAD

Avaliações da operação do "hardware" do sistema, bem como o acompanhamento de algum programa que esteja sendo executado, podem ser realizadas através das funções de manipulação do "hardware". Este bloco funciona como mostrado na figura 5.2 (C).

O acesso ao bloco de manipulação ocorre quando, no bloco de controle, o comando "M" é executado. A indicação de que o "hardware" pode ser manipulado é dado por um "prompt" enviado pelo SAD (caractere *). A partir daí os comandos existentes podem ser acionados, bastando apenas digitá-los e finalizá-los com a tecla <ENTER>. Ao finalizar o comando, é feita uma busca numa tabela que contém os comandos possíveis. Independente da existência ou não do comando, no final é feito um retorno para o modo de edição de comandos. Se o comando for CONFIG o desvio ocorre para o bloco de controle, reiniciando todo o processo descrito no item anterior. As principais funções de manipulação do "hardware" implementadas são:

REGS - Com este comando é possível verificar e alterar o conteúdo dos registros da CPU do SAD. Para executá-lo basta digitar a palavra "REGS". Os 14 registros do 8088 são mostrados na tela e o usuário pode modificá-los. A finalização do comando acontece quando é digitada a tecla <ESCAPE>.

- LCOM** — Comando usado para carregar na memória do SAD um programa do tipo COM. Este comando é muito útil quando se quer adicionar funções ao programa monitor, pois possibilita um teste antes da implementação na EPROM do sistema. Para utilizá-lo é necessário digitar "LCOM arquivo.com". Com isto o programa é carregado e está pronto para ser executado.
- EXEC** — Executa programas armazenados na memória do SAD carregados com o comando LCOM. Ao utilizá-lo o usuário deve verificar o segmento de memória e o endereço inicial onde está armazenado o programa. Se houver qualquer erro, trava-se a comunicação, sendo necessária uma nova inicialização do programa SCOMSAD. Em alguns casos o computador acoplado também deve ser reinicializado por "hardware". A execução do programa armazenado é feita digitando-se "EXEC". Caso o programa não contenha a instrução HALT, no final da execução o "prompt" * indicará que a operação foi bem sucedida.
- EDIT** — Possibilita a visualização, no vídeo, de 256 bytes da memória do SAD de forma contínua, dentro do segmento especificado pelo usuário. Pode-se também fazer alterações nos valores editados e verificar outras locações de memória. Para utilizá-lo é necessário especificar o endereço inicial (em hexadecimal) da página de memória (64 kbytes). O formato geral do comando é "EDIT endereço inicial". O encerramento da edição ocorre quando se tecla <ESCAPE>.
- FILL** — Comando usado para preencher a memória do SAD com algum valor conhecido. O preenchimento da memória, no segmento definido previamente pelo usuário, é feito em bloco, que pode ser de até 64 kbytes. O valor a ser armazenado deve ser especificado em hexadecimal. Para utilizar o comando deve-se digitar "FILL end inicial end final valor".
- DISPLAY** — Permite a visualização, de forma contínua, de um bloco de até 64 kbytes da memória do SAD, dentro do segmento especificado. Os dados são apresentados em hexadecimal, podendo-se visualizar também o segmento e o endereço de memória. O comando deve ser fornecido no formato "DISPLAY end inicial end final", com os valores em hexadecimal.

SEG - Comando usado para alterar o segmento de trabalho da memória do SAD. É através deste comando que pode-se manipular as páginas de memória do 8088, pois o valor fornecido altera os registros CS, DS e CS do sistema. O formato do comando é "SEG segmento", com o valor segmento em hexadecimal. Por exemplo, ao se digitar SEG E000, seguido de <enter>, o sistema aponta para a memória de dados.

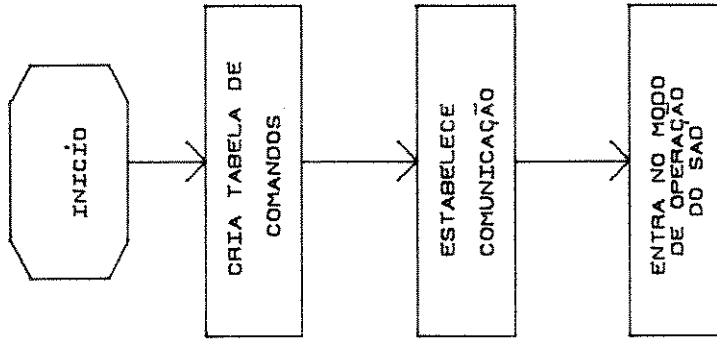


FIGURA 5 2 (A) - FLUXOGRAMA DO "SOFTWARE" SCOMSAD

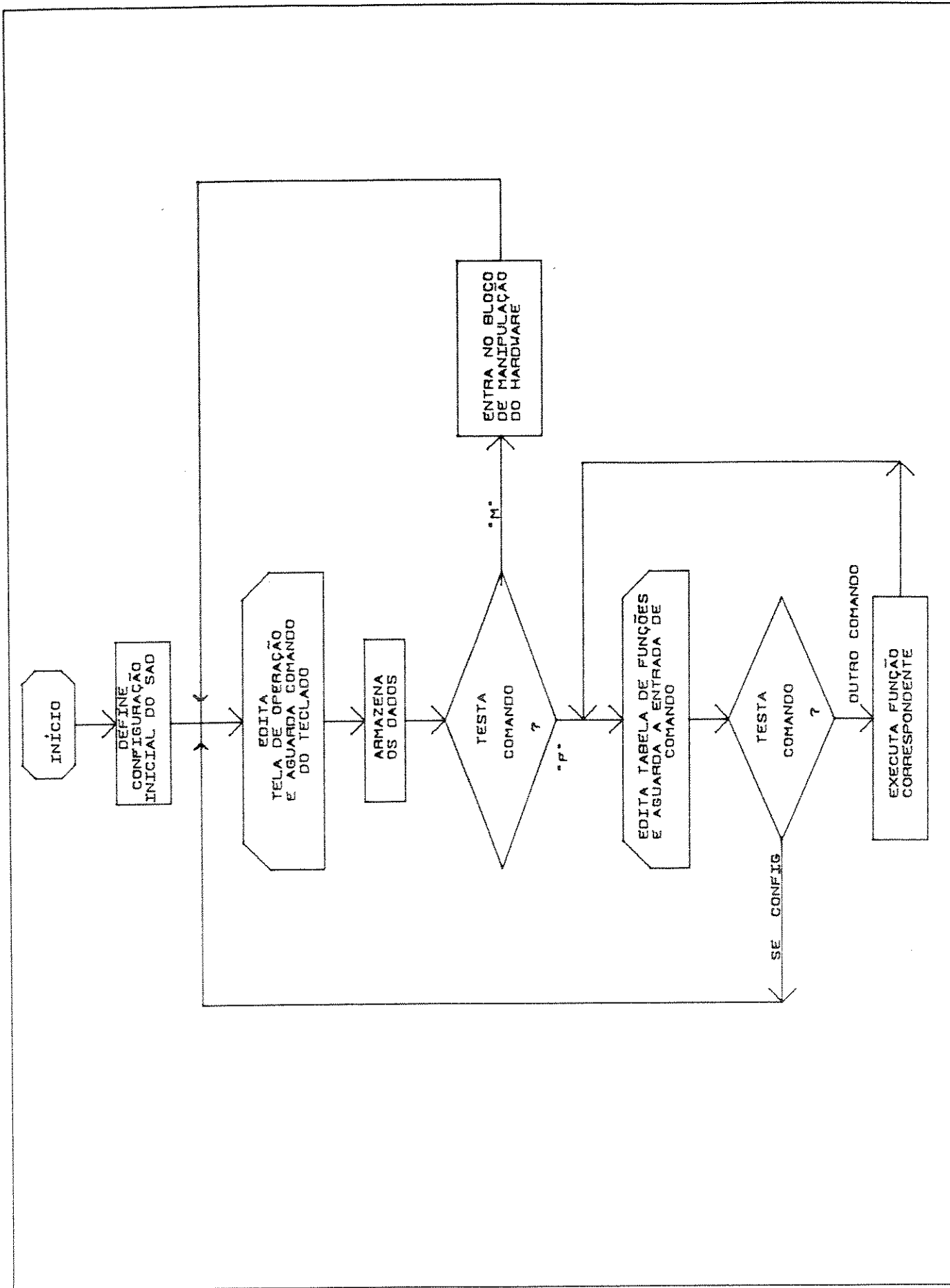


FIGURA 5.2 (B) - ESTRUTURA DO BLOCO DE CONTROLE DAS FUNÇÕES

CAPÍTULO 6 - CONSIDERAÇÕES FINAIS

Neste trabalho foi descrito o desenvolvimento de um sistema de aquisição de dados controlado por microprocessador (SAD), para operar com sinais na faixa de frequência de 0 a 50 kHz. O SAD, mostrado na foto da figura 6.1, está disponível no Laboratório de Processamento de Sinais do Departamento de Comunicações da FEE/UNICAMP, e está em operação desde julho de 1990. O SAD está operando em conjunto com um sistema gravador/reprodutor de sinais de áudio, como mostrado na figura 6.2.

As características das unidades de processamento do sinal analógico tornam o SAD uma ferramenta de grande importância para análise e desenvolvimento de algoritmos de processamento digital, registro de sinais e avaliação de resultados de processamentos digitais.

O SAD está em operação com o auxílio de um microcomputador externo, tipo IBM-PC (no modo remoto), pois ainda não foi desenvolvido o "software" de operação da unidade teclado/display. O microcomputador também se faz necessário porque é através dele que o usuário tem acesso aos dados digitalizados.

A comunicação do SAD com o microcomputador acoplado está sendo feita através da interface serial RS-232, com taxa de 9600 bits/s. Com esta taxa, a transferência de arquivos longos entre SAD-Computador acaba sendo uma tarefa tediosa. Por exemplo, para transferir um arquivo de 1 Mbytes são necessários 15 minutos. Para possibilitar maiores taxas de comunicação deve-se usar a interface GPIB. No entanto, ainda não foram feitos os testes finais para colocar esta interface em operação no SAD.

6.1 - ASPECTOS PRÁTICOS DE IMPLEMENTAÇÃO

A implementação do protótipo foi feita, em sua maior parte, utilizando-se montagens em "wire-wrapping" para interligar os circuitos digitais. Esta técnica tem como principal vantagem a rapidez com que podem ser feitas as ligações, o que viabilizou a construção, relativamente rápida, do SAD. Por outro lado, montagens em "wire-wrapping"

tornam os circuitos altamente sensíveis à ruídos, o que termina exigindo uma série de cuidados. Com isto, e sabendo-se ainda que o SAD é um conjunto de certa forma complexo de circuitos digitais e analógicos, tomaram-se os seguintes cuidados na construção do sistema:

- Isolação máxima, tanto física quanto elétrica, entre os circuitos analógicos e digitais, que inclui blindagens de circuitos analógicos, sistemas de aterramento independentes, etc.;
- Uso intensivo de dispositivos de desacoplamento;
- Grades de aterramento para as placas com maior densidade de componentes, como as de memória e DMA;
- Minimização das distâncias entre os CI's digitais, de modo a melhorar o desempenho da técnica de "wire-wrapping".

6.2 - RESULTADOS PRÁTICOS

Para se ter uma idéia da operação real do SAD digitalizou-se, a uma taxa de 8 kHz, uma senóide de 3 kHz (figura 6.3 (A)) gerada por um oscilador comum. A senóide reproduzida no SAD é apresentada na figura 6.3 (B). As figuras foram obtidas por meio de um osciloscópio digital acoplado através de uma interface GPIB a um microcomputador tipo IBM-PC, o que justifica as pequenas irregularidades presente.

Alguns trabalhos de pesquisa já estão sendo desenvolvidos com o auxílio do SAD, destacando-se:

- 1 - Estudos de Técnicas Digitais para Redução de Ruído em Gravações de Áudio, assunto de tese de mestrado de Kenia Pena de Sousa. A figura 6.4 apresenta um exemplo de aplicação dos estudos desenvolvidos por Kenia. Na figura 6.4 (A) tem-se um trecho de música instrumental com ruído impulsivo e na figura

6.4 (B) tem-se o mesmo trecho de música após aplicação das técnicas desenvolvidas.

- 2 - Estudos em Síntese de Voz para a Língua Portuguesa, que está tendo início com o desenvolvimento da tese de doutorado de Edson José Nagle. Alguns resultados dos estudos desenvolvidos por Edson Nagle são apresentados nas figuras 6.5 e 6.6. Na figura 6.5 (A) é mostrada a palavra SACI falada naturalmente e na figura 6.5 (B) a palavra SACI sintetizada. Na figura 6.6 tem-se o detalhe da vogal A.

6.3 - MELHORAMENTOS FUTUROS NO SAD

Uma análise detalhada do SAD mostra que existem alguns pontos que podem ser alterados, de modo a torná-lo mais eficiente. Os principais são:

- Número de entradas e saídas: O SAD possui uma única entrada e uma única saída analógica. Aumentando o número de entradas e saídas, o sistema poderia ser utilizado em diversas aplicações de controle de processos [2], como controle de fluxo, velocidade, temperatura, etc.,
- Faixa de Operação: O limite de frequência de 50 kHz é dado principalmente pelo conversor A/D. O uso de um conversor mais rápido pode aumentar de modo direto a faixa de operação para até 70 kHz, sem alterar a resolução.
- Taxa de Transferência de Dados: O limite de taxa de transferência de dados nas conversões, de 220 Kbytes/s, é dado pelo controlador de DMA e pelo processador 8088. Estes dispositivos estão operando com um relógio de 5 MHz. A mudança do controlador de DMA e um aumento na frequência do relógio do 8088 (até 8 MHz, na versão utilizada no SAD) permite que se trabalhe com taxas mais elevadas. Vale salientar que no caso do protótipo desenvolvido, qualquer aumento na frequência do relógio do sistema deve ser cuidadosamente

estudado, devido à utilização da técnica de "wire-wrapping" nas montagens.

- Capacidade de Processamento: As aplicações de sistemas de aquisição e controle em tempo real têm crescido significativamente nos últimos anos. O desenvolvimento destes sistemas passam, na maioria dos casos, por várias etapas de testes e simulações. Visando auxiliar estas simulações e testes, pode-se acoplar ao SAD um processador digital de sinais de alta velocidade, como, por exemplo, um TMS320C25 [5]. Esta possibilidade já está sendo estudada e deve ser em breve implementada.

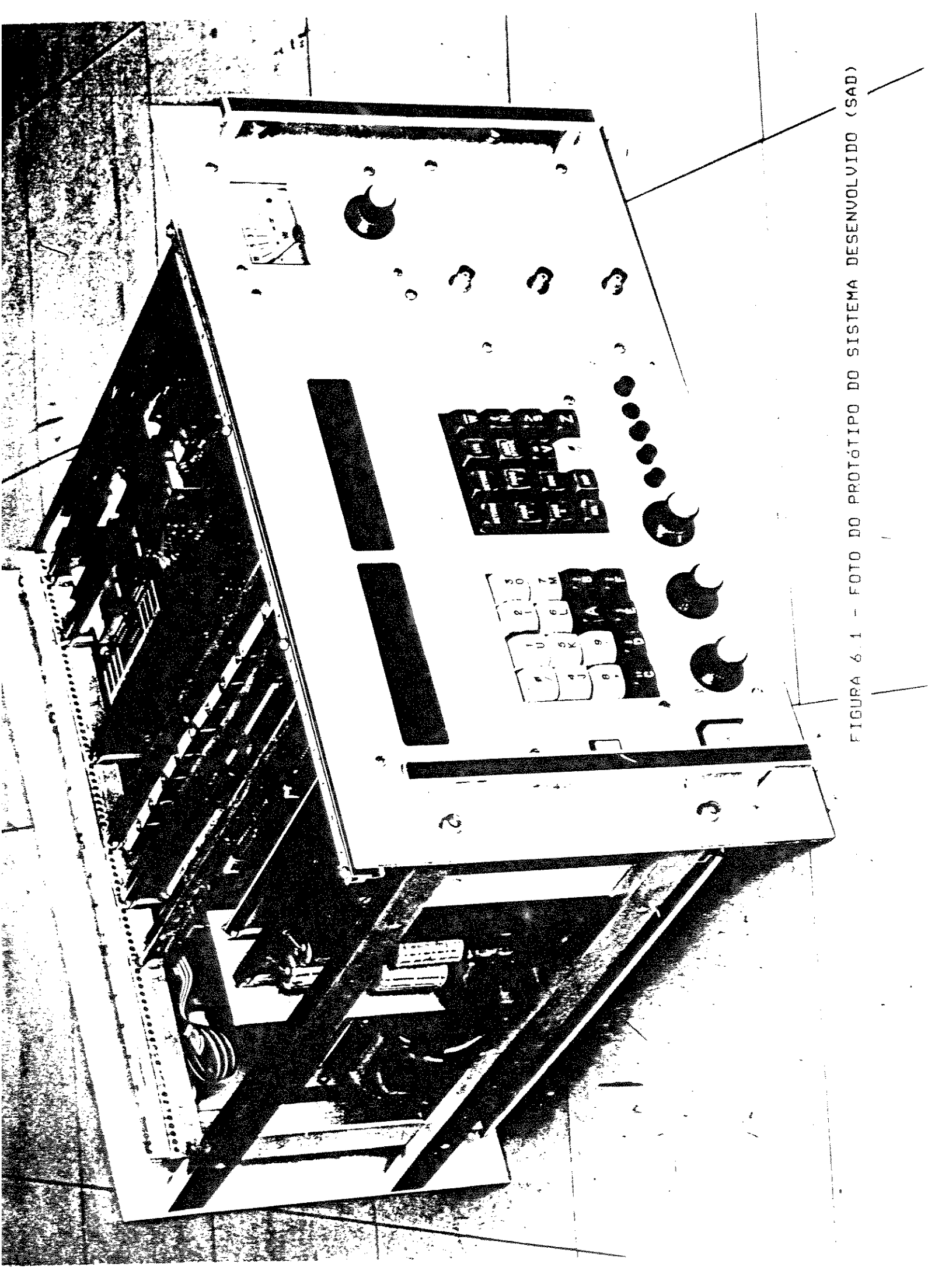


FIGURA 6.1 - FOTO DO PROTÓTIPO DO SISTEMA DESENVOLVIDO (SAD)

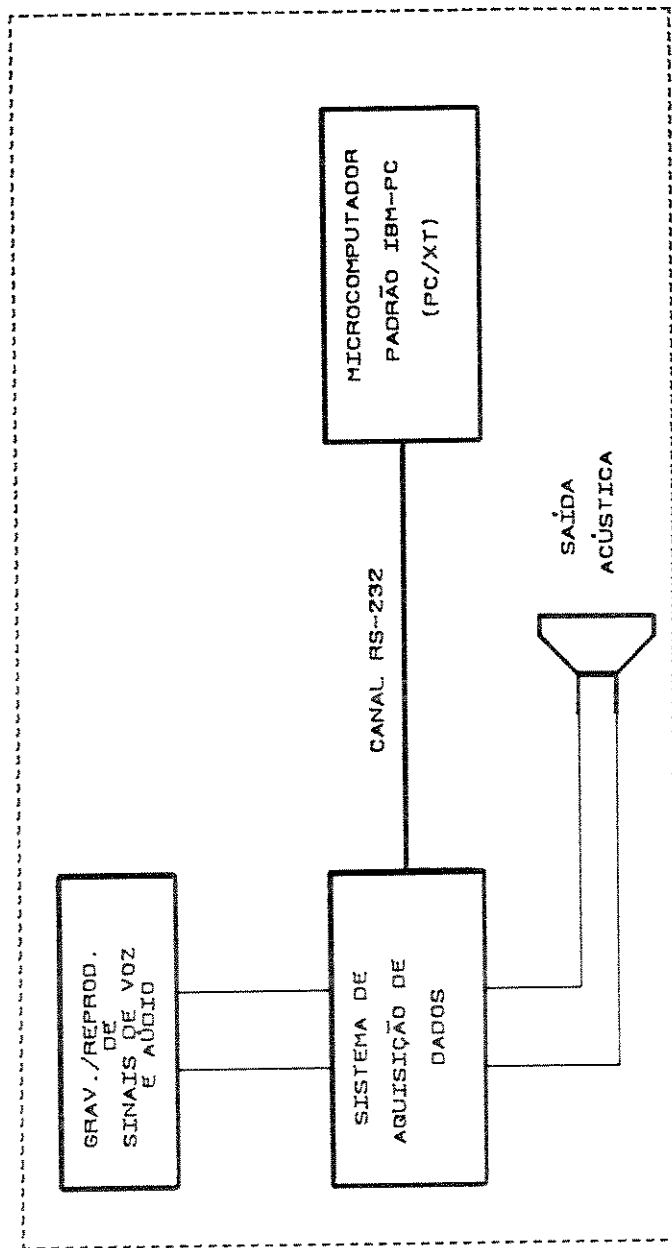


FIGURA 6 2 - DIAGRAMA DE BLOCOS DO SAD PARA APLICAÇÃO EM SINAIS DE VOZ E ÁUDIO

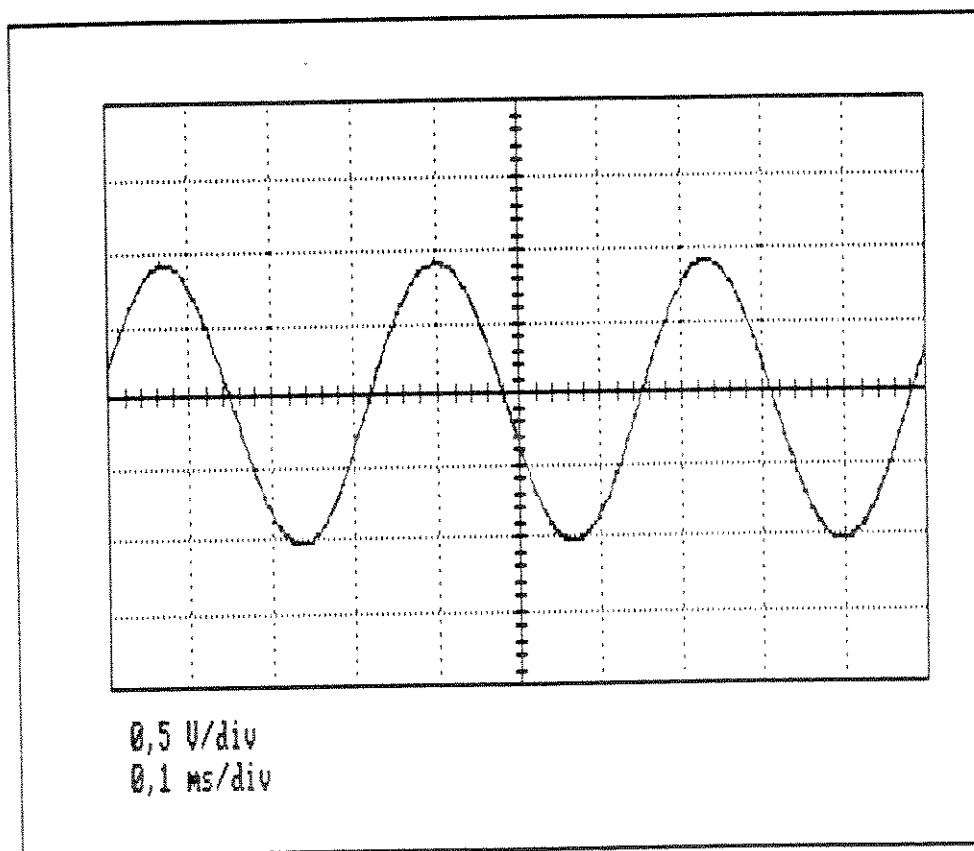
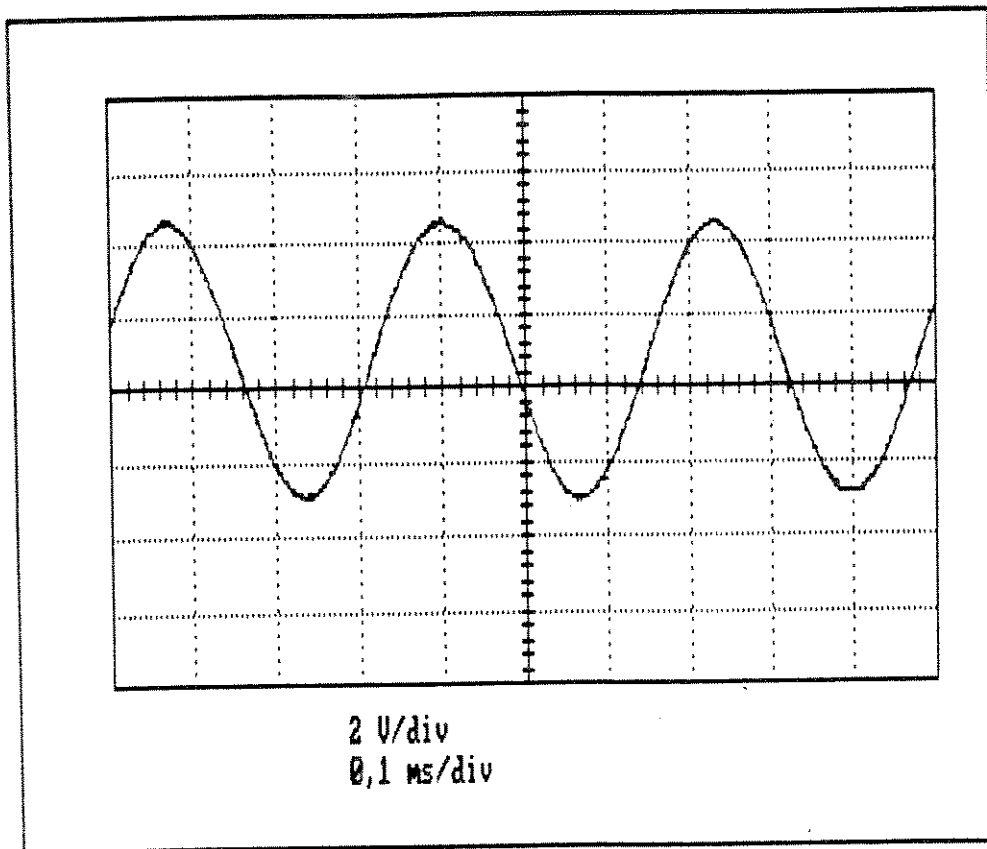


FIGURA 6.3 - (A) SENÓIDE COM FREQUÊNCIA DE 3 kHz GERADA POR OSCILADOR
(B) SENÓIDE AMOSTRADA A 8 kHz E REPRODUZIDA NO SAD

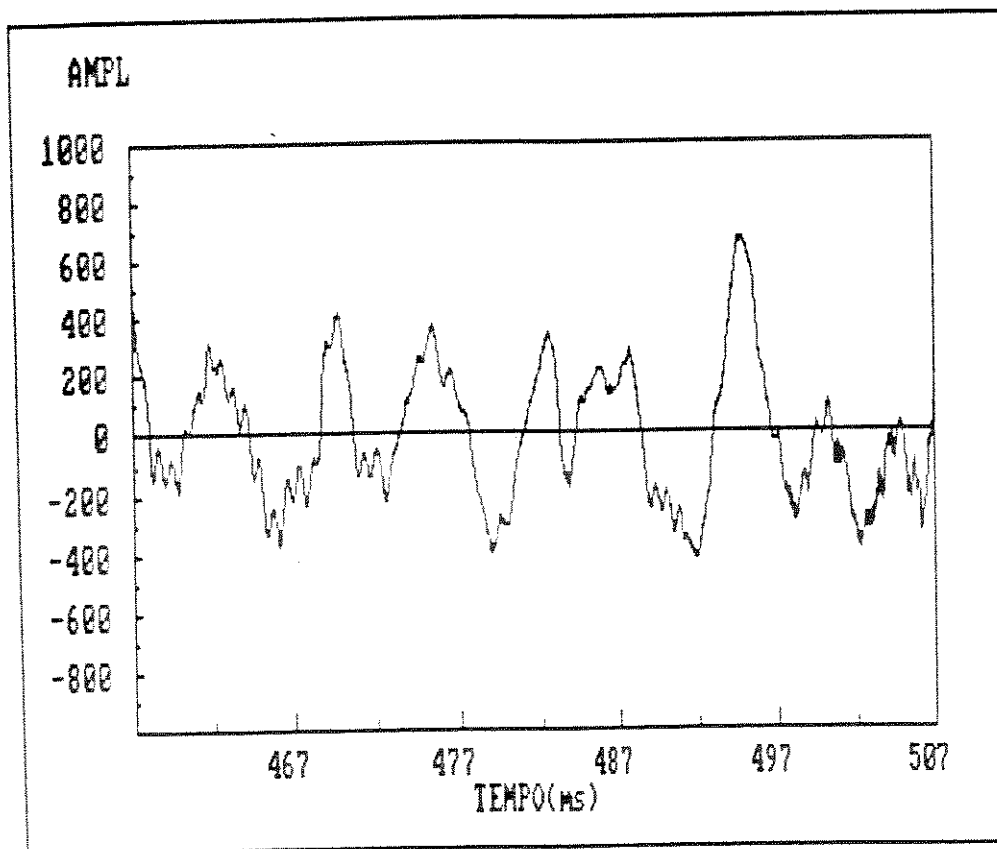
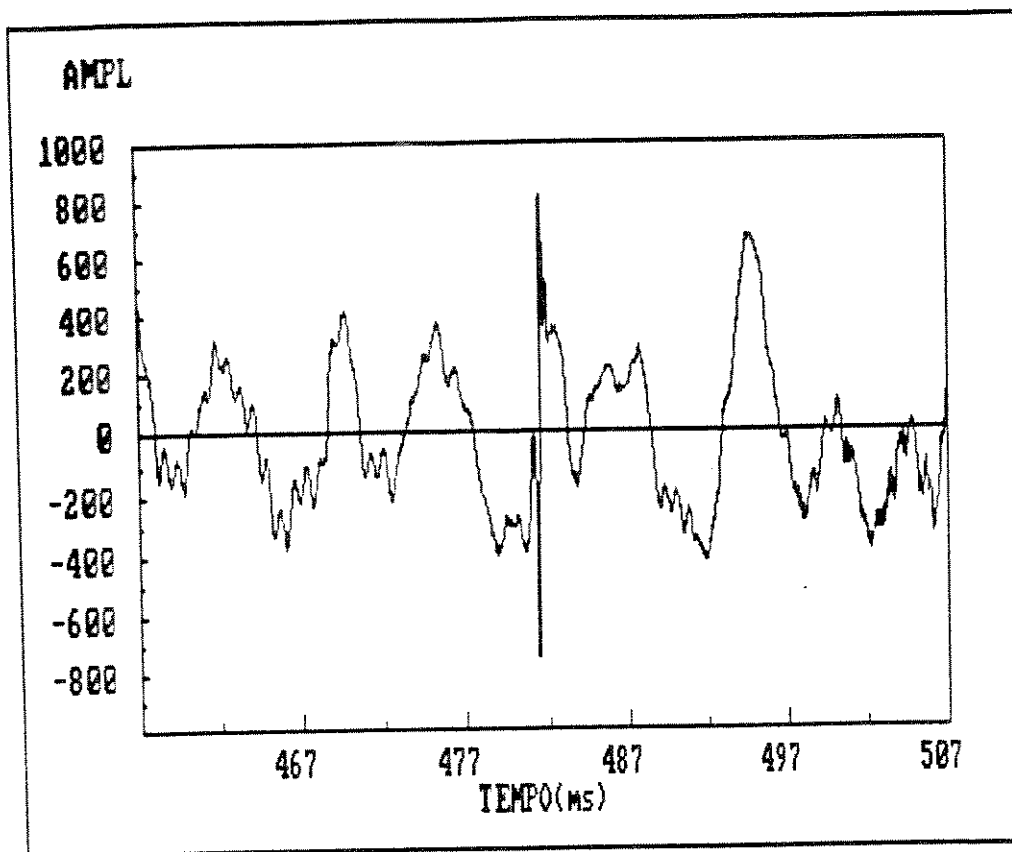


FIGURA 6.4 - MÚSICA INSTRUMENTAL COM RUÍDO IMPULSIVO (a) ANTES DO PROCESSAMENTO (b) APÓS O PROCESSAMENTO

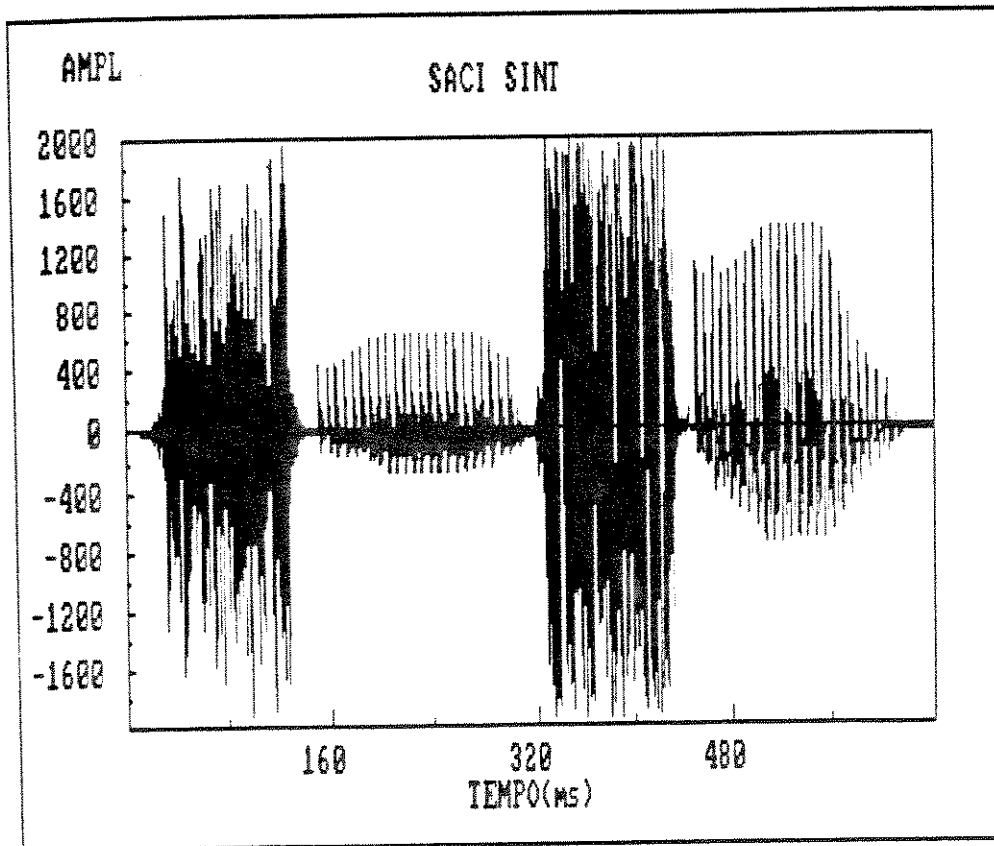
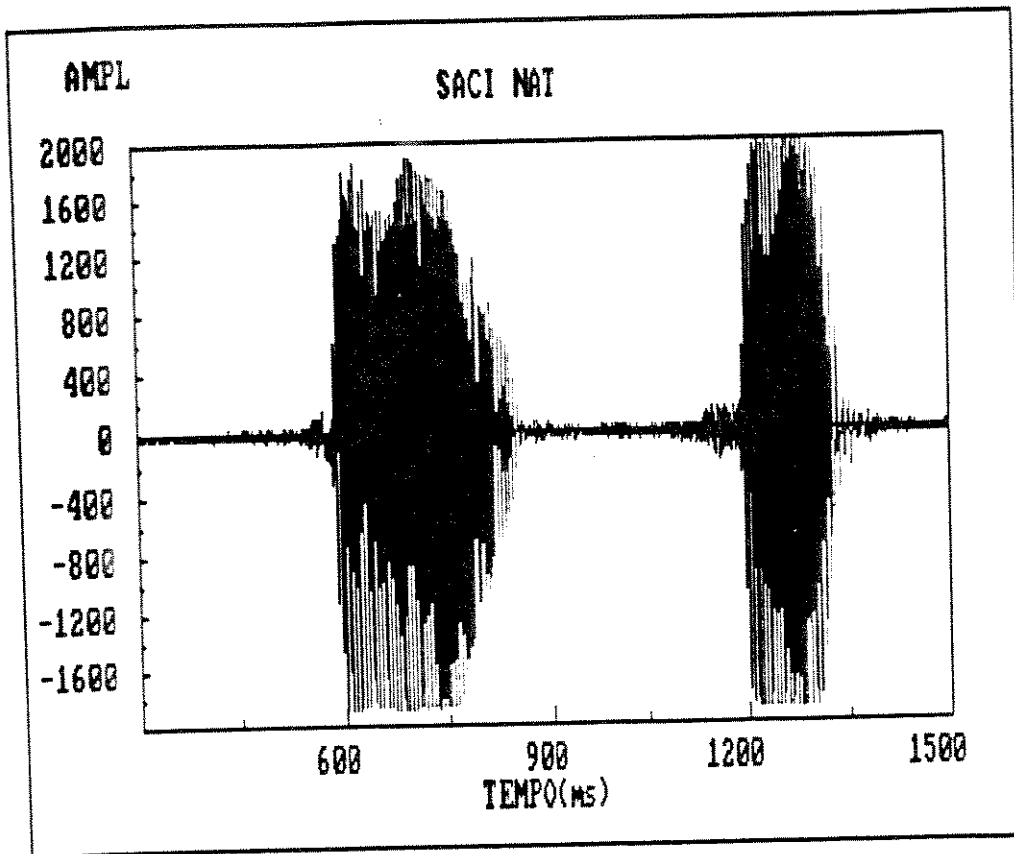


FIGURA 6.5 - (a) PALAVRA saci NATURAL (b) SINTETIZADA

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] - Analog Devices; "Analog-Digital Conversion Handbook", 3ª edição Prentice-Hall (1986).
- [2] - Seippel, R.G.; "Transducer Interfacing - Signal Conditioning for Process Control", Prentice-Hall (1988)
- [3] - Hall, D.W.; "Microprocessors and Digital Systems", 2ª edição McGraw-Hill (1984).
- [4] - Tompkins, W.J., Webster J.G.; "Interfacing Sensors to the IBM-PC", Prentice-Hall (1986).
- [5] - Texas Instruments "TMS320C25 - DSP Design Workshop", Student Guide, (1990)
- [6] - Nascimento, R. L., Jaccoud, C. I. B.; "Sistema de Conversão A/D e D/A Linear de 12 bits para Sinais de Voz e Áudio", - Anais do 6º Simpósio Brasileiro de Telecomunicações, Campina Grande-Pb (1988).
- [7]- Keithley Instruments; "Eletronic Instruments Catalog and Buyer's Guide", (1987/1988).
- [8] - Violaro, F.; "Sistema de Análise e Processamento Digital de Voz SAPDV", Anais do 5º Simpósio Brasileiro de Telecomunicações, Campinas-SP (1987)
- [9] - Rockland Scientific; "Signal Analysis Workstation", Product Information (USA-1989).
- [10]- Key Elemetric, "DSP Sona Graph Workstation", Product Information (1988)

- [11]- Nichols, E.A., Nichols, J.C., Musson, K.R.; "Data Communication for Microcomputers", McGraw-Hill (1982).
- [12]- Violaro, F.; "Nova versão do Sistema de Análise e Processamento Digital de Voz: SAPDV-A", Anais do 7º Simpósio Brasileiro de Telecomunicações, Florianópolis-SC (1989)
- [13]- TEKTRONIX; Manual Técnico (1989)
- [14]- KIKOSUI; Eletronic Test and Measurement Instrument - Power Supply, General Catalog No. KIK-3 (Japan- 1990).
- [15]- Hewlett-Packard; "Dynamic Signal Analyzer", Data Subject (1988).
- [16]- Hewlett Packard; "HP-6944 e HP-6942A, Multiprogrammer Systems for Computer Aided Test", Ordering and Configuration Guide (1988)
- [17]- SPECTRUM SIGNAL PROCESSING; Product Catalog (Canadá-1990).
- [18]- Temes, G., Lapatra, J. W.; "Circuits Synthesis and Design", McGraw-Hill (1977).
- [19]- Chiquito, J.G., Vieira, J., Bortolli, E., Oliveira, P.; "Processamento Digital de Sinais e Filtragem Adaptativa de Sinais de Baixa Freqüência", Relatório Técnico Nº 88/0220-7, FAPESP (1989).

APÊNDICE A

AVALIAÇÃO DAS TÉCNICAS DE TRANSFERÊNCIA DE DADOS

1 - Transferências com controle da CPU

Para avaliar os tempos envolvidos nas transferências por controle da CPU, fez-se uma análise simples de um programa para transferir um bloco de 64 Kbytes, gerado a partir de uma conversão A/D. A avaliação é feita com base no tempo envolvido em cada instrução necessária no programa. Para tal considera-se que o sistema utilizado é baseado no processador 8088.

A - "Polling"

A técnica de "polling" consiste de leituras periódicas no dispositivo que deve enviar dados, com a CPU sempre pronta para fazer transferências. Nos sistemas de aquisição de dados a CPU lê constantemente um bit indicativo de geração de dados, fazendo a transferência quando a indicação é positiva. O programa a seguir implementa de forma simplificada a técnica de "polling".

PROGRAMA

INSTRUÇÕES	COMENTÁRIOS	TEMPO (em Nº de Ciclos)
MOV BX,0000H	Inicia Conversão A/D e	04
MOV AL,01H	estabelece o endereço	04
OUT INICIO_AD,AL	inicial da memória.	10
LOOP: IN AL,CONTR_AD	Testa	10
AND AL,01H	se existe	04
JNZ LOOP	dado disponível.	15
IN AL,DATA-AD	Lê o dado e	10
MOV [BX],AL	armazena.	10
CMP BX,0FFFFH	Testa fim do bloco e em	04

JZ	FIM	caso positivo finaliza.	15
INC	BX	Caso contrário retorna	02
JMP	LOOP	para ler outro dado	15

FIM: (Programa de Encerramento)

TOTAL DE CICLOS ENVOLVIDOS: 103

Um ponto crítico do programa é o instante de verificação do bit de controle de geração de dados (execução do LOOP). Isto porque no momento da leitura do bit o dado pode ainda não estar disponível. Se ele ficar disponível instantes depois, deve-se prever a execução das instruções AND e JNZ no LOOP para a leitura efetiva do dado. Considerando esta situação, é preciso adicionar um tempo de atraso de 19 ciclos. Com isto, o tempo mínimo entre uma transferência e a próxima é de 122 ciclos de relógio.

Para um sistema trabalhando com um relógio de 5 MHz, o tempo envolvido numa transferência de dados seria de 24,4 microssegundos. Se o sistema usa um conversor A/D de 8 bits, ele pode fazer aquisições a uma taxa máxima de, aproximadamente, 41 Kamostras/s. Para um conversor de 12 bits esta taxa seria menor, devido à necessidade de se acrescentar mais instruções de leitura e escrita.

B - Interrupção

A técnica de interrupção possibilita a transferência de dados sem que a CPU tenha que vistoriar periodicamente o dispositivo que gera os dados. Neste caso, o dispositivo solicita serviço à CPU sempre que estiver com um dado para ser transferido. Um pedido de interrupção pode ser gerado em qualquer instante, independentemente das funções que estejam sendo executadas pelo processador.

O programa usado para avaliar a técnica de "polling" serve como base para avaliação da técnica de interrupção. As instruções a seguir são suficientes para executar uma transferência de dados.

ROTINA DE SERVIÇO

INSTRUÇÕES	COMENTÁRIOS	TEMPO (em Nº de ciclos)
IN AL,DATA_AD	; Lê dado disponível	10
MOV BX,END_REG	; Carrega o valor do endereço	04
MOV [BX],AL	; Armazena o dado	10
MOV END_REG,BX	; Salva endereço para próxima transf.	18
CMP BX,0FFFFH	; Testa se é fim de bloco	04
INC BX	; Incrementa endereço	02
JZ FIM	; Se terminou, desvia para Fim	15
IRET	; Caso contrário retorna da int	32

FIM: (Zera contador de endereço e retorna da interrupção)

Considerando apenas a rotina de serviço, seriam necessários 95 ciclos de relógio para efetuar a transferência de um dado. Além da rotina de serviço, deve-se levar ainda em consideração o tempo envolvido para atendimento da interrupção. No 8088, são necessários 51 ciclos de relógio para atendimento de uma interrupção, a partir do instante em que foi reconhecida. Sem considerar outros atrasos, que podem ocorrer principalmente nos sistemas que possuem vários níveis de interrupção, seriam necessários 146 ciclos de relógio para se transferir um dado.

Para o número de ciclos calculados anteriormente, e supondo a CPU operando com um relógio de 5 MHz, a máxima taxa de transferências que se pode obter, supondo uma conversão 8 bits, é de aproximadamente 34 kamostras/s.

2 - Transferência por DMA

A avaliação de uma transferência por acesso direto à memória (DMA) é feita considerando-se um sistema de aquisição baseado num microcomputador do tipo PC/XT, que utiliza como controlador de DMA o CI 8237-2 e como processador o CI 8088, ambos fabricados pela INTEL.

Na figura A.1 tem-se um diagrama simplificado das operações de DMA do controlador 8237-2 e das operações de "HOLD" do microprocessador

8088. As considerações abaixo supõem que o controlador é programado para fazer transferências no modo único (ver item 4.2.1, capítulo 4) e opera com dois canais.

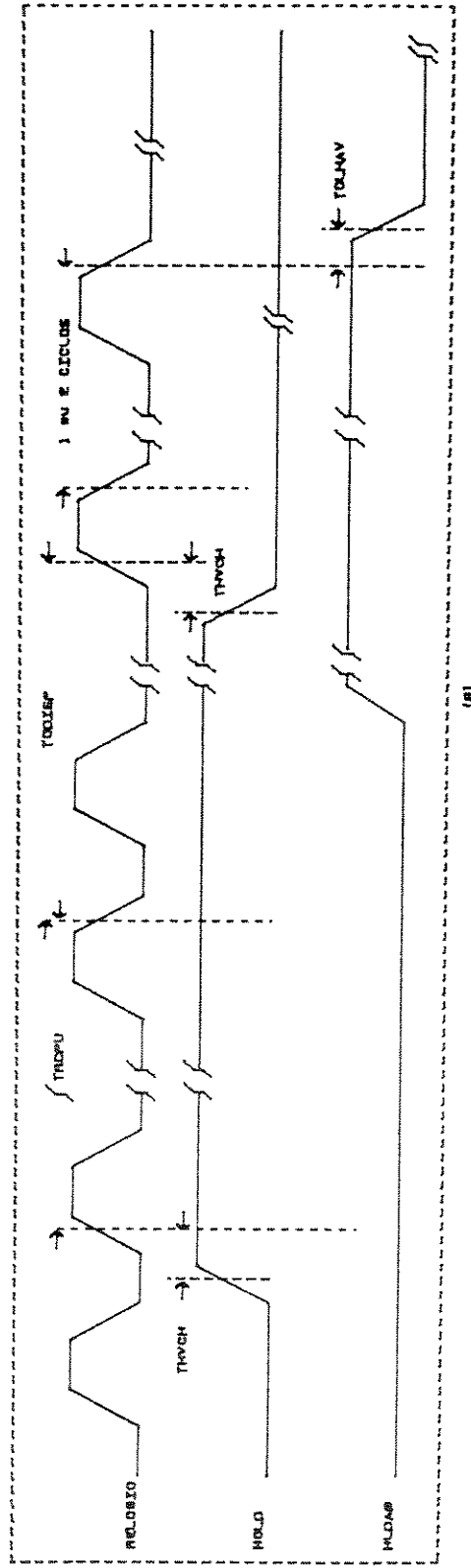
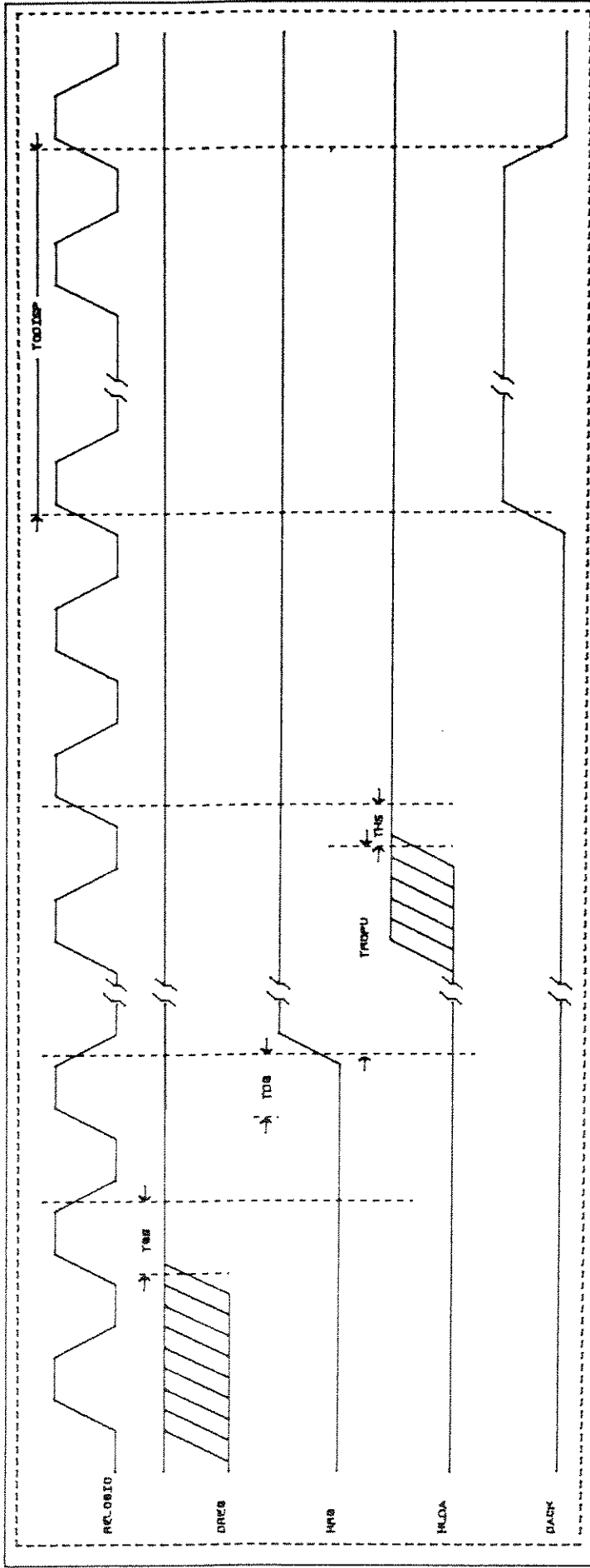
Situação Crítica: Um dispositivo solicita serviço de DMA no canal 2 instantes depois de um outro ter solicitado serviço no canal 1.

Condição de Resposta:

- 1 - O controlador 8237-2 irá completar o serviço do primeiro pedido. O tempo envolvido nesta operação é de 11 ciclos de relógio (ver figura A.1)
- 2 - Após completar o serviço referente ao canal 1, o controlador 8237-2 ainda levará 5 ciclos de relógio para liberar o sinal DACK.
- 3 - Finalmente, o controlador 8237-2 irá levar 6 ciclos de relógio para fazer a transferência solicitada.

O resultado desta operação crítica mostra que, quando dois canais de DMA são usados concorrentemente, só se tem garantido um determinado pedido após um tempo de aproximadamente 22 ciclos de relógio.

Para um sistema operando com um relógio de 5 MHz, o tempo mínimo necessário entre dois pedidos de DMA deve ser 4,4 microssegundos. A taxa de transferências máxima garantida, no caso de se utilizar um conversor de 8 bits é de, aproximadamente, 220 Kamostras/s.



TROPU - TEMPO DE RESPOSTA DA DRU (medida de 2 Ciclos de Relógio)
 TOSLSP - TEMPO DE ATRASO PARA HRS VALIDO (máximo de 100 ns)
 TOS - TEMPO DE ATRASO PARA HRS VALIDO (máximo de 100 ns)
 THS - TEMPO DE ATRASO PARA HRS VALIDO (máximo de 35 ns)
 TROCH - TEMPO DE ATRASO PARA H.LDA VALIDO (máximo de 180 ns)
 TOLHAY - TEMPO DE ATRASO PARA H.LDA VALIDO (máximo de 180 ns)

FIGURA A.1

APÊNDICE B

```

;*****
;
;      LABORATORIO DE PROCESSAMENTO DE SINAIS  DECOM/FEE/UNICAMP
;      Programa monitor para o SAD - SBSAD (Processador 8088)
;      Campinas, abril de 1990
;
;*****
;*****
;      Enderecos de perifericos
;*****

SIODT          EQU 00H          ;DADOS RS-232C (8251)
SIOCONT        EQU 01H          ;CONTROLE RS-232C

INTL           EQU 10H          ;CONT. INTERRUPTOES (ENDEREÇO BAIXO)
INTH           EQU 11H          ;      "      "      (ENDEREÇO ALTO)

TM0            EQU 08H          ;TIMER #0 (8253-Unidade Processador)
TM1            EQU 09H          ;TIMER #1
TM2            EQU 0AH          ;TIMER #2
TMCONT        EQU 0BH          ;CONTROLE DOS TIMERS

DMACOM         EQU 48H          ;CONTROLADOR DE DMA
DMAMOD         EQU 4BH
DMAREG         EQU 49H
DMASKS         EQU 4AH

DMASKW         EQU 4FH
DMATMP         EQU 4DH
DCLPFF         EQU 4CH
DMAMCL         EQU 4DH

BASE0          EQU 40H
CURR0          EQU 40H
COUNT0        EQU 41H
CURCT0         EQU 41H

BASE1          EQU 42H
CURR1          EQU 42H
COUNT1        EQU 43H
CURCT1         EQU 43H

PORT_CONTR     EQU 0A3H          ;INTERFACE PARALELA 8255 PARA
PORTA          EQU 0A0H          ;CONTROLE DAS CONVERSOES
PORTB          EQU 0A1H          ;A/D e D/A
PORTC          EQU 0A2H

CONTE0         EQU 50H          ;CONTADORES DE ENDEREÇOS DA UNIDADE
CONTE1         EQU 51H          ;DE CONTROLE DE DE DMA
CONTE2         EQU 52H

CONTP0         EQU 54H          ;CONTADORES DE PALAVRAS DA UNIDADE
CONTP1         EQU 55H          ;DE CONTROLE DE DMA

```

```

CONTP2          EQU 56H

TIMER_CONTR     EQU 0A7H          ;TIMER 8253 QUE GERA TAXAS
TIMER0          EQU 0A4H          ;DE AMOSTRAGEM PARA O SAI
TIMER1          EQU 0A5H          ;(CONV. A/D e D/A)
TIMER2          EQU 0A6H

REGBANCO        EQU 1CH          ;ENDERECO DO REGISTRADOR DE BANCO
                                   ;DE 64K (HA 32 NA PLACA DE 2 MB)

;*****
;      Vetores de interrupcao
;*****

UWARM_START     EQU 00H          ;REENTRADA NO MONITOR
VTRAP           EQU 01H          ;EXECUÇÃO PASSO A PASSO

;*****
;      Constantes
;*****

SIZE_STACK      EQU 128          ;TAMANHO DA PILHA

;*****
;      Corpo do programa
;*****

SEGMENTOS SEGMENT

ASSUME CS:SEGMENTOS, DS:SEGMENTOS, SS:SEGMENTOS, ES:SEGMENTOS

ORG             0100H

START:          CLI              ;INICIALIZACAO
                MOV AX,CS
                MOV DS,AX
                MOV SS,AX
                MOV ES,AX
                MOV SP,(OFFSET TDF_STACK)+1
                MOV AL,07H
                MOV PORTA_REG,AL
                MOV AL,01H
                MOV PORTB_REG,AL
                CALL INIT_HARD
                MOV AX, OFFSET RAM
                ADD AX,15
                AND AX,0FFF0H
                MOV WORK_SEG,AX
                MOV CL,4
                SHR WORK_SEG,CL
                MOV AX,CS
                ADD WORK_SEG,AX

```

```

MOV AX,WORK_SEG      ;INICIALIZA PSEUDO REGISTRADORES
MOV CS_REG,AX
MOV DS_REG,AX
MOV ES_REG,AX
MOV AX,SS
MOV SS_REG,AX
MOV SP_REG,SP
MOV FL_REG,0F02AH
MOV IF_REG,0000H
MOV AX,0000H
MOV AX_REG,AX
MOV BX_REG,AX
MOV CX_REG,AX
MOV DX_REG,AX
MOV BF_REG,AX
MOV SI_REG,AX
MOV DI_REG,AX
MOV BANCO_REG,AX
MOV AL,55H
MOV DMA_REG,AL
MOV DMA_AD_REG,AL
MOV AL,59H
MOV DMA_DA_REG,AL
MOV AL,0FH
MOV TAXA_REG,AL
MOV AL,0FFH
MOV CONTF0_REG,AL
MOV CONTF1_REG,AL
MOV AL,1FH
MOV CONTF2_REG,AL

```

```

LOOP1:
MOV CONST_VER,16      ;CONSTANTE VERTICAL DE EDICAO
MOV AL,'*'            ;PRDMPT PARA RS-232C
CALL OUTPUT
CALL INPUT             ;LEIA ENTRADA
AND AL,11011111B     ;TRANSFORME EM MAIUSCULA
MOV BX,OFFSET TAB_COM ;BUSCA NA TABELA DE COMANDOS
CALL SEARCH
OR BX,BX
JZ LOOP1              ;NADA FAZ SE O COMANDO NAO EXISTIR
CALL BX               ;EXECUCAO DO COMANDO
JMP LOOP1

```



```

;*****
;      'T' ou 't'
;      Inicia conversao Analogica-Digital do SAD
;*****

```

```

GRAVAR:      CALL GERAR_TAXA
              MOV AL,DMA_AD_REG
              MOV DMA_REG,AL
              CALL DMA_INICIO
              MOV AL,PORTA_REG
              OR AL,20H           ;ATIVA SAIDA DE AUDIO
              OUT PORTA,AL
              MOV AL,PORTB_REG
              AND AL,0DBH       ;ZERA CONVERSAD A/D e D/A
              OUT PORTB,AL
              PUSH AX
              CALL CONTADORES
              POP AX
              OR AL,04H
              OUT PORTB,AL      ;INICIA CONVERSAD A/D
              RET

```

```

;*****
;      'U ou 'u"
;      Inicia conversao Digital-Analogica do SAD
;*****

```

```

REPRODUZIR:  CALL GERAR_TAXA
              MOV AL,PORTA_REG
              OR AL,30H         ;ATIVA SAIDA DE AUDIO
              OUT PORTA,AL
              MOV AL,PORTB_REG
              AND AL,0DBH       ;ZERA CONVERSAD A/D e D/A
              OUT PORTB,AL
              PUSH AX
              MOV AL,DMA_DA_REG
              MOV DMA_REG,AL
              CALL DMA_INICIO
              CALL CONTADORES
              POP AX
              OR AL,20H
              OUT PORTB,AL      ;INICIA CONVERSAD D/A
              RET

```



```

;*****
;      'C' ou 'c'
;      PROGRAMA CONTADORES DE PALAVRAS
;*****

```

```

FROG_CONT:      CALL GET_START
                MOV AX,START_ADDRESS
                MOV CONTP0_REG,AL
                MOV CONTP1_REG,AH
                CALL GET_END
                MOV AX,END_ADDRESS
                MOV CONTP2_REG,AL
                RET

```

```

;*****
;      'W' ou 'w':
;      Escrever bloco de dados no segmento atual de
;      memoria de trabalho
;*****

```

```

WRITE:         CALL GET_START
                CALL GET_END
                MOV BX,START_ADDRESS
                MOV CX,END_ADDRESS

```

```

WLOOP:        CALL INPUT
                MOV DX,WORK_SEG
                MOV DS,DX
                MOV [BX],AL
                CMP BX,CX
                JZ WEND
                INC BX
                MOV AX,CS
                MOV IX,AX
                MOV DS,DX
                JMP WLOOP

```

```

WEND:         MOV AX,CS
                MOV IX,AX
                MOV DS,DX
                RET

```

```

;*****
;      'R' ou 'r':
;      Ler bloco de dados no segmento atual de
;      memoria de trabalho
;*****

```

```

READ:         CALL GET_START
                CALL GET_END
                MOV BX,START_ADDRESS
                MOV CX,END_ADDRESS

```

```

RLOOP:        MOV DX,WORK_SEG
                MOV DS,DX
                MOV AL,[BX]

```

```

                PUSH AX
                MOV AX,CS
                MOV DX,AX
                POP AX
                MOV DS,DX
                CALL OUTPUT
                CMP BX,CX
                JZ REND
                INC BX
                JMP RLOOP
REND:           RET

```

```

;*****
; 'B' ou 'b' *
; Chaveia os bancos de 64 Kbytes do SAD *
; *
;*****

```

```

SET_BANCO:     CALL GET_START
                MOV AX,START_ADDRESS
                OUT REGBANCO,AL
                RET

```

```

;*****
; 'F' ou 'f': *
; Preencher bloco de dados no segmento atual de *
; memoria de trabalho *
;*****

```

```

FILL:          CALL GET_START
                CALL GET_END
                CALL INPUT
                MOV BX,START_ADDRESS
                MOV CX,END_ADDRESS
                PUSH DS
                MOV DX,WORK_SEG
                MOV DS,DX
FLOOP:         MOV [BX],AL
                CMP BX,CX
                JZ FEND
                INC BX
                JMP FLOOP
FEND:          POP DS
                RET

```

```

;*****
;      'G' ou 'g':
;      Executar no segmento atual de memoria de
;      trabalho a partir de endereco dado
;*****

```

```

EXECUTE:      MOV BX,BX_REG      ;RESTAURA BX
              MOV CX,CX_REG      ;RESTAURA CX
              MOV DX,DX_REG      ;RESTAURA DX

              MOV BP,BP_REG      ;RESTAURA BP
              MOV SI,SI_REG      ;RESTAURA SI
              MOV DI,DI_REG      ;RESTAURA DI

              MOV AX,ES_REG      ;RESTAURA ES
              MOV ES,AX

              MOV AX,CS          ;CS NO STACK
              PUSH AX            ;PARA PERMITIR ENCERRAMENTO
              MOV AX,OFFSET INT_WAST ;DE PROCESSO FAR RETURN
              PUSH AX

              MOV AX,FL_REG      ;FLAGS NO STACK
              PUSH AX
              MOV AX,CS_REG      ;CS NO STACK
              PUSH AX
              MOV AX,IP_REG      ;IP NO STACK
              PUSH AX
              MOV AX,DS_REG      ;DS NO STACK
              PUSH AX
              MOV AX,AX_REG      ;RESTAURA AX
              POP DS             ;RESTAURA DS
              IRET               ;RESTAURA IP, CS, FLAGS

```

```

;*****
;      'S' ou 's':
;      Setar segmento de memoria de trabalho
;*****

```

```

SETSEG:      CALL GET_START
              MOV AX,START_ADDRESS
              MOV WORK_SEG,AX

              RET

```

```

;*****
;      'K' ou 'k':
;      Setar constante para deslocamento vertical
;      durante edicao de memoria
;*****

```

```

SET_CONST:      PUSH AX
                CALL INPUT
                MOV CONST_VER,AL
                POP AX
                RET

```

```

;*****
;      'E' ou 'e'
;      Entrar no modo edicao
;*****

```

```

EDIT:          CALL GET_START
EDIT_LOOP:     CALL INPUT
                CMP AL,'S'
                JNZ EDIT_CONT
                RET
                ;SAIA DO MODO EDICAO
EDIT_CONT:     MOV BX,OFFSET TAB_EDIT
                CALL SEARCH
                CMP BX,0000H
                JZ EDIT_LOOP
                CALL BX
                JMP EDIT_LOOP

```

```

TAB_EDIT      DB 'H'
                DW OFFSET ONE_AHEAD
                DB 'h'
                DW OFFSET ONE_BACK
                DB 'V'
                DW OFFSET K_AHEAD
                DB 'v'
                DW OFFSET K_BACK
                DB 'R'
                DW OFFSET NMV_READ
                DB 'W'
                DW OFFSET WRITE_AHEAD
                DB 'w'
                DW OFFSET NMV_WRITE
                DB 00H

```

```

;*****
;      ONE_AHEAD:
;*****

```

```

ONE_AHEAD:     PUSHF
                INC START_ADDRESS
                POPF
                RET

```

```
; *****
; ONE_BACK: *
; *****
```

```
ONE_BACK:      PUSHF
                DEC  START_ADDRESS
                POPF
                RET
```

```
; *****
; K_AHEAD: *
; *****
```

```
K_AHEAD:      PUSHF
                PUSH AX
                XOR  AX,AX
                MOV  AL,CONST_VER
                ADD  START_ADDRESS,AX
                POP  AX
                POPF
                RET
```

```
; *****
; K_BACK: *
; *****
```

```
K_BACK:      PUSHF
                PUSH AX
                XOR  AX,AX
                MOV  AL,CONST_VER
                SUB  START_ADDRESS,AX
                POP  AX
                POPF
                RET
```

```
; *****
; NMV_READ: *
; *****
```

```
NMV_READ:    PUSH AX
                PUSH BX
                PUSH DS
                MOV  BX,START_ADDRESS
                MOV  AX,WORK_SEG
                MOV  DS,AX
                MOV  AL,[BX]
                CALL OUTPUT
                POP  DS
                POP  BX
                POP  AX
                RET
```

```
;*****  
; WRITE_AHEAD: *  
;*****
```

```
WRITE_AHEAD:    PUSHF  
                PUSH AX  
                PUSH BX  
                PUSH DS  
                CALL INPUT  
                MOV BX,START_ADDRESS  
                MOV DX,WORK_SEG  
                MOV DS,DX  
                MOV [BX],AL  
                MOV AL,[BX]  
                POP DS  
                CALL OUTPUT  
                INC START_ADDRESS  
                POP BX  
                POP AX  
                POPF  
                RET
```

```
;*****  
; NMV_WRITE: *  
;*****
```

```
NMV_WRITE:     PUSH AX  
                PUSH BX  
                PUSH DS  
                CALL INPUT  
                MOV BX,START_ADDRESS  
                MOV DX,WORK_SEG  
                MOV DS,DX  
                MOV [BX],AL  
                MOV AL,[BX]  
                POP DS  
                CALL OUTPUT  
                POP BX  
                POP AX  
                RET
```

```

;*****
;   'X' ou 'x':
;   Transmitir pseudo registradores
;   Ordem de transmissao:
;       AH, AL, BH, BL, CH, CL, DH, DL,
;       SPH, SPL, BPH, BPL, SIH, SIL,
;       DIH, DIL, IPH, IPL, CSH, CSL,
;       DSH, DSL, SSH, SSL, ESH, ESL,
;       FLAGSH, FLAGSL
;*****

```

```

SEND_REGS:      MOV BX,OFFSET AX_REG
                MOV DX,14
                MOV CL,8
SR_LOOP:        MOV AX,[BX]
                SHR AX,CL
                CALL OUTPUT
                MOV AX,[BX]
                CALL OUTPUT
                ADD BX,2
                DEC DX
                JNZ SR_LOOP
                RET

```

```

;*****
;   'Y' ou 'y':
;   Recebe e seta pseudo registradores
;   Ordem de recepcao:
;       AH, AL, BH, BL, CH, CL, DH, DL,
;       SPH, SPL, BPH, BPL, SIH, SIL,
;       DIH, DIL, IPH, IPL, CSH, CSL,
;       DSH, DSL, SSH, SSL, ESH, ESL,
;       FLAGSH, FLAGSL
;*****

```

```

SET_REGS:       MOV BX,OFFSET AX_REG
                MOV DX,14
ST_LOOP:        CALL INPUT
                MOV [BX]+1,AL
                CALL INPUT
                MOV [BX],AL
                ADD BX,2
                DEC DX
                JNZ ST_LOOP
                RET

```

```

;*****
;      'M' ou 'm':
;      Transmite segmento inicial da RAM
;*****

```

```

RAM_START:      PUSHF
                PUSH AX
                PUSH BX
                PUSH CX

                MOV AX,OFFSET RAM
                ADD AX,15
                AND AX,0FFF0H
                MOV CL,4
                SHR AX,CL
                MOV BX,CS
                ADD AX,BX
                PUSH AX

                MOV CL,8
                SHR AX,CL
                CALL OUTPUT
                POP AX
                CALL OUTPUT
                POP CX
                POP BX
                POP AX
                POPF
                RET

```

```

;*****
;      Subrotinas de uso geral no programa
;*****

```

```

;*****
;      GERA UMA TAXA DE AMOSTRAGEM DE    F KHz DE ACORDO *
;      COM O CONTEUDO DE AL, ATRAVES DO 8253
;*****

```

```

GERAR_TAXA:     MOV AL,36H
                OUT TIMER_CONTR,AL
                MOV AL,02H
                OUT TIMER0,AL
                MOV AL,00H
                OUT TIMER0,AL

                MOV AL,01110100B
                OUT TIMER_CONTR,AL
                MOV AL,TAXA_REG
                OUT TIMER1,AL

                MOV AL,00
                OUT TIMER1,AL

```


RET

```
;*****
;          PROGRAMA CONTADORES DE ENDEREÇOS E PALAVRAS      *
;          PARA TRANSFERENCIA VIA DMA                       *
;*****
```

```
CONTADORES:      MOV AL,00H
                  OUT CONTE0,AL
                  OUT CONTE1,AL
                  OUT CONTE2,AL

                  MOV AL,CONTP0_REG
                  OUT CONTP0,AL
                  MOV AL, CONTP1_REG
                  OUT CONTP1,AL
                  MOV AL,CONTP2_REG
                  OUT CONTP2,AL
                  RET
```

```
;*****
;          SEARCH:                                          *
;          Busca de endereço em tabela                     *
;          Al na tabela: BX = BX(byte)                    *
;          AL auxente:  BX= 0000h                         *
;*****
```

```
SEARCH:          PUSHF
                  PUSH AX
SEARCH_LOOP:     CMP CS:BYTE PTR [BX],0
                  JZ NOT_FOUND
                  CMP AL,CS:[BX]
                  JNZ TRY_NEXT
                  INC BX
                  MOV BX,CS:[BX]
                  POP AX
                  POPF
                  RET
TRY_NEXT:       ADD BX,3
                  JMP SEARCH_LOOP
NOT_FOUND:      XOR BX,BX
                  POP AX
                  POPF
                  RET
```

```

;*****
;      GET_START:                                     *
;      Recepcão de endereço inicial (2 bytes)      *
;*****

```

```

GET_START:      PUSHF
                PUSH BX
                MOV BX,OFFSET START_ADDRESS
ENT_AD:         CALL INPUT
                MOV CH,AL
                PUSH AX
                PUSH CX
                CALL INPUT
                MOV CL,AL
                MOV [BX],CX
                POP CX
                POP AX
                POP BX
                POPF
                RET

```

```

;*****
;      GET_END:                                       *
;      Recepcão de endereço final (2 bytes)        *
;*****

```

```

GET_END:        PUSHF
                PUSH BX
                MOV BX,OFFSET END_ADDRESS
                JMP ENT_AD

```

```

;*****
;      Subrotinas de atendimento a interrupções    *
;*****

```

```

;*****
;      INT_WAST                                     *
;      Software interrupt para reentrada no       *
;      monitor (warm start)                       *
;*****

```

```

INT_WAST:       CLI
                PUSH DS                ;SALVA "DS" E "AX" DO PROCESSO
                PUSH AX                ;INTERROMPIDO NO STACK DO MESMO
                MOV AX,CS              ;DS <-- DS DO MONITOR
                MOV DS,AX
                POP AX                 ;SALVA AX
                MOV AX_REG,AX

```

```

POP AX ; SALVA DS
MOV DS_REG, AX
POP AX ; SALVA IP
MOV IP_REG, AX
POP AX ; SALVA CS
MOV CS_REG, AX
POP AX ; SALVA FLAGS
MOV FL_REG, AX
MOV AX, SP ; SALVA SP
MOV SP_REG, AX
MOV AX, SS ; SALVA SS
MOV SS_REG, AX
MOV AX, CS ; SS, SP <-- SS, SP DO MONITOR
MOV SS, AX
MOV SP, (OFFSET TOP_STACK)+1
MOV BX_REG, BX ; SALVA BX
MOV CX_REG, CX ; SALVA CX
MOV IX_REG, IX ; SALVA IX
MOV BP_REG, BP ; SALVA BP
MOV SI_REG, SI ; SALVA SI
MOV DI_REG, DI ; SALVA DI
MOV AX, ES ; SALVA ES
MOV ES_REG, AX

MOV AL, 0FFH ; SINALIZA REENTRADA NO MONITOR
CALL OUTPUT
MOV BX, OFFSET AX_REG ; TRANSMITE REGS VIA RS-232C
MOV DX, 14 ; QUATORZE REGISTRADORES
MOV CL, 8 ; CONSTANTE PARA DESLOCAMENTOS
TRANS_REG: MOV AX, [BX] ; AH, AL, BH, BL, CH, CL, DH, DL, SPH, SPL,
SHR AX, CL ; BPH, BPL, SIH, SIL, DIH, DIL, IPH, IPL,
CALL OUTPUT ; CSH, CSL, DSH, DSL, SSH, SSL, ESH, ESL,
MOV AX, [BX] ; FLAGSH, FLAGSL
CALL OUTPUT
ADD BX, 2
DEC DX
JNZ TRANS_REG
JMP LOOP1 ; "*" E AGUARDE O PROXIMO COMANDO

```

```

; *****
; INT_TRAP *
; *
; Trap interrupt para execucao passo a *
; passo *
; *****

```

```
INT_TRAP: JMP INT_WAST
```

```

; *****
; Subrotinas dependentes do hardware *
; *****

```

```

;*****
;      INIT_HARD:                                     *
;      Inicializacao do hardware                     *
;*****
INIT_HARD:      MOV  VDSR,00H           ;RS-232C (8251)
                MOV  VDTR,00H
                MOV  AL,0CEH
                OUT  SIOCONT,AL
                MOV  AL,27H
                OUT  SIOCONT,AL

TIMERINI:      MOV  AL,37H             ;TIMER #1: 1250KHZ/18
                OUT  TMCNT,AL          ;PARA GERAR (T = 4MS/256)
                MOV  AL,18H           ;MEMORY REFRESH
                OUT  TM0,AL
                MOV  AL,00H
                OUT  TM0,AL

DMAINI:        MOV  AL,01010000B      ;MEMORY TO MEMORY DISABLE
                                                ;ADDRESS HOLD DISABLE
                                                ;ENABLE CONTROLLER
                                                ;FIXED PRIORITY
                                                ;X
                                                ;DREQ SENSE ACTIVE LOW
                                                ;DACK SENSE ACTIVE LOW

                OUT  DMACOM,AL
                MOV  AL,01011000B      ;CHANNEL 0
                                                ;READ TRANSFER
                                                ;ENABLE AUTOINITIALIZATION
                                                ;ADDRESS INCREMENT
                                                ;SINGLE MODE SELECT

                OUT  DMAMOD,AL
                MOV  AL,00H            ;CLEAR BYTE POINTER FLIP-FLOP
                OUT  DCLPFF,AL
                MOV  AX,2000H          ;LOAD ADDRESS REGISTER AND

COUNTER

                OUT  BASE0,AL
                MOV  AL,AH
                OUT  BASE0,AL
                MOV  AX,256
                OUT  COUNT0,AL
                MOV  AL,AH
                OUT  COUNT0,AL
                MOV  AL,00001100B      ;ENABLE DMA TRANSFERS ON CHANNEL 0
                OUT  DMASKW,AL         ;AND CHANNEL 1

                MOV  AL,0101010101B    ;CHANNEL 1
                OUT  DMAMOD,AL

                MOV  AL,0FFH           ;LOAD ADDRESS REGISTER AND COUNTER

```

```

OUT BASE1,AL
MOV AL,00H
OUT BASE1,AL
MOV AL,0FFH
OUT COUNT1,AL
MOV AL,00H
OUT COUNT1,AL

```

```

MOV AL,00001100B ;ENABLE DMA TRANSFERS ON CHANNEL 0
OUT DMASKW,AL ;AND CHANNEL 1

```

```

INIT_INTERFP: MOV AL,80H ;INICIALIZA 8255 PROGRAMANDO SEUS
OUT PORT_CONTR,AL ;PORTS COMO SAIDA
MOV AL,PORTA_REG ;CONFIGURA SAD COM GANHO DE 0dB
OUT PORTA,AL ;E FILTRO DE 3,4 KHz
MOV AL,PORTB_REG
OUT PORTB,AL

```

```

INTINI: MOV AL,17H ;CONTROLADOR DE INTERRUPCOES
OUT INTL,AL
MOV AL,20H ;ICW2
OUT INTH,AL
MOV AL,03H ;ICW4
OUT INTH,AL
MOV AL,0FFH ;DESABILITE TODOS OS CANAIS
MOV MASCARA,AL ;DE INTERRUPCAO
OUT INTH,AL

```

```

VETINI: PUSH DS ;CONSTRUA VETORES DE
XOR AX,AX ;INTERRUPCAO
MOV DS,AX
MOV BX,4*VWARM_START ;VETOR DE REENTRADA NO MONITOR
MOV [BX],OFFSET INT_WAST
MOV [BX]+2,CS
MOV BX,4*VTRAP ;VETOR DE TRAP (UNICO PASSO)
MOV [BX],OFFSET INT_TRAP
MOV [BX]+2,CS
POP DS
RET

```

```

;*****
;  PROGRAMACAO DO CONTROLADOR DE DMA 8237 PARA AS      *
;  CONVERSOES A/D e D/A                               *
;*****

```

```

DMA_INICIO:      MOV AL,DMA_REG          ;CHANNEL 1
                                     ;READ TRANSFER/WRITE TRANSF
                                     ;ENABLE AUTOINITIALIZATION
                                     ;ADDRESS INCREMENT
                                     ;SINGLE MODE SELECT

                OUT DMAMOD,AL

                MOV AL,0FFH            ;LOAD ADDRESS REGISTER AND COUNTER
                OUT BASE1,AL
                MOV AL,00H
                OUT BASE1,AL
                MOV AL,0FFH
                OUT COUNT1,AL
                MOV AL,00H
                OUT COUNT1,AL
                MOV AL,00001100B      ;ENABLE DMA TRANSFERS
                OUT DMASKW,AL
                RET

```

```

;*****
;  INPUT:                                             *
;  Recepcao de um byte pela RS-232C                 *
;  Retorno:          A1 = caracter                   *
;*****

```

```

INPUT:          PUSHF
                PUSH BX
                PUSH DS
                MOV AX,CS
                MOV BX,AX
                MOV DS,BX
INPUT_LOOP:    IN AL,SIOCONT
                AND AL,02H
                JZ INPUT_LOOP
                IN AL,SIODT
                MOV BL,AL
                MOV AL,VIDTR
                XOR AL,01H
                MOV VIDTR,AL
                ROL AL,1
                XOR AL,27H
                OUT SIOCONT,AL
                MOV AL,BL
                POP DS
                POP BX
                POPF
                RET

```

```

;*****
;      IN_STAT:                                     *
;      Status da recepcao RS-232C                 *
;      Retorno:      A1 = 00h -> nao ha byte *
;                  A1 = 01h -> ha byte *
;*****

```

```

IN_STAT:      PUSHF
              IN AL,SIOCONT
              AND AL,02H
              ROR AL,1
              POPF
              RET

```

```

;*****
;      OUTPUT:                                     *
;      Transmissao de um byte pela RS-232C        *
;      Parametros:      A1 = byte *
;*****

```

```

OUTPUT:      PUSHF
              PUSH AX
              PUSH BX
              PUSH DS
              PUSH AX
              MOV AX,CS
              MOV BX,AX
              POP AX
              MOV DS,BX
              MOV BL,AL
OUTPUT_LOOP: IN AL,SIOCONT
              AND AL,80H
              ROL AL,1
              XOR AL,01H
              MOV BH,AL
              MOV AL,VDSR
              XOR AL,BH
              JNZ OUTPUT_LOOP
              MOV AL,BL
              OUT SIODT,AL
              MOV AL,VDSR
              XOR AL,01H
              MOV VDSR,AL
              POP DS
              POP BX
              POP AX
              POPF
              RET

```

```
; *****
; Area da Pilha *
; *****
```

```
TOP_STACK      DB SIZE_STACK DUP (?)
               DB (?)
```

```
; *****
; Variaveis do Programa *
; *****
```

```
WORK_SEG      DW (?)
START_ADDRESS DW (?)
END_ADDRESS   DW (?)
DATA_BYTE     DB (?)
CONST_VER     DB (?)
MASCARA       DB (?)
VDSR          DB (?)
VDTR          DB (?)
DMA_REG       DB (?)
DMA_AD_REG    DB (?)
DMA_DA_REG    DB (?)
BANCO_REG     DW (?)
PORTA_REG     DB (?)
PORTB_REG     DB (?)
TAXA_REG      DB (?)
CONTP0_REG    DB (?)
CONTP1_REG    DB (?)
CONTP2_REG    DB (?)
```

```
; *****
; Pseudo registradores *
; *****
```

```
AX_REG        DW (?)
BX_REG        DW (?)
CX_REG        DW (?)
DX_REG        DW (?)
SP_REG        DW (?)
BP_REG        DW (?)
SI_REG        DW (?)
DI_REG        DW (?)
IP_REG        DW (?)
CS_REG        DW (?)
DS_REG        DW (?)
SS_REG        DW (?)
ES_REG        DW (?)
FL_REG        DW (?)
```



```
;*****  
; Segmento inicial da RAM para carga de programas *  
;*****
```

```
RAM          DW $  
SEGMENTOS    ENDS  
END          START
```