

UNIVERSIDADE ESTADUAL DE CAMPINAS - UNICAMP

FACULDADE DE ENGENHARIA ELETRICA

DEPARTAMENTO DE ENGENHARIA BIOMEDICA

PROJETO E CONSTRUÇÃO DE UM ANALISADOR DE ESPECTRO EM
TEMPO REAL PARA RESPOSTA DOPPLER DE SINAIS
ULTRA-SÔNICOS NA FAIXA DE 150 A 3000 Hz

Autor : Alexandre Henrique Hermini
Orientador : Prof. Dr. Salde Jorge Calil

Tese apresentada à Faculdade de Engenharia Elétrica da
UNICAMP para obtenção do grau de Mestre em Engenharia Elétrica

Campinas - 1992

Este exemplar corresponde à versão final da tese
defendida por Alexandre Henrique Hermini
e aprovada pela Comissão
Ju'gadora em 18.03.92.

Salde Jorge Calil
Orientador

AOS MEUS PAIS

CAMILO HENRIQUE E MARIA LUIZA

POR SUA DEDICAÇÃO AO LONGO

DE MINHA VIDA,

A MINHA COMPANHEIRA VANIA SCHNEIDER
PELO CARINHO E INCENTIVO
E MEUS IRMÃOS ADRIANO E ANDRÉ

AGRADECIMENTOS

- Ao Prof. Dr. Saide Jorge Calil pela orientação e apoio.
- Aos Professores Alberto Cliquet Jr., Eduardo T. Costa, José W.M. Bassani, Maria A.C. Farias, Rosana A. Bassani, Sergio S. Mühlen e Vera L.S.N. Button pela colaboração e amizade.
- À Sra. Marcia S.I. Camargo pelo apoio na digitação do trabalho.
- Aos técnicos Eugênio C. Carrara e Marcio T. Watoniki, Edward Werninghaus e Marcello Borges pelo apoio na confecção do protótipo e testes.
- À física Rita Elaine F.Corte, Enga. Erica Wall, Icaro F. Bellentani, Humberto R.Gamba e Carlos Marcelo G. de Godoy, pela amizade e incentivo.
- Aos engenheiros Adriano R.Leusqui e Waldomiro N. dos Santos pelo trabalho no programa de aquisição e apresentação.
- Ao desenhista Mauro Martinazzo, pela confecção dos diagramas e figuras.
- À enfermeira Dalva A. dos S. Galeano pelo apoio nos testes clínicos.
- A todos os amigos e companheiros não citados mas que participaram desta jornada, o meu muito obrigado.

INDICE

CAPÍTULO 1 - INTRODUÇÃO.....	01
CAPÍTULO 2 - FUNDAMENTOS TEÓRICOS.....	04
2.1 - O Efeito Doppler.....	04
2.2 - Sistemas Doppler na Avaliação Vascular....	07
2.2.1 - Composição do Sangue.....	07
2.2.2- Comportamento do Fluxo.....	08
2.3 - Sistemas Ultra-sônicos Doppler.....	11
2.4 - Analisadores de Espectro.....	14
2.4.1 - Analisador Multicanais.....	15
2.4.2 - Processamento Digital de Sinais.....	17
2.3.4 - Filtros de Varredura.....	17
CAPÍTULO 3 - METODOLOGIA.....	23
3.1 - Introdução.....	23
3.2 - Descrição do Sistema.....	26
3.3 - Módulo de Condicionamento de Sinais.....	27
3.3.1 - Amplificador de Entrada.....	28
3.3.2 - Filtro Passa Faixa.....	29
3.4 - Módulo de Conversão AD.....	33
3.5 - Módulo de Processamento Digital.....	37

3.6 - Módulo de Processamento Analógico.....	43
3.7 - Interface Paralela.....	49
3.8 - Programa de Aquisição e Apresentação.....	50
3.9 - Fonte de Alimentação.....	58
3.9.1 - Fonte de + 5 [V].....	58
3.9.2 - Fonte de $\pm$ 15 [V].....	60
CAPÍTULO 4 - TESTES E RESULTADOS.....	63
4.1 - Testes do Filtro de Entrada.....	63
4.2 - Testes do Filtro Passa-Faixa de 70 kHz.....	65
4.3 - Testes do Sistema de Deslocamento do Espectro.	67
4.3 - Teste de Desempenho do Analisador de Espectro.	69
4.4 - Análise de Espectro Doppler Ultra-sônico.....	72
CAPÍTULO 5 - CONCLUSÕES E SUGESTÕES.....	74

CAPÍTULO 1 - INTRODUÇÃO

Os diferentes tipos de processamento que podem ser utilizados no tratamento de sinais Doppler ultra-sônicos têm levado ao desenvolvimento de sistemas não invasivos, para auxílio à avaliação do comportamento do fluxo vascular. Os métodos de processamento podem ser classificados em duas técnicas: conversão frequência-tensão e análise espectral (HERMINI e CALIL, 1990).

Devido a característica do perfil de velocidades desenvolvido no interior de um vaso sanguíneo, o sinal Doppler obtido de avaliações vasculares apresenta diferentes componentes de frequência. Este conjunto de frequências se constitui no espectro Doppler (WINJ et alii, 1987).

Sistemas que utilizam processamento por conversores frequência-tensão, fornecem em sua saída um sinal cuja amplitude é proporcional à frequência do sinal de entrada. Embora estes sistemas utilizem circuitos simples e de baixo custo, apresentam a desvantagem de tratar o sinal Doppler como uma única frequência,

resultante do valor médio das componentes do espectro, induzindo erros na avaliação (LUNT, 1975 ; GOSLING, 1976).

A análise espectral é uma técnica de processamento capaz de fornecer informações da amplitude das componentes de frequência de um sinal (LATHI, 1979). Quando empregada em sistemas de avaliação vascular, esta técnica consiste na forma mais completa de processamento de sinais Doppler ultra-sônicos (BROWN, 1982).

Uma característica de fundamental importância na concepção de um sistema para avaliação do fluxo vascular é que o processamento seja realizado em tempo real, isto é, que os resultados sejam apresentados simultaneamente à entrada dos dados, permitindo ao operador repetir a investigação nos pontos de maior interesse durante o exame e seleccionar os resultados a serem arquivados (ZUECH et alii, 1984).

Os primeiros registros da utilização da análise espectral de sinais Doppler em avaliações vasculares, foram feitos por Kaneko (1965) e processava as informações "off-line". Quatro anos mais tarde Light (1969) demonstrou as vantagens do processamento em tempo real, utilizando um sistema baseado em um conjunto de filtros operando em paralelo, cada qual centrado em uma faixa de frequência do espectro a ser analisado (ZUECH et alii, 1976 ; COGHLAN et alii, 1976). Esta técnica embora opere em tempo real conforme descrito no capítulo 2 (item 2.4.1), apresenta a desvantagem de necessitar circuitos complexos e registradores especiais para impressão dos resultados (LUNT, 1975).

Coghlan et alii (1973) desenvolveram um sistema de análise

espectral baseado em "time-compression", que opera em tempo real. Este sistema realiza a varredura do espectro sobre um único filtro passa faixa, utilizando-se de multiplicadores analógicos de frequência. Para permitir a operação em tempo real é utilizada uma compressão, no tempo, do sinal de entrada reduzindo o tempo necessário à análise de cada componente de frequência. Esta técnica reduz a complexidade do sistema em relação ao desenvolvido por Light (1969).

Uma outra técnica utilizada para este tipo de análise, consiste em um processamento matemático, geralmente baseado na transformada rápida de Fourier e implementada em computadores. Esta análise foi implementada em tempo real por Di Giulio Maria et alii (1990), e utiliza de componentes dedicados para processamento matemático (TEXAS TMS 320). Schildwein e Evans (1989) utilizaram-se deste mesmo tipo de processador para implementar a análise espectral através de processamento por modelamento autoregressivo.

A não disponibilidade de componentes para processamento matemático no mercado nacional, quando do início deste trabalho, levou ao desenvolvimento de um analisador de espectro que utilizasse os componentes disponíveis. O analisador desenvolvido é baseado nas técnicas de "time-compression" (COGHLAN et alii, 1973) e filtro de varredura (ATKINSON e WOODCOOCK, 1982) (vide Capítulo 2). A técnica utilizada neste trabalho diferencia-se daquelas descritas por Coghlan et alii (1973) e Atkinson e Woodcoock (1982), por realizar de maneira digital a varredura do espectro

sobre o filtro, conforme descrito no capítulo 3 deste trabalho.

A descrição dos testes e os resultados obtidos são apresentados no capítulo 4 deste trabalho.

CAPÍTULO 2 - FUNDAMENTOS TEÓRICOS

2.1 - O Efeito Doppler

O som é uma forma de energia mecânica que se propaga em meios materiais sob a forma de ondas longitudinais. Se o meio de propagação é o ar e a frequência de oscilação do som encontra-se na faixa de 20 a 20.000 Hz, esta forma de energia é capaz de sensibilizar o sistema auditivo humano íntegro. Sinais sonoros com frequência superior a 20.000 Hz não são audíveis, e denominados ultra-som (NCRP, 1986).

Quando um transmissor emite uma onda ultra-sônica em direção a um objeto estacionário, por espalhamento ou reflexão, parte do sinal emitido retorna à fonte com a mesma frequência do sinal emitido. Caso haja movimento relativo entre a fonte e o refletor, o sinal de retorno será percebido com uma alteração de frequência proporcional à velocidade de deslocamento do objeto. Este fenômeno físico é denominado efeito Doppler (MAGNIN, 1986; NCRP, 1986; FOLMER-JOHNSON, 1968).

A figura 2.1 ilustra o fenômeno descrito acima, e a equação 2.1 sua representação matemática.

$$f_r = f_t + \frac{2 \cdot V}{C} \cdot f_t \quad 2.1$$

Onde :

f_r = Frequência do sinal recebido

f_t = Frequência do sinal transmitido

V = Velocidade de deslocamento do objeto

C = Velocidade de propagação do som no meio em questão



Fig. 2.1 - Efeito Doppler

A equação 2.1 somente é válida para caso em que o objeto que reflete o sinal se desloca na mesma direção da propagação do som. No caso do som não se propagar na mesma direção do movimento do refletor, introduz-se o fator cosseno do ângulo (θ) formado entre as direções de propagação do som e do refletor. Este fator multiplicará o valor do desvio de frequência apresentado na equação 2.1 (PINARDI e SAINZ, 1977). Esta situação está representada matematicamente na equação 2.2 e demonstrada na figura 2.2.

$$f_r = f_t + \frac{2 V}{C} \cos \theta \cdot f_t \quad 2.2$$

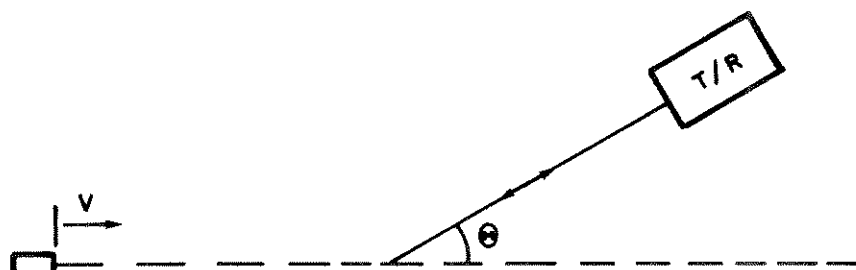


Figura 2.2 - Efeito Doppler obtido quando a direção de propagação do som e a direção do movimento do objeto são diferentes.

Assim o desvio de frequência Doppler (f_d) é obtido a partir da diferença entre a frequência recebida (f_r) e a transmitida (f_t), como apresentado na equação 2.3:

$$f_d = \frac{2 \cdot V}{C} \cos \theta \cdot f_t \quad 2.3$$

Satomura e Franklin utilizaram-se de instrumentação baseada no efeito Doppler de sinais ultra-sônicos, como técnica não invasiva para auxílio na avaliação do comportamento do fluxo vascular (FELIX et alii, 1975).

2.2 - Aplicações de sistemas ultra-sônicos no estudo do comportamento do fluxo vascular.

Para melhor entendimento de como estes sistemas podem ser utilizados no estudo do comportamento do fluxo vascular, faz-se necessário tecer algumas considerações sobre esse comportamento e sobre a composição do sangue.

2.2.1 - Composição do Sangue

Na parte líquida do sangue, o plasma, encontram-se partículas em suspensão, hemácias (representando aproximadamente 45% do volume de células do sangue), leucócitos e plaquetas. Presume-se que as hemácias sejam as maiores responsáveis pelo retorno do som, pois embora os leucócitos possuam maior volume celular que as hemácias, estas apresentam-se em maior concentração (ATKINSON e WOODCOCK, 1982).

As hemácias possuem forma de disco com aproximadamente 7 micra de diâmetro por 2 micra de espessura. Uma vez que os sistemas ultra-sônicos destinados à avaliação vascular utilizam-se de sinais com frequências entre 2 e 10 MHz (FELIX et alii, 1975), resultando em comprimentos de onda entre 770 e 154 micra, respectivamente, e sendo as hemácias muito menores que estes comprimentos de onda, ao atingir uma hemácia, o sinal ultra-sônico sofre um efeito denominado espalhamento, que é responsável pelo

retorno à fonte de parte do sinal emitido (ATKINSON e WOODCOOCK, 1982; GOSLING, 1976 ; ROBERTS, 1974).

As informações apresentadas até este ponto permitem associar o desvio de frequência Doppler à velocidade de deslocamento de uma hemácia, uma vez que esta é uma partícula em movimento que promove o retorno do sinal ultra-sônico ao transmissor.

2.2.2 - Comportamento do Fluxo

O fluxo contínuo de um fluido newtoniano em um tubo rígido apresenta um gradiente de velocidades distribuído ao longo da seção transversal do tubo, com baixa velocidade próximo às paredes e velocidade máxima no centro do tubo. Entre o centro e as paredes do tubo, as velocidades são distribuídas de forma parabólica, segundo a equação 2.4, atribuído o valor 2 (em razão da distribuição parabólica) para o expoente "n" (WIJN et alii, 1987).

$$V_r = V_m \cdot \left[1 - \left(\frac{r}{R} \right)^n \right] \quad 2.4$$

A figura 2.3 representa o perfil de velocidades desenvolvido no interior do tubo.

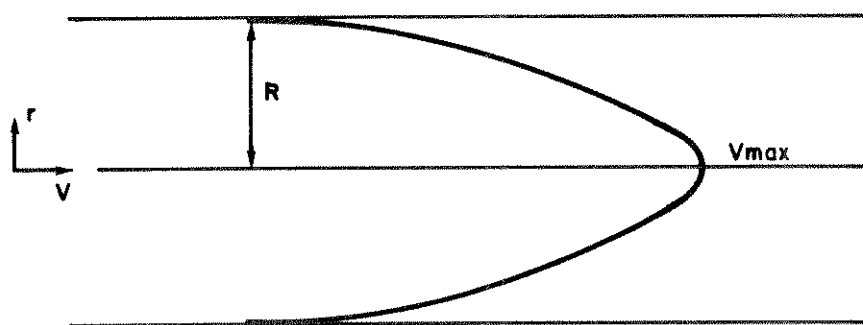


Fig. 2.3 - Perfil de velocidades desenvolvido por um fluxo contínuo de um fluido newtoniano (adaptado de Scheu, 1972)

Em uma artéria, o fluxo é pulsátil, apresentando diferenças de comportamento nos períodos sistólico e diastólico. Durante a sístole, o fluxo apresenta um perfil plano, ou seja, a velocidade do sangue é praticamente constante em toda seção transversal do vaso, com exceção das lâminas muito próximas à parede, onde a velocidade é quase nula em relação às regiões centrais do vaso. Na diástole, a distribuição das velocidades aproxima-se de um perfil parabólico, comportando-se de forma semelhante a um fluido ideal (WIJN et alii, 1987).

O perfil de fluxo durante o período sistólico pode ser equacionado matematicamente substituindo-se o valor do expoente da equação 2.4 por 10 (WIJN et alii, 1987). A figura 2.4 ilustra a distribuição das velocidades do fluido ao longo da seção transversal do vaso, mostrando a relação entre velocidade e posição da lâmina em relação ao centro do vaso.

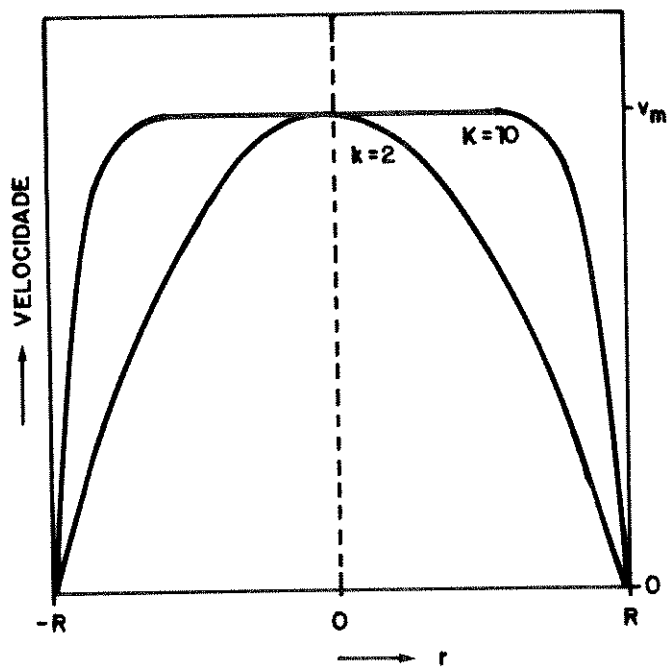


fig.2.4 - Distribuição de velocidades ao longo da seção transversal o vaso (adaptado de WIJN et alii, 1987).

A presença de diferentes velocidades durante o mesmo instante de tempo faz com que a resposta Doppler resultante de um vaso sanguíneo seja composta por um conjunto de sinais, cada qual com uma frequência proporcional à velocidade das hemácias que promoveram o retorno do sinal à fonte. A este conjunto de sinais denomina-se espectro de frequências (WIJN et alii, 1987; THOMPSON et alii, 1990).

2.3 - Sistemas ultra-sônicos Doppler

Sistemas ultra-sônicos baseados no efeito Doppler vêm se mostrando de extrema versatilidade no estudo não invasivo do comportamento do fluxo vascular, uma vez que a resposta Doppler pode ser processada de diferentes maneiras (LUNT, 1975; ATKINSON e WOODCOCK, 1982; ZUECH et alii, 1984; SILLESEN et alii, 1988), fornecendo informações numa razão direta entre complexidade do sistema e confiabilidade das informações obtidas.

Os sistemas Doppler ultra-sônicos utilizados em avaliação vascular podem ser divididos em dois grupos:

- Sistemas de onda contínua.
- Sistemas pulsáteis.

Os sistemas de onda contínua (OC) utilizam transdutores compostos por 2 cristais, montados lado a lado (figura 2.5), um como transmissor e outro como receptor. O cristal transmissor emite continuamente um sinal de frequência fixa em direção ao vaso sanguíneo. Devido à velocidade das partículas do sangue em movimento, este sinal retornará ao cristal receptor com um desvio de frequência que situa-se na faixa de áudio, quando obtido a partir de avaliações vasculares (LUNT, 1975; STEVENS, 1982; RICHARDSON et alii, 1987).

A figura 2.5 apresenta o diagrama ,em blocos, de um sistema ultra-sônico de onda contínua.

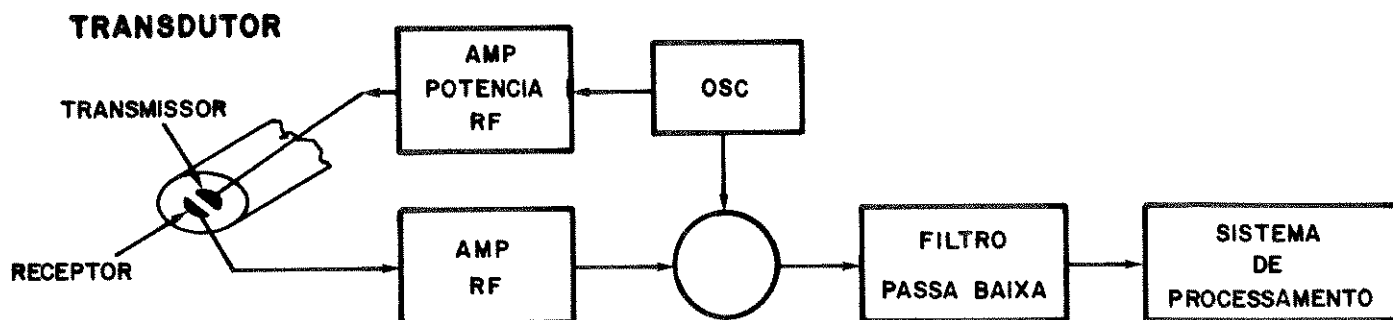


Fig. 2.5 - Diagrama em blocos de um sistema transmissor-receptor ultra-sônico de onda contínua.

A forma mais simples de apresentação do sinal Doppler consiste na demodulação e filtragem do sinal ultra-sônico, aplicação deste a um amplificador de potência de áudio e posteriormente a um alto-falante, fornecendo sons, cuja frequência é proporcional à velocidade do fluxo.

Uma desvantagem desta técnica é que sendo a análise realizada a partir dos sons obtidos em diversos pontos ao longo do vaso, e considerando-se que o ângulo formado pela posição do transdutor em relação ao eixo do vaso também altera a frequência dos sons produzidos (pela alteração do fator cosseno da equação 2.3) chega-se a conclusões errôneas. Outra desvantagem deste método é o fato de não produzir um registro gráfico das informações obtidas, dificultando uma análise comparativa e o armazenamento destas informações.

Um registro gráfico simples e de baixo custo pode ser obtido

aplicando-se o sinal Doppler a conversores frequência-tensão que utilizam detetores por cruzamento de zero. Esses conversores apresentam uma tensão de saída proporcional ao número de vezes que o sinal de entrada cruza a linha de zero, ou seja, sua frequência média no intervalo de tempo considerado. Este método apresenta limitações, uma vez que trata o espectro da resposta Doppler como uma única frequência, resultante do valor médio quadrático das componentes do sinal de entrada, não considerando a complexidade do espectro. O gráfico apresentado na saída, fornece informações da velocidade média quadrática instantânea em função do tempo (LUNT, 1975).

A análise espectral apresenta-se como a mais completa forma de processamento dos sinais Doppler de onda contínua, uma vez que trata a resposta Doppler como um espectro de frequências, fornecendo informações da amplitude das componentes de frequência do sinal, apresentando alta precisão na análise dos sinais, a partir de circuitos relativamente simples e de baixo custo (BROWN, 1982).

Sistemas Doppler pulsáteis fornecem informações de velocidade e resolução espacial das lâminas de fluxo que compõem a resposta Doppler. Isto é obtido através da emissão periódica de pulsos ultra-sônicos, ou seja, ao invés da emissão contínua, como no caso dos sistemas OC, um único cristal (na maioria dos casos) emite um "burst" e a recepção é ativada em diferentes intervalos de tempo, entre dois "bursts" consecutivos. Desta forma, como a velocidade de propagação do som em tecidos biológicos é

praticamente constante (1540 m/s), o tempo necessário ao retorno do "burst" é diretamente proporcional à distância da partícula em relação ao cristal, sendo possível analisar através de um sistema pulsátil de múltiplos canais, as lâminas de velocidade que compõem o perfil de velocidade no vaso sobre a investigação (GAMBA, 1989).

Este método, embora forneça informações precisas e confiáveis, possui a desvantagem de necessitar equipamentos cujos circuitos eletrônicos são complexos e de alto custo.

2.4 - Analisador de Espectro

A análise espectral consiste em um processamento capaz de fornecer informações da amplitude das componentes de frequência de um sinal (LATHI, 1979).

Através deste processamento é possível registrar a amplitude das componentes de frequência do espectro Doppler. A amplitude de cada faixa de frequência é diretamente proporcional à quantidade de hemácias que produziram o desvio de frequência do sinal, fornecendo maior número de informações sobre a investigação realizada, aumentando a confiabilidade do diagnóstico (THOMPSON et alii, 1990).

Uma característica importante de um sistema de análise espectral para auxílio na avaliação do fluxo vascular, é que o mesmo forneça informações em tempo real, permitindo ao operador

repetir a investigação nos pontos de maior interesse durante o mesmo exame, bem como a seleção dos resultados do processamento a serem arquivados (ZUECH et. alii, 1984)

Existem três técnicas descritas na literatura, que fornecem a análise do espectro Doppler em tempo real (Hermini e Calil, 1990):

A - Analisadores multicanais ou processamento paralelo.

B -Processamento digital de sinais (ex. FFT e modelagem auto-regressiva)

C - Filtros de varredura.

2.4.1 - Analisadores multicanais

São compostos por um banco de filtros eletrônicos (analógicos ou digitais), cada qual centrado na faixa de frequência a ser analisada, fornecendo na saída de cada célula de filtro, a amplitude da componente na qual o filtro encontra-se centrado. (ZUECH et alii, 1984). A figura 2.6 ilustra esta técnica.

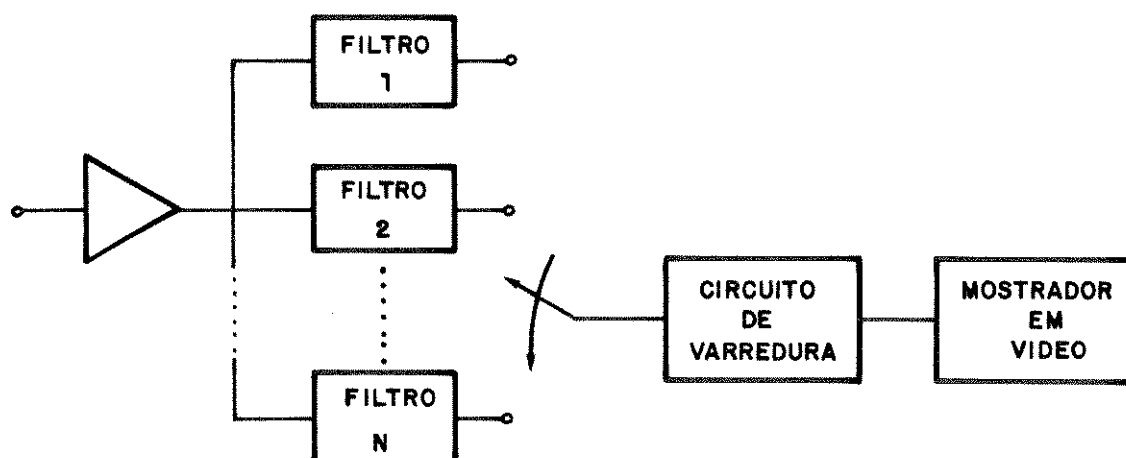


Fig. 2.6 - Analisador de espectro utilizando processamento paralelo.

Através de uma varredura rápida das amplitudes dos sinais de saída dos filtros, são obtidos histogramas da amplitude em função da frequência (ATKINSON e WOODCOCK, 1982).

Uma desvantagem desta técnica de processamento é a necessidade de um filtro específico para cada faixa de frequências a ser analisada, tornando o sistema complexo e volumoso. Além disso, sua inflexibilidade resulta de que alterações nos parâmetros dos filtros implicam em um novo projeto. O tempo de resposta do filtro também consiste em uma desvantagem deste método quando deseja-se a operação do sistema em tempo real (ZUECH et alii, 1984).

2.4.2 - Processamento digital de sinais

Consiste em um processo matemático, geralmente baseado na Transformada Rápida de Fourier (FFT) e implementado em computadores (SCHLINDWEIN e EVANS, 1989 ; DI GIULIOMARIA et alii, 1990). O maior problema para implementação desta técnica é a necessidade de processadores de alta velocidade para obtenção do processamento em tempo real. Em testes de laboratório realizados pelo autor, utilizando microcomputador compatível com IBM PC-XT ("clock" de 8 MHz), um tempo médio de processamento de 100 ms por análise foi necessário, com software em Turbo Basic (Borland-1987 V-1.0), o que descaracteriza o processamento em tempo real.

Atualmente encontram-se no mercado exterior processadores digitais de sinais com instruções dedicadas a este fim, que possibilitam o processamento em tempo real. Entretanto, seu custo e dificuldade de obtenção no mercado nacional no momento da definição deste trabalho, fizeram com que esta alternativa fosse abandonada.

2.3.4 - Filtros de varredura

Uma outra técnica para realização da análise espectral consiste no emprego do método heterodino de deslocamento do espectro da resposta Doppler (COGHLAN et alii, 1973). O produto do deslocamento é aplicado a um filtro passa faixa de banda fixa

(ATKINSON e WOODCOOCK, 1982). Este sistema apresenta a vantagem de utilizar um único módulo de filtro para todas as frequências a serem analisadas.

Existem dois métodos para se obter a análise espectral por meio de filtros de varredura (ATKINSON e WOODCOOCK, 1982):

- a - Fixando a banda do sinal e deslocando-se a faixa do filtro.
- b - Fixando a faixa do filtro e varrendo-se o espectro sobre ele.

As figuras 2.7. a e 2.7. b ilustram estes métodos:

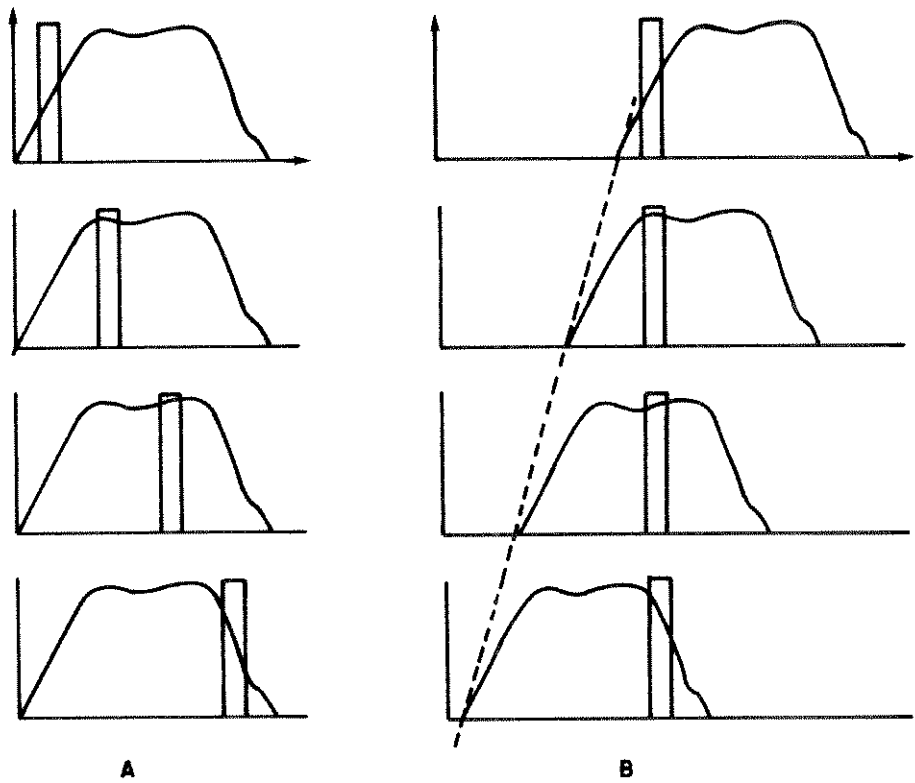


Fig. 2.7 - (a) Varredura da faixa do filtro sobre o espectro;
(b) Varredura do espectro sobre a faixa do filtro
(adaptado de Atkinson e Woodcoock, 1982).

Ambos os métodos acima citados apresentam a vantagem de utilizarem um único filtro, o que reduz a complexidade e custo do sistema, porém não permitem sua aplicação direta na construção de analisadores de espectro, uma vez que necessitam um sinal estacionário durante a análise, e o sinal Doppler apresenta rápidas variações espectrais durante o ciclo cardíaco, necessitando alterações para permitir sua aplicação em sistemas que operem em tempo real (ATKINSON e WOODCOCK, 1982).

A opção de se construir um filtro de banda variável resulta no desenvolvimento de circuitos complexos e de controle extremamente difícil. Em razão disto, Atkinson e Woodcock (1982) sugerem a varredura do espectro do sinal sobre um filtro de frequência fixa.

A maneira mais simples de se implementar esta varredura consiste em multiplicar o espectro Doppler pelo sinal de saída de um oscilador controlado por tensão (VCO). Neste caso, o incremento na frequência de oscilação determinará o deslocamento do espectro, ou seja, as baixas frequências na saída do VCO promovem pequenos deslocamentos do espectro enquanto altas frequências promovem grandes deslocamentos.

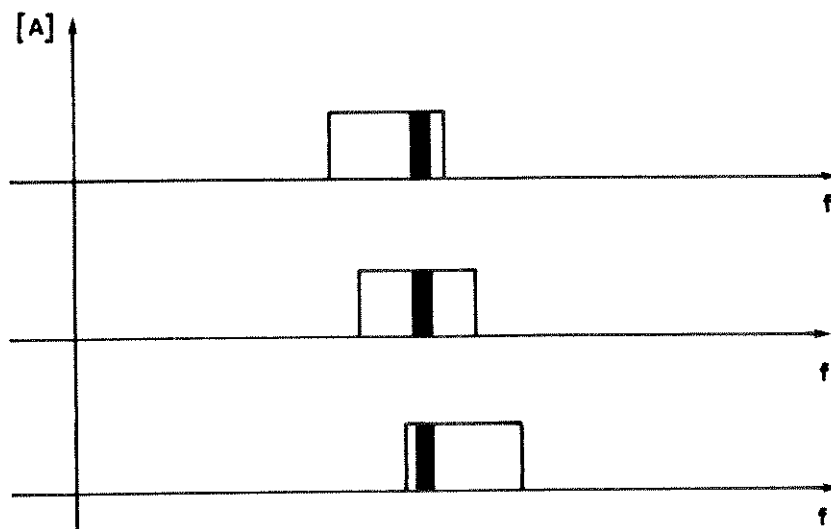


fig.2.8 - Deslocamento do espectro através de batimento heterodino.

Este método, entretanto, necessita filtros com banda passante muito estreita e rápido tempo de resposta para permitir operação em tempo real, o que inviabiliza sua construção (ZUECH, et alii, 1984).

Para análise da amplitude de um sinal com frequência f , faz-se necessário um tempo mínimo de $1/f$, concluindo-se que, quanto maior a frequência menor o tempo gasto na análise. Para o caso de análise da componente de frequência de 150 Hz, o tempo gasto será:

$$T = \frac{1}{150 \text{ Hz}} = 666 \text{ ms}$$

Expandindo este cálculo para todas as componentes do espectro de frequência sob análise, torna-se necessário um tempo mínimo de 50 ms, no caso de 29 componentes de frequência (número utilizado no analisador desenvolvido, conforme apresentado no capítulo 3 deste trabalho) o que inviabiliza o processamento em tempo real.

Através da técnica de compressão no tempo ("Time-compression") proposta por Coghlan et alii (1973), é possível incrementar a banda do sinal e assim reduzir o tempo de análise das componentes de frequência. Um fator de incremento apropriado permitirá a análise em tempo real.

Este processo consiste em adquirir o sinal a uma taxa T e recuperá-lo à taxa N, onde $N \gg T$, obtendo uma razão de deslocamento N/T (HERMINI e CALIL, 1990), produzindo uma expansão na banda do sinal e com isso reduzindo o tempo necessário à análise de cada componente de frequência.

Um sistema básico de "time-compression" desloca todo espectro de frequência de um fator determinado, denominado M. Após deslocado, o espectro é submetido a um filtro de varredura, como descrito anteriormente. Como exemplo, para $M=100$, tem-se:

F_{in}	T_{an}	M	F_{out}	T_{an}'
150 Hz	6,66 ms	100	15.000 Hz	66 μs
1000 Hz	1,00 ms	100	100.000 Hz	10 μs
3000 Hz	0,33 ms	100	300.000 Hz	3.3 μs

Tab. 2.1 - Relação entre frequências e tempo de análise.

Onde:

F_{in} = Frequência do sinal

T_{an} = tempo necessário à análise do sinal

F_{out} = Frequência do sinal após deslocamento

T_{an}' = Tempo necessário à análise do sinal deslocado

CAPITULO 3 - METODOLOGIA

3.1 - Introdução

A técnica utilizada no sistema desenvolvido é baseada em "Time-Compression", descrito por Coghlan et alii (1973), modificada para produzir uma varredura do espectro do sinal sobre um filtro passa faixa de banda fixa.

De acordo com a proposta inicial deste trabalho, o sistema desenvolvido apresenta as seguintes características:

- Análise de 29 componentes de frequência entre 150 e 3000 Hz, divididas em 3 faixas de resolução :
 - 150 a 800 Hz resolução de 50 Hz
 - 900 a 1.600 Hz resolução de 100 Hz
 - 1.800 a 3.000 Hz resolução de 200 Hz
- Apresentação gráfica em vídeo 3 D - tempo (x), frequência

- (y) e amplitude (z - escala cinza);
- Quatro níveis de amplitude em escala de cinza;
- Análise das 29 componentes em 17,6 ms;
- Opção de seleção e impressão dos gráficos desejados.

Visando reduzir a complexidade e os custos do equipamento, decidiu-se por realizar uma varredura digital do espectro, o que elimina a necessidade do estágio do filtro de varredura por multiplicação de frequências (heterodino), que além de apresentar-se crítico em razão da instabilidade do oscilador analógico, necessita componentes de alta velocidade de estabilização e produz um estreitamento da banda analisada com o aumento da frequência, segundo testes de bancada realizados pelo autor.

A modificação implementada na técnica de "time-compression" para produzir a varredura digital, consiste em variar, de forma controlada, a taxa de recuperação do sinal, por meio de um sistema de processamento digital por hardware (item 3.6). Se o sinal for adquirido a taxa T [Hz] constante, e recuperado a taxa N [Hz], agora variável, o coeficiente de deslocamento ($D=T/N$) será também variável, produzindo um deslocamento do espectro sobre o filtro, como apresentado na figura 3.1:

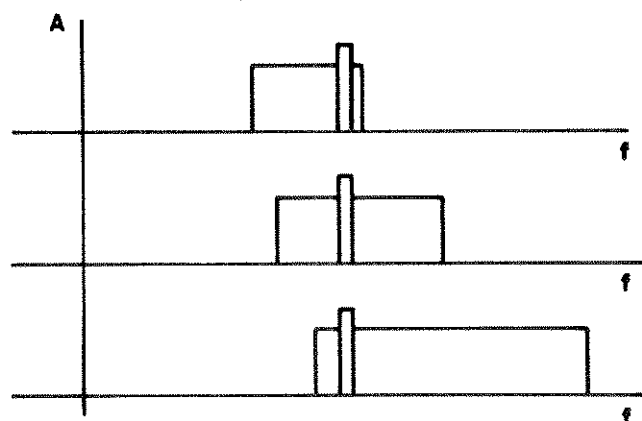


Fig 3.1 - Deslocamento do espectro sobre o filtro passa faixa utilizando varredura digital por "time-compression"

A cada incremento do coeficiente D , uma faixa de frequência do espectro é deslocada para a faixa de frequência do filtro e medida a amplitude deste sinal. A varredura de todo o espectro sobre o filtro permite o levantamento da amplitude de cada componente de frequência de interesse do sinal. A partir deste ponto o programa desenvolvido neste trabalho compõe um gráfico tridimensional em vídeo, apresentando as componentes de frequência no eixo das ordenadas, o tempo no eixo das abcissas e a amplitude de cada componente representada através de uma escala de cinza. É possível através, do teclado, interromper a aquisição, e solicitar a impressão do gráfico desejado.

Neste trabalho, foi definida a frequência do filtro passa-faixa em 70 kHz, em razão do compromisso entre os tempos de resposta dos componentes eletrônicos (amplificadores operacionais, conversores AD e DA) e a realização do processamento em tempo real.

Os valores do coeficiente D , que determinam o fator de

multiplicação do espectro, são obtidos dividindo-se a frequência do filtro pela frequência da componente a ser analisada. Por exemplo, na análise da amplitude da componente de 200 Hz, tem-se:

$$D = \frac{f_c}{f_a} = \frac{70.000 \text{ Hz}}{200 \text{ Hz}} = 350 \quad 3.1$$

onde:

D = Fator de deslocamento do espectro.

f_c = Frequência central do filtro passa faixa.

f_a = Frequência central da faixa a ser analisada.

Este procedimento é expandido a todas as componentes do espectro (Tabela 3.1; item 3.5).

A metodologia utilizada para implementação desta técnica será descrita a seguir.

3.2 - Descrição do Sistema

O sistema desenvolvido é composto por sete módulos:

- 1 - Módulo de condicionamento de sinais;
- 2 - Módulo de conversão analógico-digital (CAD)
- 3 - Módulo de processamento digital por hardware;
- 4 - Módulo de processamento analógico
- 5 - Interface paralela
- 6 - Programa de aquisição e apresentação dos resultados
- 7 - Módulo de fonte de alimentação

O diagrama em blocos do sistema está representado na figura

3.2.

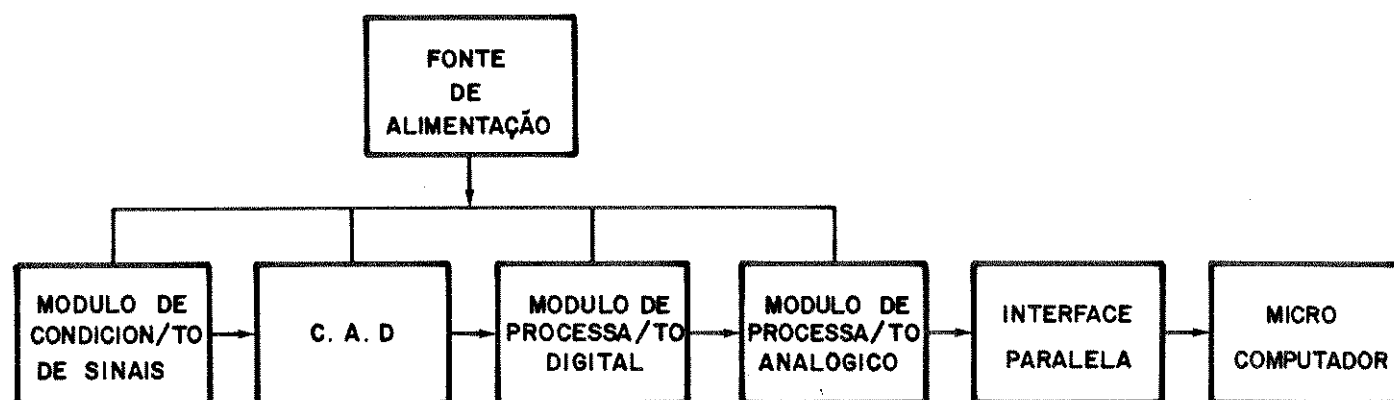


Fig. 3.2 - Diagrama em blocos do analisador de espectro.

Nos anexos são apresentados os diagramas esquemáticos destes módulos, bem como o programa desenvolvido para aquisição dos dados e mostragem em vídeo.

3.3 - Módulo de Condicionamento de Sinais.

Dadas as características do analisador desenvolvido, este pode processar sinais provenientes de qualquer equipamento Doppler de onda contínua que apresentem em sua saída de audio, sinais na faixa de 150 a 3000 Hz. Em razão da grande diversidade de amplitudes e frequências destes sinais, faz-se necessário limitar

estes parâmetros, compatibilizando-os com o módulo CAD. O condicionamento dos sinais é realizado por um amplificador de ganho ajustável entre 0 e 10, através de um potenciômetro no painel frontal, limitando os sinais a faixa de amplitude até $1.28 V_p$ e um filtro passa-faixa, o que permite a passagem de sinais entre 150 e 3000 Hz, com a finalidade de eliminar o efeito de "aliasing" na digitalização dos sinais (ATKINSON e WOODCOCK, 1982).

3.3.1 - Amplificador de entrada

O amplificador de entrada foi implementado a partir de amplificadores operacionais de baixo custo, uma vez que os níveis de tensão e faixa de frequência destes atendem perfeitamente os requisitos do sistema, apresentados no início deste item

Para se obterem os ganhos escolhidos com alta impedância de entrada, foi utilizado um isolador ("buffer") nesse estágio, uma vez que a fase não é um parâmetro importante para a análise, em razão do analisador fornecer apenas informações da amplitude das componentes de frequência do sinal sem resolução de sentido (capítulo 5). Em seguida, um estágio inversor para ajuste do ganho na faixa especificada foi introduzido. O circuito final está representado na figura 3.3.

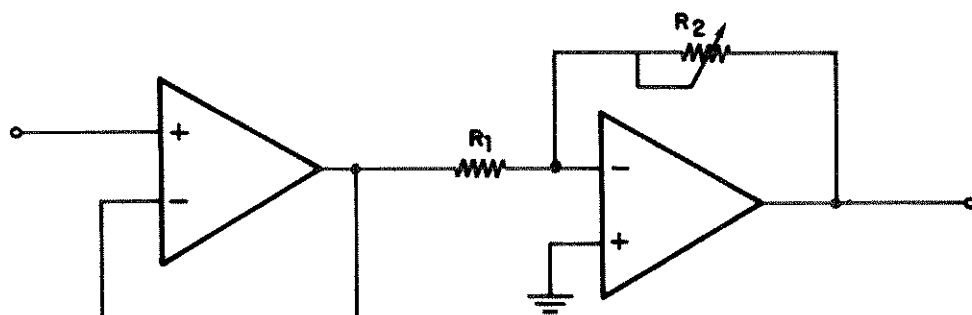


Fig. 3.3 - Amplificador de entrada.

3.3.2 - Filtro passa faixa.

Para obtenção de um filtro passa faixa de resposta plana entre 150 e 3000 Hz, optou-se pela associação em cascata de um filtro passa-altas com um passa-baixas, do tipo Butterworth. A escolha deste tipo de filtro deu-se em razão da melhor resposta em frequência apresentada por este tipo de filtro (STOUT e KAUFMAN, 1976).

Para o projeto dos filtros, utilizou-se a metodologia descrita por Stout e Kaufman (1976), projetando-se inicialmente o filtro passa-baixas, com os seguintes parâmetros:

$f_c = 3000 \text{ Hz}$

$A_v = 1,0$

tipo = BUTTERWORTH

N. de polos = 3

A configuração escolhida está representada na figura 3.4:

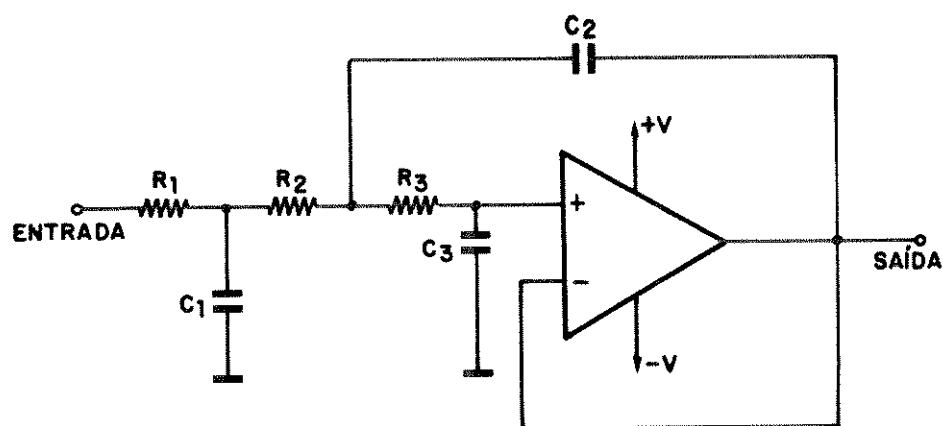


Fig. 3.4 - Diagrama do filtro passa-baixas.

A partir de valores obtidos em tabela da referência acima citada, foram determinadas as seguintes constantes:

$$C''_1 = 1,392$$

$$C''_2 = 3,546$$

$$C''_3 = 0,2024$$

Nesta etapa são calculadas os valores das coeficientes que

determinarão a frequência de corte do filtro:

$$C_1' = \frac{C''_1}{2 \pi f_c} = \frac{1,392}{6000 \pi} = 7,385 \times 10^{-5}$$

$$C_2' = \frac{C''_2}{2 \pi f_c} = \frac{3,546}{6000 \pi} = 18,81 \times 10^{-5}$$

$$C_3' = \frac{C''_3}{2 \pi f_c} = \frac{0,2024}{6000 \pi} = 1,074 \times 10^{-5}$$

estipulando um valor para C_2 :

$$C_2 = 10 \text{ nF}$$

A partir dos coeficientes C'_x , serão calculados R , C_1 e C_3 :

$$R = \frac{C_2'}{C_2} = \frac{18,81 \cdot 10^{-5}}{10 \cdot 10^{-9}} = 18,81 \text{ k}\Omega$$

$$C_1 = \frac{C_1'}{R} = \frac{7,385 \cdot 10^{-5}}{7.500} = 9,84 \text{ nF} = 9,8 \text{ nF}$$

$$C_3 = \frac{C_3'}{R} = \frac{1,074 \cdot 10^{-5}}{7.500} = 1,43 \text{ nF}$$

desta forma tem-se :

$$R_1 = R_2 = R_3 = 18,2 \text{ k}\Omega$$

$$C_1 = 3,9 \text{ nF}$$

$$C_2 = 10 \text{ nF}$$

$$C_3 = 560 \text{ pF}$$

Como a atenuação desejada é de 120 dB/década, a fim de permitir a discriminação das frequências a serem analisadas, são

necessários 6 polos na frequência de corte, o que levou a conexão de 2 células em série, uma vez que cada uma possui 3 polos.

De maneira análoga, foi projetado o filtro passa altas utilizando a configuração apresentada na figura 3.5:

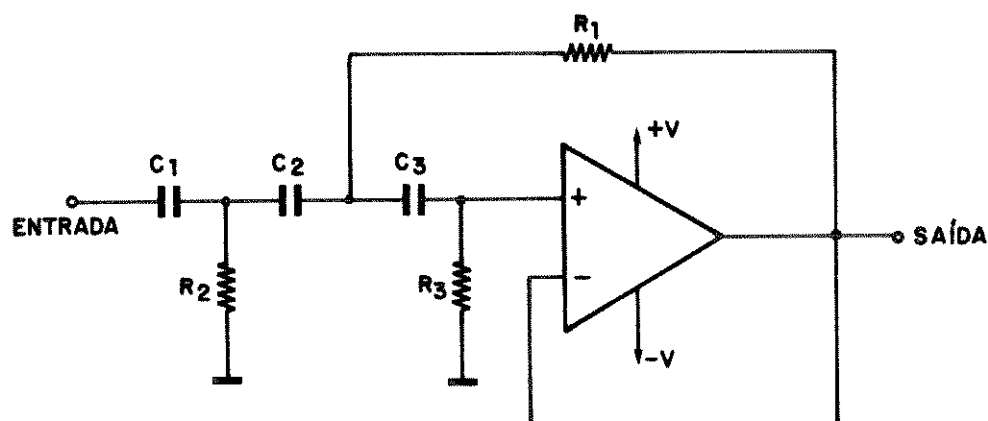


Fig. 3.5 - Filtro passa altas.

Os valores dos parâmetros desejados para este filtro são os seguintes:

$$f_c = 150 \text{ Hz}$$

$$C_x = 100 \text{ nF}$$

Tipo = Butterworth

$$N. \text{ de polos} = 3$$

Os coeficientes R'_x de tabela são os seguintes:

$$R'_1 = 0,2820$$

$$R'_2 = 0,7184$$

$$R'_3 = 4,941$$

A seguir, é calculado o valor do coeficiente que determina

a frequência de corte do filtro

$$C = \frac{1}{2 \pi f_c} = \frac{1}{2 \pi \cdot 150} = 1,061 \cdot 10^{-3}$$

Uma vez escolhido o valor para os capacitores, o fator de multiplicação para escolha dos resistores será:

$$K = \frac{C}{C_1} = \frac{C}{C_2} = \frac{C}{C_3} = \frac{1,061 \times 10^{-3}}{100 \cdot 10^{-9}} = 1,061 \cdot 10^4$$

Os valores dos resistores serão:

$$R_1 = K R'_1 = 1,061 \times 10^4 \cdot 0,2820 = 2.992 \, \Omega = 3,3 \, k\Omega$$

$$R_2 = K R'_2 = 1,061 \times 10^4 \cdot 0,7184 = 7.622 \, \Omega = 7,68 \, k\Omega$$

$$R_3 = K R'_3 = 1,061 \times 10^4 \cdot 4,941 = 52.424 \, \Omega = 52,3 \, k\Omega$$

Assim como para o filtro passa baixas, serão colocados 2 células em série visando obter a atenuação desejada, ou seja, 120 db/década.

O diagrama em blocos deste módulo é mostrado na figura 3.6:



Fig. 3.6 - Diagrama em blocos do condicionador de sinais.

O diagrama esquemático deste módulo encontra-se no anexo I.

3.4 - Módulo de Conversão AD.

Segundo testes de bancada realizados, para recuperação de um sinal digitalizado é necessária uma frequência de amostragem três vezes superior à máxima frequência deste sinal. Estes resultados confirmam aqueles publicados por Coghlan et alii, (1973).

Como mencionado anteriormente, são analisados sinais com frequência até 3 kHz, Foi escolhida uma frequência de amostragem de 10 kHz, que além de atender a exigência acima, é facilmente obtida a partir da divisão da frequência de osciladores a cristal comercialmente disponíveis.

Este módulo é composto por 2 circuitos como mostra a figura 3.7:

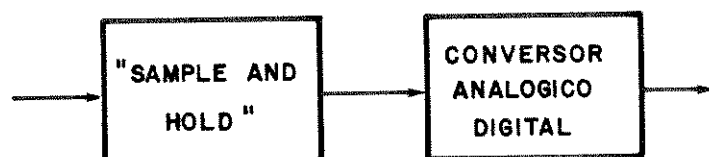


Fig. 3.7 - Módulo de conversão AD.

O circuito amostrador tem por função manter o nível de tensão estável durante o período de conversão, atualizando este

nível a cada nova amostragem, como ilustrado na figura 3.8:

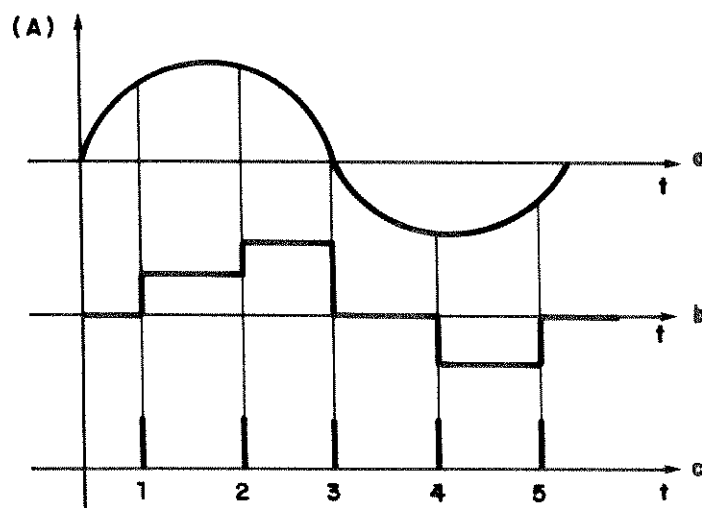


fig. 3.8 - (a) sinal analógico, (b) sinal amostrado,
(c) intervalos de amostragem.

O conversor AD deve realizar a conversão entre cada intervalo de amostragem, quando os dados já encontram-se estáveis.

Foi escolhido o circuito integrado (CI) AD 582 da Analog Devices (1982) para operar como "sample and hold", o qual atende as necessidades do projeto, pode ser adquirido no mercado nacional, e possui baixo custo. Este CI apresenta as seguintes características:

- Impedância de entrada 30 M Ω
- Variação de "off-set" em toda faixa de temperatura (10 a 50 °C) - 4 mV

- Tempo de aquisição - 6 μ s para 0.1%
- Tempo para estabilização com 20 V_{p-p} - 500 ns

Como conversor AD foi escolhido o CI AD670 da Analog Devices (1982), o qual apresenta as seguintes características:

- Conversão por aproximações sucessivas
- 8 bits de resolução
- 10 μ s de tempo de conversão

A escolha deste conversor deu-se por apresentar a melhor opção entre tempo de conversão, resolução, disponibilidade no mercado e custo.

O modo de operação do conversor AD é determinado pelo nível lógico colocado nas seguintes entradas de controle:

- BPO/ $\overline{\text{UPO}}$ - determina se a operação é unipolar ou bipolar.
- FORMAT - determina se a saída é binário simples ou complemento de dois.

A faixa de operação é definida pela configuração do amplificador diferencial de entrada do conversor. Como neste projeto o conversor AD está programado para operar no modo bipolar com saída binária, as tensões de entrada podem ser de ± 128 mV, com resolução de 1 mV ou $\pm 1,28$ V, com resolução de 10 mV. A última opção foi escolhida por apresentar menor susceptividade a ruídos.

A figura 3.9, mostra a sequência dos sinais de controle que atuam neste módulo:

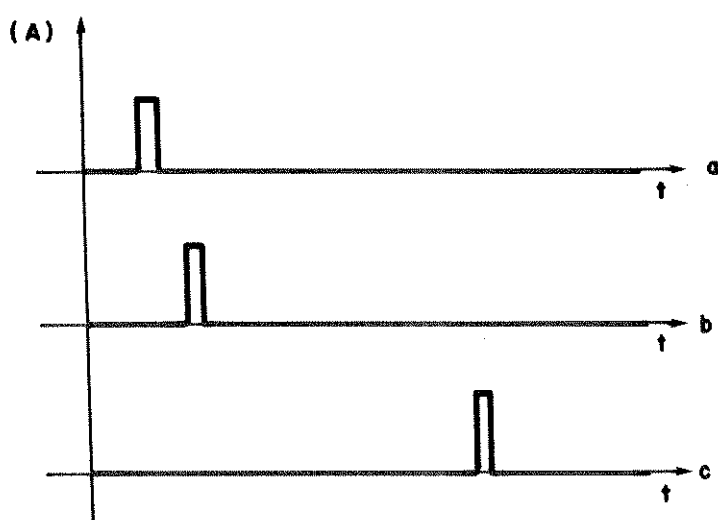


Fig. 3.9 - (a) sinal de SH, (b) sinal de RW, (c) sinal de STATUS.

Após amostrado e estável o sinal, o conversor AD recebe um sinal para início de conversão (RW). Aproximadamente $10 \mu s$ após este sinal, o CI fornece os 8 bits referentes à digitalização do nível de tensão da entrada e em seguida envia o sinal de fim de conversão (STATUS), avisando que o dado na saída do conversor AD já se encontra disponível para leitura.

O diagrama esquemático deste módulo é apresentado no Anexo II.

3.5 - Módulo de Processamento Digital.

Como mencionado na introdução deste capítulo, é realizada uma expansão na banda do sinal, através da variação do valor do coeficiente de deslocamento D, visando a varredura de todas as componentes do espectro sobre um filtro passa-faixa.

Para permitir o processamento em tempo real, são utilizados 2 blocos de memória RAM estática, tipo 93L422 (FARCHILD, 1979); sendo que os mesmos funcionam simultaneamente, sendo um bloco como entrada (armazenamento do sinal digitalizado) e outro como saída (recuperação do sinal para análise), alternando estas funções periodicamente. Cada bloco de memória é composto por 256 bytes de 8 bits, cujos endereçamentos são fornecidos por contadores em anel.

Uma característica importante deste tipo de memória é a existência de barramento de entrada (escrita) e saída (leitura) independentes entre si, como mostra a figura 3.10. Esta característica diminui a complexidade do sistema, pois permite a conexão física do barramento de entrada ao conversor AD, e o de saída ao módulo de processamento. A habilitação do modo de operação da memória (entrada ou saída) é feita por um sinal de controle (R/W), conforme indicado na figura 3.10.

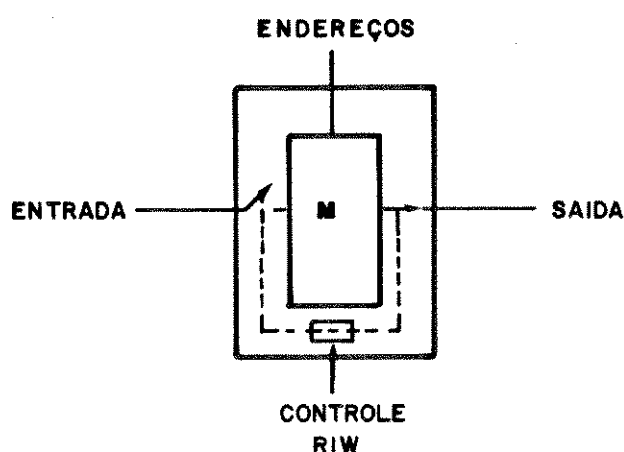


Fig.3.10 - Estrutura da memória RAM.

Em um determinado instante, o bloco selecionado como entrada armazena os dados provenientes do conversor AD e seu endereço é incrementado à frequência de amostragem deste conversor, neste caso 10 kHz. Simultaneamente o segundo bloco selecionado como saída fornece dados ao conversor DA, do módulo de processamento analógico, a uma frequência de recuperação (f_r), obtida a partir da divisão do sinal de um oscilador mestre de 14 MHz, através de um divisor programável (figura 3.12).

O diagrama em blocos da figura 3.11 mostra as memórias RAM e os contadores em anel que fornecem as taxas de armazenamento e recuperação dos dados:

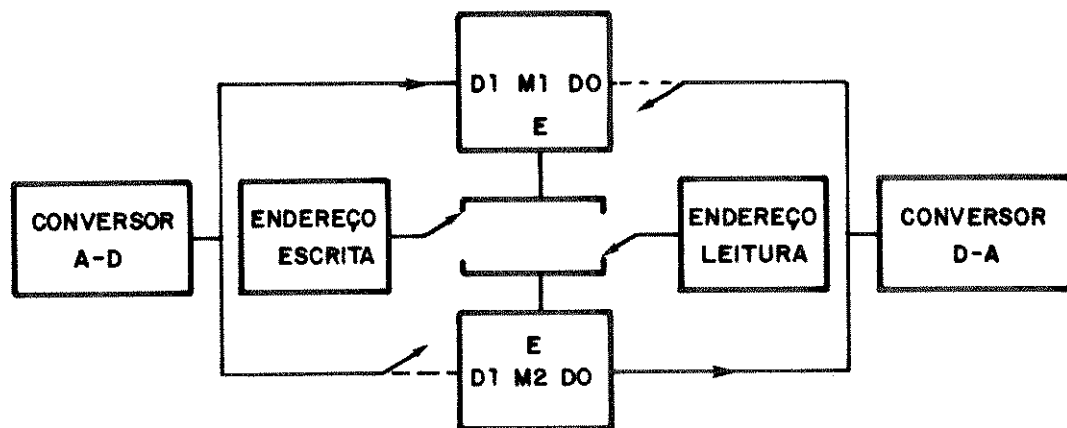


Fig. 3.11 - Diagrama em blocos do banco de memórias.

O fator de divisão da frequência do oscilador mestre é obtido utilizando o seguinte cálculo:

$$P = \frac{f_o}{D \times f_{am}} \quad 3.2$$

onde:

P = Fator de divisão

f_o = Frequência do sinal do oscilador mestre (14 MHz)

D = Fator de multiplicação do espectro

f_{am} = Frequência de amostragem (10 kHz)

Aplicando as equações 3.1 e 3.2 a todas as 29 faixas centrais das frequências a serem analisadas, é obtida a tabela 3.1 contendo os fatores de divisão (P) associados ao fator de deslocamento (D) e frequência a ser analisada (f_a [Hz]).

fa	P	D
150	3	466,68
200	4	350,00
250	5	280,00
300	6	233,33
350	7	200,00
400	8	175,00
450	9	155,55
500	10	140,00
550	11	127,27
600	12	116,66
650	13	107,69
700	14	100,00
750	15	93,33
800	16	87,5
900	18	77,77
1000	20	70,00
1100	22	63,63
1200	24	58,33
1300	26	53,84
1400	28	50,00
1500	30	46,66
1600	32	43,75
1800	36	38,88
2000	40	35,00
2200	44	31,81
2400	48	29,16
2600	52	26,92
2800	56	25,00
3000	60	23,33

Tab 3.1 - *Relação de deslocamento de frequência.*

Desta forma, existe um fator de deslocamento específico para cada faixa de frequência, sendo que uma análise das 29 faixas, realiza uma leitura completa da memória (256 "Bytes") em cada um dos fatores D apresentados na tabela 3.1.

Para implementação do divisor foram utilizados contadores programáveis (74F193), operando no modo decrescente ("down"). Neste modo, o sinal de entrada é dividido pelo valor carregado em

suas entradas de programação. Em razão dos fatores de divisão não serem sequenciais, estes valores são fornecidos por uma memória tipo PROM de 32 bytes de 8 bits (74S288), cujo endereço fornecido por um contador em anel é incrementado ao final da análise de cada faixa de frequência, conforme apresentado na figura 3.12.

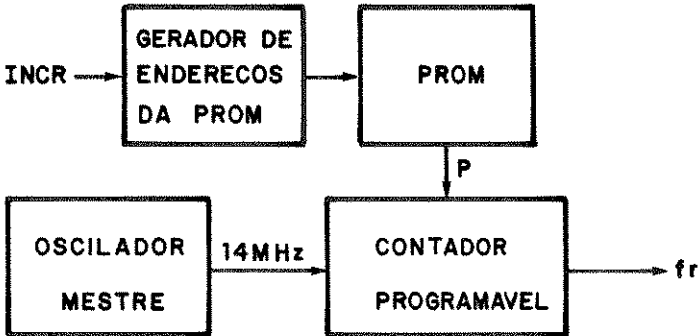


Fig.- 3.12 - Divisor programável do módulo de processamento digital.

O mapa com o conteúdo da memória PROM utilizada é mostrado no anexo III.

Para melhor entendimento da metodologia, serão tomados como exemplo a análise das componentes de 150 e 200 Hz.

De acordo com os dados apresentados na tabela 3.1, para análise da componente de 150 Hz é necessário carregar o valor 3 na entrada do divisor, a fim de que o contador em anel (que gera o endereço de recuperação) seja incrementado à frequência de 4.66

MHz (14 MHz : 3). Esta contagem abrange os valores entre 00 e FFh, para acesso a todos os endereços da memória PROM. Ao final da leitura da memória, é enviado um sinal de controle ao módulo de processamento analógico para informar o fim da análise da componente de 150 Hz e em seguida é carregado o valor 4 no divisor, valor este correspondente à análise da componente de 200 Hz.

O procedimento descrito acima é adotado para todas as faixas de frequência do espectro, até a análise de 3 kHz, quando é enviado ao microcomputador um sinal informando o fim do processamento daquele intervalo. Neste instante os blocos são alternados, isto é, o bloco que forneceu os dados para análise, passa a ser selecionado como entrada e será preenchido com novos dados, e o bloco preenchido com os dados digitalizados passa a operar como saída e o processo se reinicia com a análise da componente de 150 Hz.

O diagrama esquemático deste módulo esta representado no anexo IV

3.6 - Módulo de Processamento Analógico.

Este módulo é composto por quatro blocos:

- 1 - Conversor digital analógico (CDA)
- 2 - Filtro passa faixa de 70 kHz
- 3 - Detector de pico
- 4 - Conversor analógico - digital (CAD)

Após processados digitalmente (deslocamento do espectro), os dados devem ser reconvertidos para a forma analógica, a fim de comporem o sinal a ser aplicado ao filtro passa-faixa, permitindo a retirada somente dos sinais que se encontram na faixa de 70 kHz.

O conversor DA escolhido foi o CI AD DAC08 da Analog Devices pelas mesma razões apresentadas na escolha do CAD. As características deste componente que interessam para este projeto são as seguintes:

- Alta velocidade de estabilização - 85 ns;
- Resolução de 8 bits;
- Saída em corrente;
- Nível lógico de entrada compatível com TTL.

Como é necessário um sinal de tensão na saída do conversor, foram utilizados dois resistores como conversor corrente-tensão e um amplificador de instrumentação para fornecer uma saída bipolar, neste caso ± 10 V, recuperando a forma original do sinal. A figura 3.13 mostra a configuração do conversor DA.

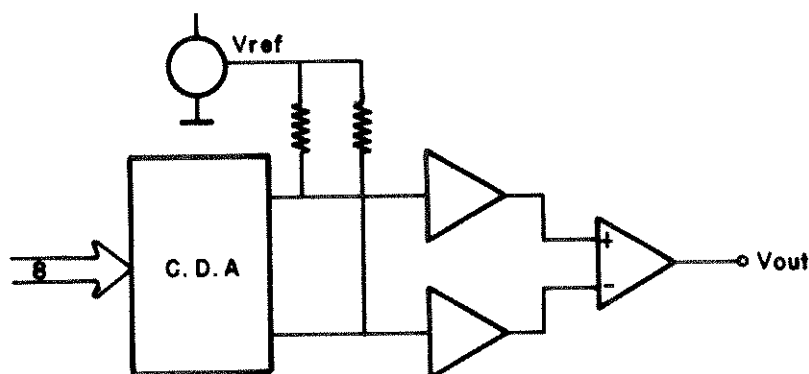


Fig.3.13 - Conversor DA.

Como mencionado na introdução deste capítulo, os sinais são aplicados a um filtro passa-faixa de 70 kHz, cuja atenuação é calculada visando permitir a passagem de uma faixa de frequências por análise.

Para determinação do fator de atenuação do filtro foram considerados os dois limites de frequências analisados pelo equipamento, ou seja, 150 e 3000 Hz.

A faixa centrada em 150 Hz tem como limite superior componentes de frequência inferiores a 175 Hz, a partir do qual são analisadas na faixa de 200 Hz. Para faixa centrada em 3000 Hz, o filtro deve permitir a passagem de sinais a partir de 2900 Hz,

uma vez que frequências inferiores a esta são compreendidos pela faixa de 2800 Hz. Os limites, inferior para a faixa de 150 Hz e superior para a de 3000 Hz não foram considerados em razão da simetria da banda do filtro passa-faixa, e do corte realizado pelo filtro do condicionador de sinais.

Uma vez que todas as componentes são deslocadas em frequência de acordo com o fator de deslocamento (D), segundo a equação 3.1 pode ser construída a tabela 3.2:

f_{in}	D	f_{out}
150	466,6	70
175	466,6	81,6
2900	23,3	67,6
3000	23,3	70

Tabela 3.2 - Frequência de saída em função do fator de deslocamento (D)

A partir de testes de bancada realizados , com resultados apresentados no capítulo 4, foram utilizadas quatro células de filtros passa-faixa do tipo realimentação múltipla visando obter a atenuação descrita na tabela 3.2. A configuração de cada célula do filtro é apresentada na figura 3.14 (Stout e Kaufman, 1976). Para projeto da célula, foi utilizada a metodologia descrita por Stout e Kaufman (1976).

Os valores da frequência central do filtro (f_0), do fator de qualidade (Q) e do ganho na frequência f_0 (H), são:

$$f_0 = 70 \text{ kHz}$$

$$Q = 10$$

$$H = 1$$

É escolhido um valor prático para os capacitores:

$$C = 470 \text{ pF}$$

É calculado o valor do resistor R_3 para a frequência de corte:

$$R_3 = \frac{2Q}{2\pi f_0 C} = \frac{2 \cdot 10}{2\pi \cdot 70 \cdot 10^3 \cdot 470 \cdot 10^{-12}} = 98,7 \text{ kHz}$$

O valor de R_1 vem de :

$$R_1 = \frac{Q}{2\pi \cdot f_0 \cdot C \cdot H} = \frac{10}{2\pi \cdot 70 \cdot 10^3 \cdot 470 \cdot 10^{-12} \cdot 1} = 48,4 \text{ kHz}$$

O valor de R_2 vem de :

$$R_2 = \frac{Q}{(2\pi \cdot f_0 \cdot C) (2Q^2 - H)} = \frac{10}{(2\pi \cdot 70 \cdot 10^3 \cdot 470 \cdot 10^{-12}) (2 \cdot 10^2 - 1)}$$

$$R_2 = 249 \Omega$$

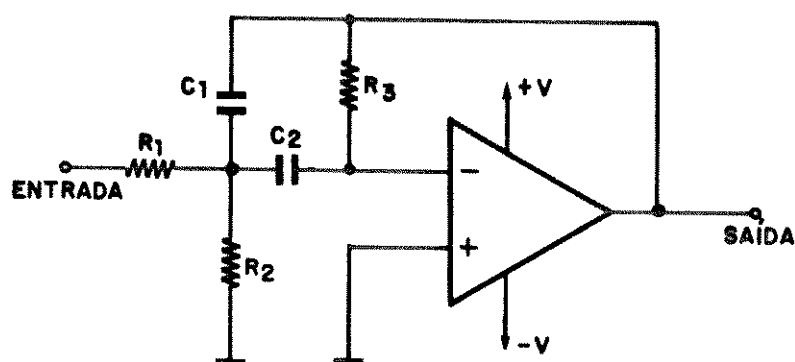


Fig.3.14 - Célula do filtro passa faixa.

Após a análise de cada faixa de frequência do espectro, é necessário o registro da sua amplitude. Isto é feito através de um circuito detector de pico baseado em Horowitz e Hill (1980), com a introdução de um circuito de "reset" a fim de descarregar completamente o capacitor, e permitir o armazenamento da amplitude da próxima componente. O circuito desenvolvido está representado na figura 3.15:

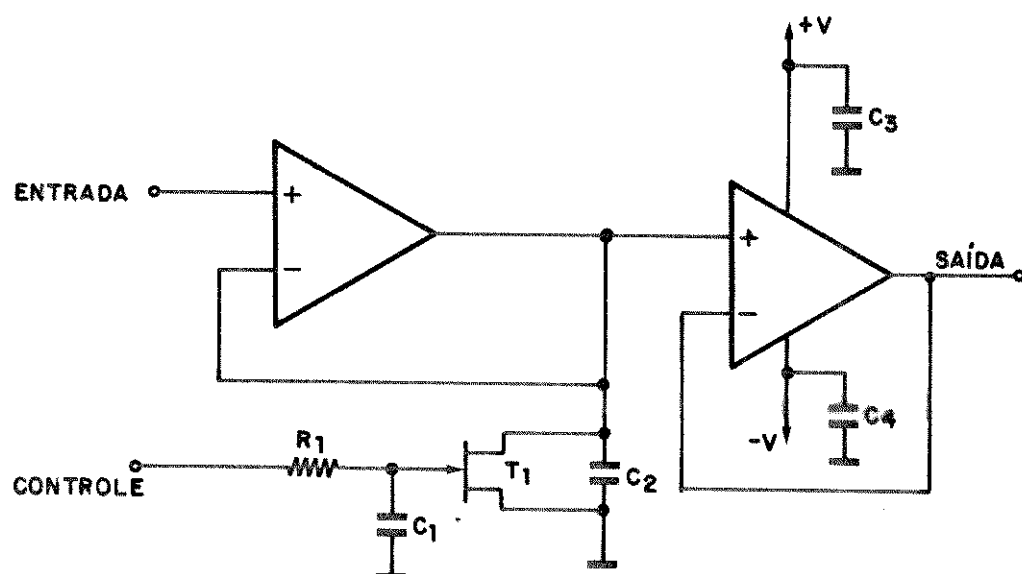


Fig. 3.15 - Circuito do detector de pico.

A saída do detector de pico é aplicada à uma rede de três comparadores de alta velocidade, dispostos de maneira a compor um conversor AD paralelo (figura 3.16), cuja saída fornece um "byte" de três "bits" informando ao microcomputador a amplitude da componente de frequência analisada.

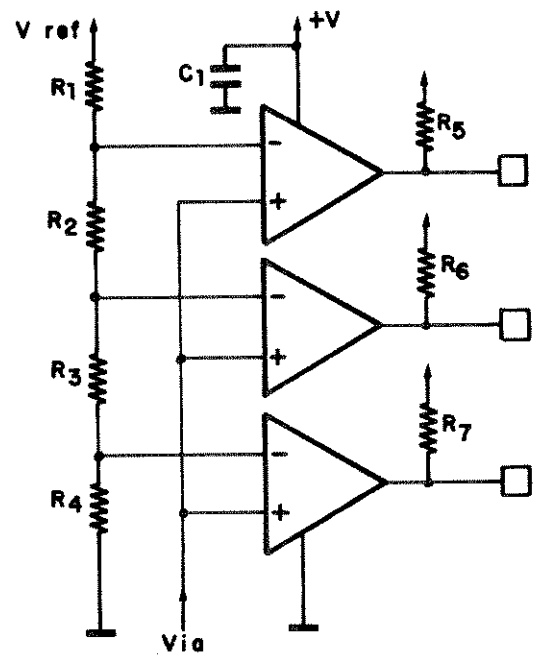


Fig. 3.16 - Conversor AD paralelo.

O diagrama esquemático deste módulo é apresentado no Anexo V.

3.7 - Interface Paralela.

O Analisador de Espectro fornece, após cada varredura, a

amplitude referente a uma determinada faixa de frequência da resposta Doppler do sinal ultra-sônico. Estas amplitudes são fornecidas de maneira sequencial num total de 29 a cada 17,6 ms.

Representando o que foi descrito num espectro de linha:

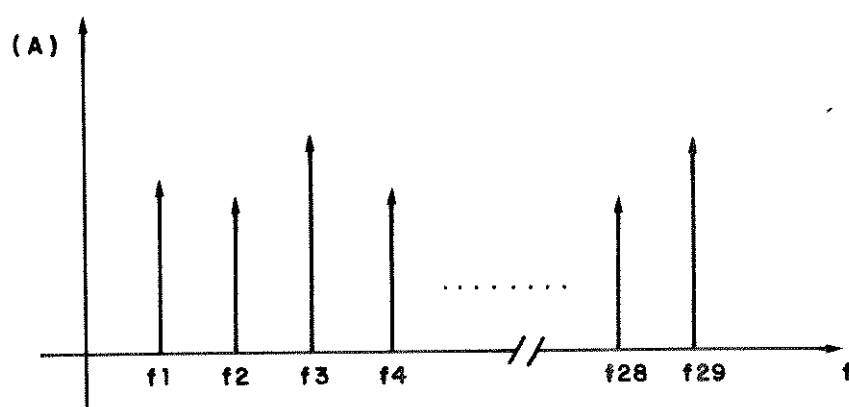


Fig. 3.17 - Representação do espectro em gráfico de linhas.

O equipamento desenvolvido fornece a amplitude da componente de frequência f_1 do sinal, em seguida a de f_2 e assim sequencialmente até f_{29} , quando então a contagem se reinicia a partir de f_1 .

Cada vez que um dado estiver disponível para leitura, um sinal de controle (EOC - "End Of Conversion") irá a nível baixo ("0"). Este sinal é usado para sincronizar a leitura do dado pelo microcomputador.

O Analisador também fornece um sinal de controle que irá a nível alto (1) quando o dado a ser lido for referente à amplitude f_{29} . Este sinal serve para sincronizar a apresentação com o analisador e foi denominado UC (Última Componente).

O desenvolvimento da placa de interface paralela está descrito no anexo VI.

3.8 - Programa de aquisição e apresentação dos resultados.

O programa desenvolvido visa atender às seguintes necessidades:

- a-) Cada dado é um código de 3 bits disponível para leitura na porta A da Porta paralela;
- b-) Os dados são lidos e processados em tempo real, ou seja, a uma frequência próxima de 10 KHz;
- c-) O processamento do dado consiste em armazená-lo na memória, classificá-lo em 4 níveis, atribuir a cada nível uma tonalidade de cinza e mostrar, em escala cinza, o resultado do processamento.
- d-) Cada linha da tela corresponde a uma faixa de frequência (f_1 , f_2 , ..., ou f_{29}) e cada coluna da tela a um intervalo de tempo.

A necessidade "a" implicou na inclusão de comandos no programa para programar a porta paralela no modo desejado, conforme descrito no Anexo VI.

A necessidade "b" determinou a linguagem de programação a ser utilizada. Optou-se por fazer a inicialização do modo gráfico, a programação da porta paralela, o controle de variáveis e a mostragem de "menus" na linguagem C. Isto porque esta linguagem é bastante próxima da linguagem de máquina, permitindo o uso de programas em Assembly como simples funções e possui comandos e

funções para a manipulação de textos e gráficos.

A necessidade "d"- determinou que a tela deveria ser preenchida de forma que o resultado final fosse o apresentado na tabela 3.3

		Colunas (tempo)					
		01	02	03	78	79	80
linhas (componentes de frequência)	01 (F29)	*	*	*	*	*	*
	02 (F28)	*	*	*	*	*	*
	03 (F27)	*	*	*	*	*	*
	27 (F03)	*	*	*	*	*	*
	28 (F02)	*	*	*	*	*	*
	29 (F01)	*	*	*	*	*	*

Tabela 3.3 - Forma de preenchimento da tela.

A aquisição, classificação, apresentação e armazenamento dos dados é realizada em "Assembly", com parâmetros fornecidos pelo programa em C. Isto foi feito devido ao fato de que um programa é mais rápido em "Assembly" do que em qualquer outra linguagem de programação.

A parte do programa que não necessita de grande velocidade foi desenvolvida em C e a parte que deve ser rápida em "Assembly".

Para a confecção da parte do programa em "Assembly" utilizou-se o montador MASM versão 4.0 da Microsoft. Para a parte em C utilizou-se o compilador TURBO C versão 1.5 da Borland, o qual além de prover funções para manipulação da tela em alta resolução, também possibilitou a junção da parte em "Assembly" com a parte em C, resultando em um único programa final.

Os 4 níveis fornecidos pelo conversor AD do módulo de processamento analógico, foram determinados da seguinte forma:

Se dado = 0h então = nível 0;

Se dado = 1h então = nível 1;

Se dado = 2h então = nível 2;

Se dado = 4h então = nível 3.

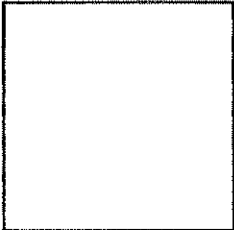
Conforme mencionado anteriormente, para cada nível associou-se uma tonalidade de cinza. Para se conseguir tal efeito, utilizou-se a tela de alta resolução do micro. Nesta tela de 640x200 pixels, cada pixel está diretamente associado a um bit da memória gráfica, ou seja, se o bit contiver o valor 1, então o

pixel correspondente estará sendo mostrado (estará aceso) e se contiver o valor 0, não será visível (estará apagado). Sendo assim, deve-se ter na memória gráfica mais bits 0 quando a tonalidade desejada estiver mais próxima do preto (fundo de tela) e mais bits 1 quando estiver mais próxima do branco.

Para representação de 29 faixas de frequência em 200 linhas, cada componente deve ocupar 4 linhas e 8 colunas para cada componente. A escolha de 8 colunas para cada dado deu-se pelo fato de que na memória gráfica tem-se em cada endereço a determinação de 8 pontos em 8 colunas adjacentes. Sendo assim, basta enviar um byte para o endereço correto para que uma linha do caracter 4x8 resultante fosse especificada. Como o caracter possuía 4 linhas, para especificá-lo por completo bastou associar a cada caracter 4 bytes. Isto foi feito para 4 caracteres, cada qual associado a um nível, conforme apresentado na tabela 3.4

Nível 0:

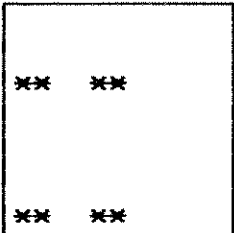
00000000
00000000
00000000
00000000



Caracter preto na tela ou
branco na impressora

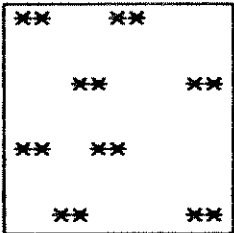
Nível 1:

00000000
11001100
00000000
11001100



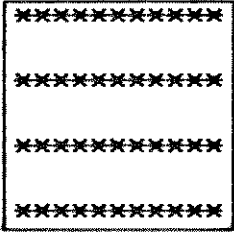
Nível 2:

11001100
00110011
11001100
00110011



Nível 3:

11111111
11111111
11111111
11111111



Caracter branco na tela
ou preto na impressora

Tab. 3.4 - Caracteres de vídeo para composição da escala cinza.

Conforme descrito anteriormente, a memória gráfica consiste de um segmento com os "offsets" de 0000 a 1F3Fh associados às linhas pares do vídeo e os "offsets" de 2000h a 3F3Fh às linhas

ímpares. Assim tem-se a seguinte representação da tela na memória, segundo a tabela 3.5:

Colunas	0	1	2	...	77	78	79
linha 000	0000h	0001h	0002h		0047h	0048h	0049h
linha 001	2000h	2001h	2002h		2047h	2048h	2049h
linha 002	0050h	0051h	0052h		0097h	0098h	0099h
linha 003	2050h	2051h	2052h		2097h	2098h	2099h
linha 198	1EF0h	1EF1h	1EF2h		1F37h	1F38h	1F39h
linha 199	3EF0h	3EF1h	3EF2h		3F37h	3F38h	3F39h

offsets no segmento B800h

offset XXXX: pixel 1 pixel2 pixel 8

Tab. 3.5 - Representação da tela na memória do microcomputador.

No processamento do dado, este deve ser classificado segundo seu valor, atribuindo um caracter de escala cinza, e depois colocado na memória gráfica, obedecendo a configuração desta.

O preenchimento da tela foi iniciado na primeira coluna pela parte inferior (end.3EF0h) em direção à parte superior até chegar à faixa de frequência f29, quando então reinicia na parte inferior da coluna seguinte (end. 3EF1h) e assim por diante.

Para o armazenamento dos dados escolheu-se o segmento 5000h

da memória RAM do microcomputador. Armazenou-se em cada "offset" deste segmento um dado lido. Quando o segmento estava completo, o preenchimento era reiniciado a partir do offset 0000. Com isto foi possível armazenar os dados equivalentes a $\{\text{int } 65536 / (29 \times 80)\} = 16$ telas, ou melhor, os dados equivalentes às últimas 16 telas mostradas.

No final de cada tela as variáveis tinham de ser atualizadas pelo programa em C, causando a interrupção na leitura e consequente perda de dados. Sendo assim, tornou-se necessária a elaboração de um processo que assegurasse que a tela começasse a ser preenchida com o dado relativo à componente F1.

Para tornar isto possível introduziu-se no programa um "looping" que é executado antes do início do preenchimento de cada tela, enquanto o sinal da última componente (UC) não fosse recebido. Isto significa que após o preenchimento da tela, o programa interrompe a leitura de dados, atualiza as variáveis e fica em "looping" até que a última componente de frequência esteja disponível. Em seguida reinici-se a aquisição com a certeza de que o próximo dado lido será relativo a F1, o qual será colocado no canto inferior esquerdo do vídeo, iniciando o preenchimento de nova tela. Com isto perde-se uma coluna inteira enquanto o programa fica esperando o sinal UC. Porém, isto não tem importância uma vez que uma coluna no início de cada tela não altera a morfologia do gráfico produzido.

No Anexo VII são apresentados os fluxogramas e listagens dos programas desenvolvidos.

3.9 - Fonte de alimentação.

Para fornecer as tensões de alimentação necessárias ao funcionamento dos circuitos, foi desenvolvido um conjunto de três fontes de alimentação com regulação linear. Este tipo de fonte foi escolhido em razão de utilizar circuitos simples implementados com componentes de baixo custo.

A partir do consumo dos circuitos, as tensões e correntes necessárias são as seguintes:

+ 5 V @ 1,6 A

± 15 V @ ± 500 mA

Aplicando uma margem de segurança de 50% nos limites de corrente, os valores para projeto passam a ser os seguintes:

+ 5 V @ 2,1 A

± 15 V @ ± 750 mA

3.9.1 - Fonte de + 5 [V].

Foi utilizado um transformador com tomada central e retificação em onda completa, para obter um menor fator de "ripple".

O diodo escolhido foi o SKE 3/02, com as seguintes

características:

$$V_r = 100 \text{ V}$$

$$I_d = 3 \text{ A}$$

Para cálculo do capacitor de filtro foi utilizada a equação

3.3

$$C = I \frac{dt}{dv} = 2.4 \cdot \frac{8,33 \cdot 10^{-3}}{5} = 3.998 \mu\text{F} \quad \text{eq. 3.3}$$

onde:

C = valor do capacitor de filtro

I = corrente máxima sobre a carga

dt = período do sinal retificado

dv = tensão de "ripple" admitida

O valor comercial escolhido para o capacitor de filtro foi
4.700 μF

Como elemento regulador de tensão, foi utilizado o CI LM
323K, apresentando as seguintes características:

$$I_o = 3 \text{ A}$$

$$V_o = 5 \text{ V} \pm 5\%$$

$$V_{in} > 7,5 \text{ V}$$

Em função dos níveis de tensão e corrente exigidos pelo
regulador, foi especificado o transformador desta fonte. Uma vez
que o capacitor se carrega com o valor da tensão de pico, tem-se:

$$V_{rms} = \frac{V_p}{\sqrt{2}}$$

2

A mínima tensão necessária no secundário do transformador será:

$$V_p = (V_{in} + \frac{V_r}{2}) * 1.5 = 15 \text{ V}$$

-

Assim o mínimo valor aceito para a tensão de saída do transformador será:

$$V_{rms} = \frac{V_p}{2} = \frac{15}{2} = 10,6 \text{ V}$$

O valor comercial mais próximo para o transformador é 12 V @ 3 A, o que atende perfeitamente as exigências projeto.

3.9.2 - Fonte de $\pm 15 \text{ V}$.

Para o projeto desta fonte, foram realizados os cálculos da fonte de + 5 V, e utilizados o mesmo tipo de retificação e filtro e, observando apenas a inversão de polaridade do diodo retificador na construção da fonte de - 15V.

Foi utilizado um transformador com tomada central, do qual foram retiradas as tensões positiva e negativa, conforme mostra a figura 3.23.

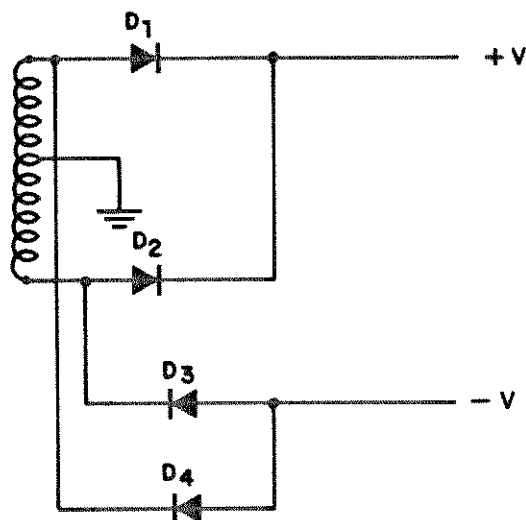


Fig. 3.23 - Retificador para fonte simétrica.

O diodo 1N 4001 foi o que melhor atendeu às solicitações do projeto, pois além de satisfazer as exigências de tensão e corrente da fonte, apresenta baixo custo e boa disponibilidade no mercado. Suas características são as seguintes:

$$V_r = 50 \text{ V}$$

$$I_d = 1 \text{ A}$$

Da mesma forma como para a fonte de 5 V, o capacitor foi calculado a partir da equação 3.3:

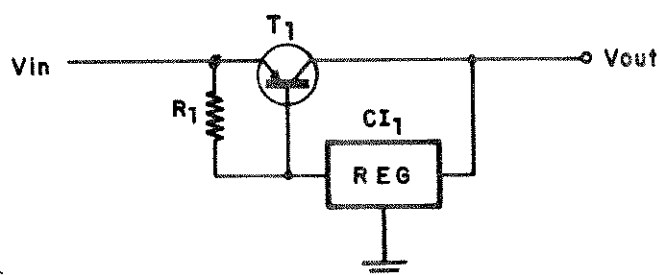
$$C = I \cdot \frac{dt}{dv} = 0,75 \cdot \frac{8,33 \times 10^{-9}}{3} = 2,080 \mu\text{F}$$

O valor comercial escolhido para o capacitor foi $2,200 \mu\text{F}$

Como elemento regulador, foram utilizados CI's 7815 e 7915, para as fontes positiva e negativa, respectivamente. Segundo especificações de manual (NATIONAL, 1982), estes CI's operam com correntes de até 1 A, porém testes de bancada mostraram que para

correntes superiores a 600 mA a tensão de saída começa a cair, perdendo a regulação.

Para suprir esta falha, foi utilizado o amplificador de corrente proposto pela NATIONAL, 1982, apresentado na figura 3.24



* NPN p/ 79XX

Fig. 3.24 - Amplificador de corrente para reguladores 78XX e 79XX.

O esquema elétrico final da fonte é apresentado no anexo VIII.

CAPITULO 4 - TESTES E RESULTADOS

4.1 - Testes do filtro de entrada (Anexo I).

Este teste foi realizado para avaliar a resposta em frequência do filtro de entrada, a faixa de análise (150 a 3.000 Hz).

Para estes testes foram utilizados um gerador de funções programável (TEKTRONIX FG5010) e um osciloscópio digital de 100 MHz (TEKTRONIX 2230A), no esquema de interligação da figura 4.1:

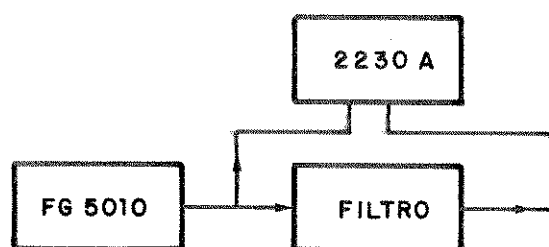


Fig.4.1 - Diagrama em blocos do circuito de testes do filtro de entrada

Para a realização deste teste, foram utilizados os seguintes parâmetros:

- ganho do amplificador de entrada = 1;
- amplitude do sinal de entrada = 3 V_{pp};
- faixa de frequência de 40 a 10.000 Hz;

O gráfico obtido neste teste é apresentado na figura 4.2, elaborado através do programa "GRAPHER V. 2.0", da Golden Software Corp. - 1986.

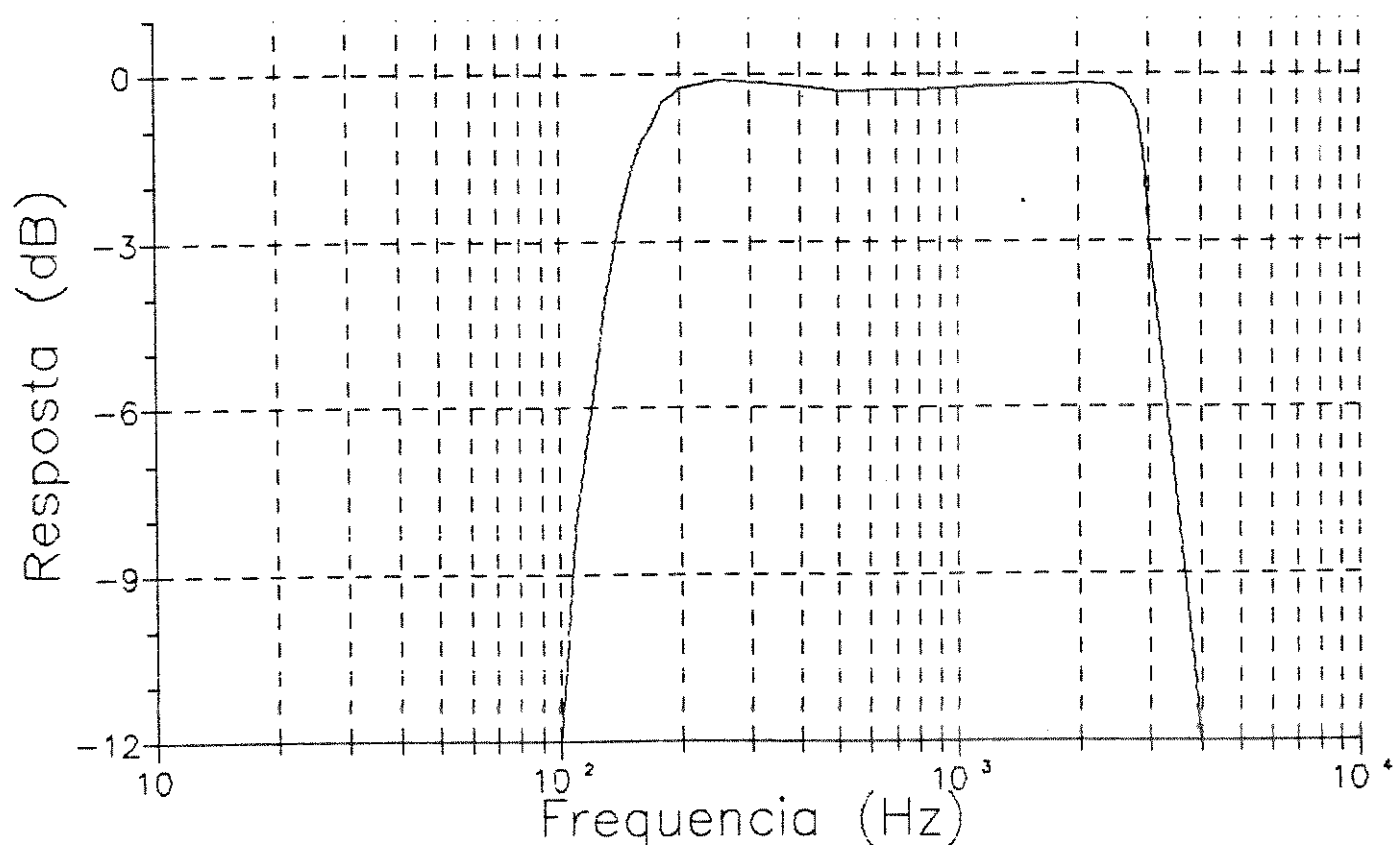


Fig. 4.2 - Resposta em frequência do filtro de entrada.

A atenuação de 1.8 dB na faixa de 150 Hz causada pelo bloco do filtro passa alta, embora atenuie uma faixa de frequência de interesse, não compromete o desempenho do analisador, devido à sua resolução de amplitude em quatro níveis. Essa resolução de amplitude atribui a cada nível da escala de cinza uma faixa de 300 mV, e 1,8 dB representa 60 mV em toda a faixa de operação do CAD, resultando em uma perda cinco vezes menor que a resolução de amplitude do sistema. Apesar da atenuação apresentada pelo filtro passa-alta, seus parâmetros foram mantidos para minimizar os efeitos do movimento das paredes das artérias, artefato indesejado, que situa-se na faixa de 30 Hz, quando utilizado um equipamento Doppler de 4 MHz (LUNT, 1975).

A resposta do filtro passa-baixas garante a passagem de toda faixa de análise e atenua as altas frequências, que podem provocar o efeito de "aliasing" (ATKINSON e WOODCOCK, 1984).

4.2. - Testes do filtro passa-faixa de 70 kHz.

Para este teste foi utilizada a mesma metodologia descrita no item 4.1, com exceção da faixa de frequência, que foi utilizada entre 60 e 80 kHz.

A curva de resposta em frequência deste filtro é apresentada na figura 4.3.

Segundo os cálculos apresentados na tabela 3.3 (item 3.6), é necessário que o filtro atenuie sinais superiores a 81 kHz e

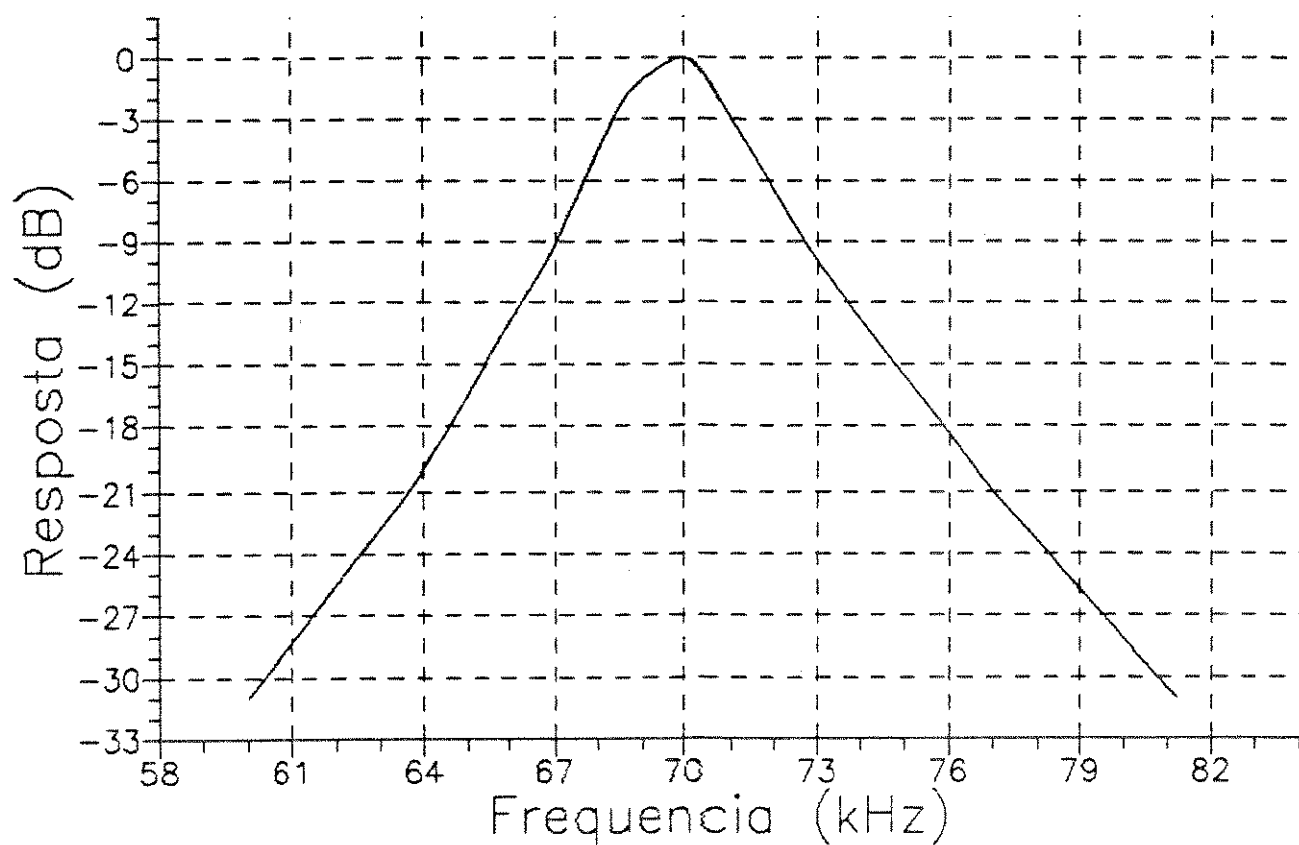


Fig. 4.3 - Resposta em frequência do filtro passa faixa 70 kHz.

inferiores a 65 kHz, e nessas faixas os fatores de atenuação apresentados são de 31 e 17 dB respectivamente, atendendo as especificações de projeto, uma vez que permite a separação entre as faixas de frequência analisadas.

4.3 - Testes do sistema de multiplicação do espectro.

A avaliação do sistema de multiplicação foi feita através de um processo de comparação entre a frequência de entrada do CAD e a de saída do CDA. No teste foi aplicado à entrada do CAD, uma senóide de 1 V_p, obtida de um gerador de funções HP 3310B, em 5 componentes de frequência entre 150 Hz e 3000 Hz (tabela 4.1) e medidos os parâmetros (tensão, frequência e forma de onda) dos sinais, utilizando um contador universal HP 5314 A e um osciloscópio TEKTRONIX 2465BDM para visualização da forma de onda do sinal obtido. O fator de divisão, P, (vide item 3.5) foi obtido através da variação manual de chaves tipo "Dip-switch", possibilitando combinações entre 00h e FFh, o qual será aplicado à entrada de programação do divisor.

A configuração do sistema de teste está representada na figura 4.2.

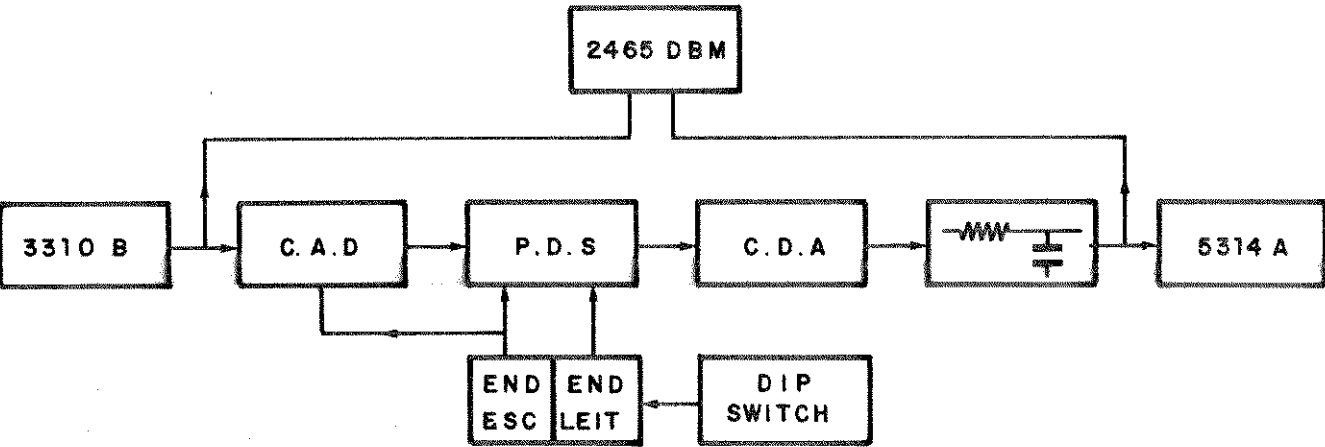


Fig. 4.2 - Sistema de Testes do sistema de deslocamento da frequência.

A partir dos resultados dos testes, foi montada a tabela

4.1.

Fo	P	Fge	Fai	Fao	Vao
14MHz	03	4.66MHz	150Hz	70kHz	6.2
14MHz	04	3.52MHz	200Hz	70kHz	6.8
14MHz	20	700 kHz	1 kHz	70kHz	7.8
14MHz	40	350 kHz	2 kHz	70kHz	7.8
14MHz	60	233 kHz	3 kHz	70kHz	7.8

Tab. 4 . 1 - Resultados dos testes do sistema de deslocamento de frequência.

onde:

Fo = Frequência sinal do oscilador mestre

P = Fator de divisão

Fge= Frequência do sinal aplicado ao gerador de endereços de leitura

Fai= Frequência do sinal analógico aplicado ao CAD de entrada

Faout = Frequência do sinal de saída do CDA.

Vao = Amplitude em Volts do sinal na saída do filtro passivo.

Os resultados apresentados na tabela 4.1 mostram que o deslocamento de frequência responde ao valor carregado no divisor programável, validando a tecnica proposta e permitindo a implementação da varredura automática através da memória PROM.

4.3 - Teste de desempenho do analisador de espectro.

Este teste foi realizado para avaliar a forma de apresentação dos resultados do processamento. Sinais senoidais em diferentes faixas de frequência e amplitude , compreendidos na faixa processada pelo equipamento, foram aplicados utilizando um gerador de funções programável (TEKTRONIX FG5010).

Os resultados são apresentados em vídeo do microcomputador em forma de gráficos (frequência na escala vertical e amplitude em escala de cinza), e obtida a impressão das telas referentes à análise espectral do sinal de entrada.

A partir dos resultados da análise foram montados os gráficos da figura 4.3.

Freq. (Hz)	Amplitude (V)			
	I	II	III	IV
150	0,1	0,4	0,7	1,25
300	0,1	0,4	0,7	1,25
1 k	0,1	0,4	0,7	1,25
2 k	0,1	0,4	0,7	1,25
3 k	0,1	0,4	0,7	1,25

Tab. 4.2 - Parâmetros do sinal de teste do analisador.

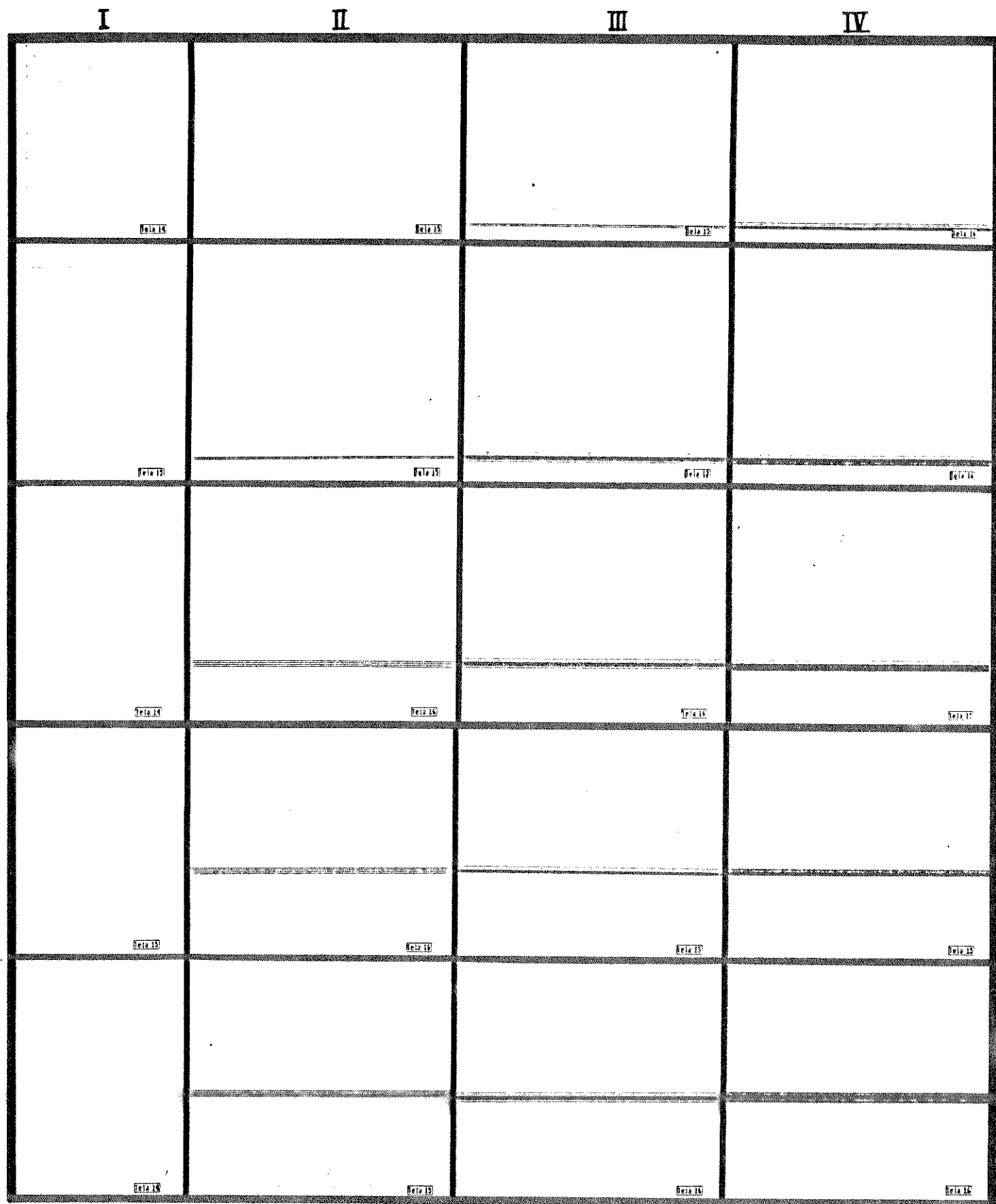


Fig.4.3 - Análise espectral de sinais senoidais.

As colunas de I a IV da figura 4.3 mostram os quatro níveis de cinza correspondentes à amplitude de cada uma das cinco componentes de frequência do espectro, comprovando a capacidade do analisador em identificar a frequência e a amplitude do sinal aplicado à sua entrada.

Este desempenho é válido para todas as faixas de frequência com exceção da faixa de 150 Hz, onde foi detectado um erro na escala de amplitude apresentada. Se comparado visualmente os níveis de cinza entre as cinco faixas de frequência analisadas, observa-se que para a mesma amplitude (colunas) a faixa de 150 Hz apresenta menor intensidade de cinza que as demais faixas (menor amplitude, vide item 3.8), em razão da resposta de frequência do amplificador diferencial do conversor digital-analógico, conforme apresentado no capítulo 5.

Os resultados obtidos nestes testes comprovam o desempenho do equipamento onde foi definido sua utilização para análises qualitativas do espectro de frequências do sinal de entrada.

4.4 - Análise do espectro Doppler ultra-sônico.

Este teste foi executado através de um método de comparação visual entre as curvas da análise espectral obtidas do equipamento desenvolvido e aquelas publicadas em literatura, para uma mesma região sob análise.

Assim, na elaboração deste teste, foi necessário aplicar à

entrada do analisador um sinal Doppler ultra-sônico, obtido a partir da resposta de um determinado vaso sanguíneo. Para tal, foi utilizado um equipamento ultra-sônico marca PARKS, mod. UNILAB II, com transdutor de 4 MHz, obtendo-se a resposta de artérias carótida comum direita e braquial esquerda, como mostram os resultados da figura 4.4 a e b, respectivamente.

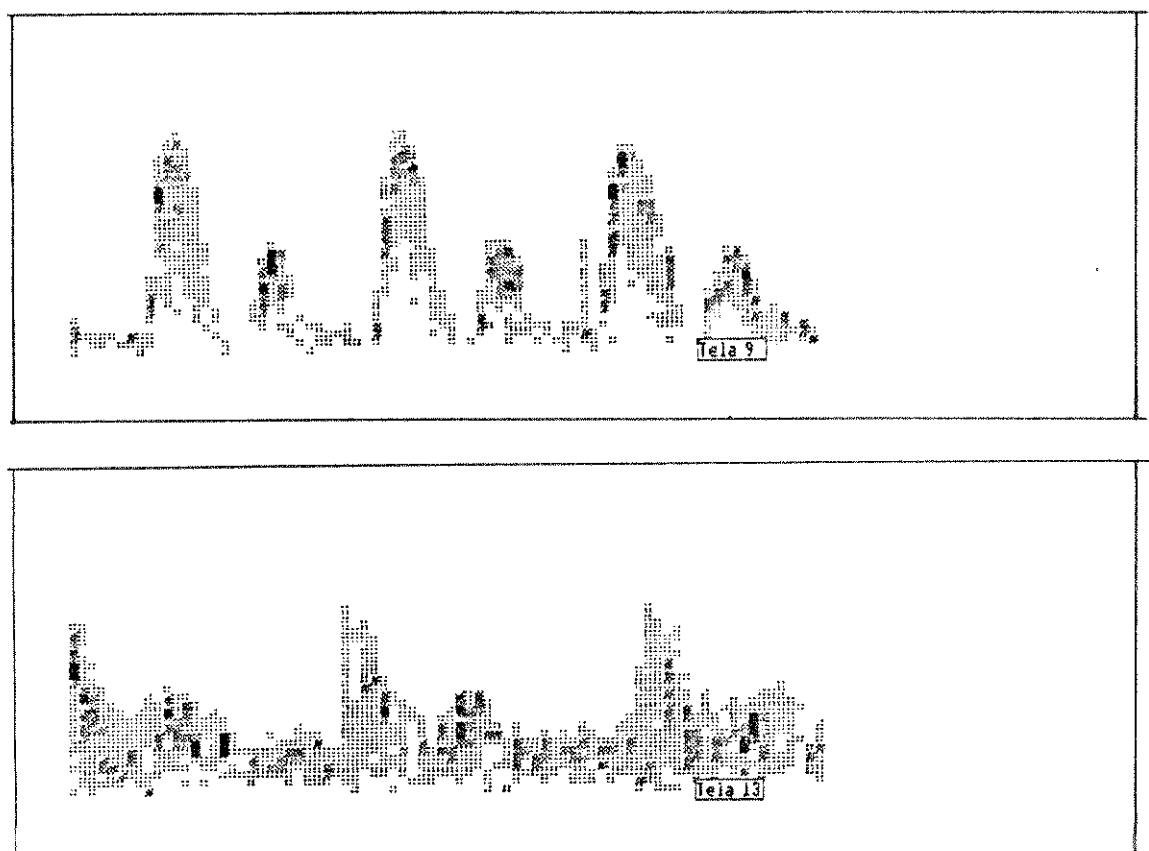


Fig. 4.4 - (a) Resposta Doppler do fluxo da artéria carótida
(b) Resposta Doppler do fluxo da artéria braquial.

Os gráficos obtidos por Gosling, 1976 (1973) são apresentados na figura 4.5:

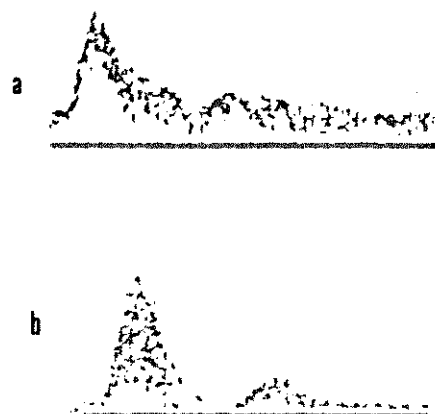


Fig. 4.5 - (a) Resposta Doppler do fluxo na artéria carótida ;
(b) Resposta Doppler do fluxo na artéria braquial.
(Reproduzido de Coghlan et alli, 1973)

CAPÍTULO 5 - CONCLUSÕES E SUGESTÕES

Na análise da faixa de 150 Hz obtida no teste descrito anteriormente (item 4.3), observou-se uma atenuação na amplitude do sinal, através da comparação do nível da escala de cinza nesta frequência com as demais faixas analisadas.

Na avaliação do tempo de resposta dos componentes do equipamento, verificou-se que os amplificadores operacionais (LF 357) utilizados na construção do diferencial para o CDA, possuem faixa de passagem de 20 MHz. Por outro lado, segundo Stout e Kaufman (1976), estes amplificadores devem operar com sinais cuja frequência seja no máximo um décimo da faixa de passagem do componente, ou seja, 2 MHz. Como o sinal de 150 Hz deve ser deslocado de 466 vezes, os dados são fornecidos ao CDA a uma frequência de 4,66 MHz, excedendo o limite de resposta do amplificador, por consequência atenuando a amplitude do sinal aplicado ao filtro passa-faixa. A partir de 300 Hz, onde a frequência de recuperação é 2,33 MHz, a resposta do analisador passa a ser linear.

Para solucionar este problema, sugere-se a escolha de um novo amplificador operacional com banda passante superior a 40 MHz para a construção do diferencial utilizado na saída do CDA. Esta sugestão não pôde ser aplicada, tendo em vista que até a conclusão deste trabalho, não havia disponibilidade de componente no mercado nacional que apresentasse esta característica.

Como forma de aprimorar o analisador, propõe-se um novo estudo do circuito de controle, para avaliar a implementação deste a partir de micro controladores, reduzindo a complexidade do circuito eletrônico.

No trabalho desenvolvido por Prytherch e Evans, (1985), foi utilizado um sistema bi-direcional, permitindo a análise de informações referentes ao fluxo reverso. Esta informação é importante quando da análise de sinais provenientes de vasos periféricos, que apresentam alta impedância, como é o caso das artérias femural. Para implementação desta característica no analisador desenvolvido, sugere-se a fornecimento de mais um "bit" ao micro computador, informando o sentido do fluxo, colocando a faixa inicial do gráfico (ou seja 150 Hz) deslocada da base da tela, permitindo a apresentação das informação de fluxo reverso.

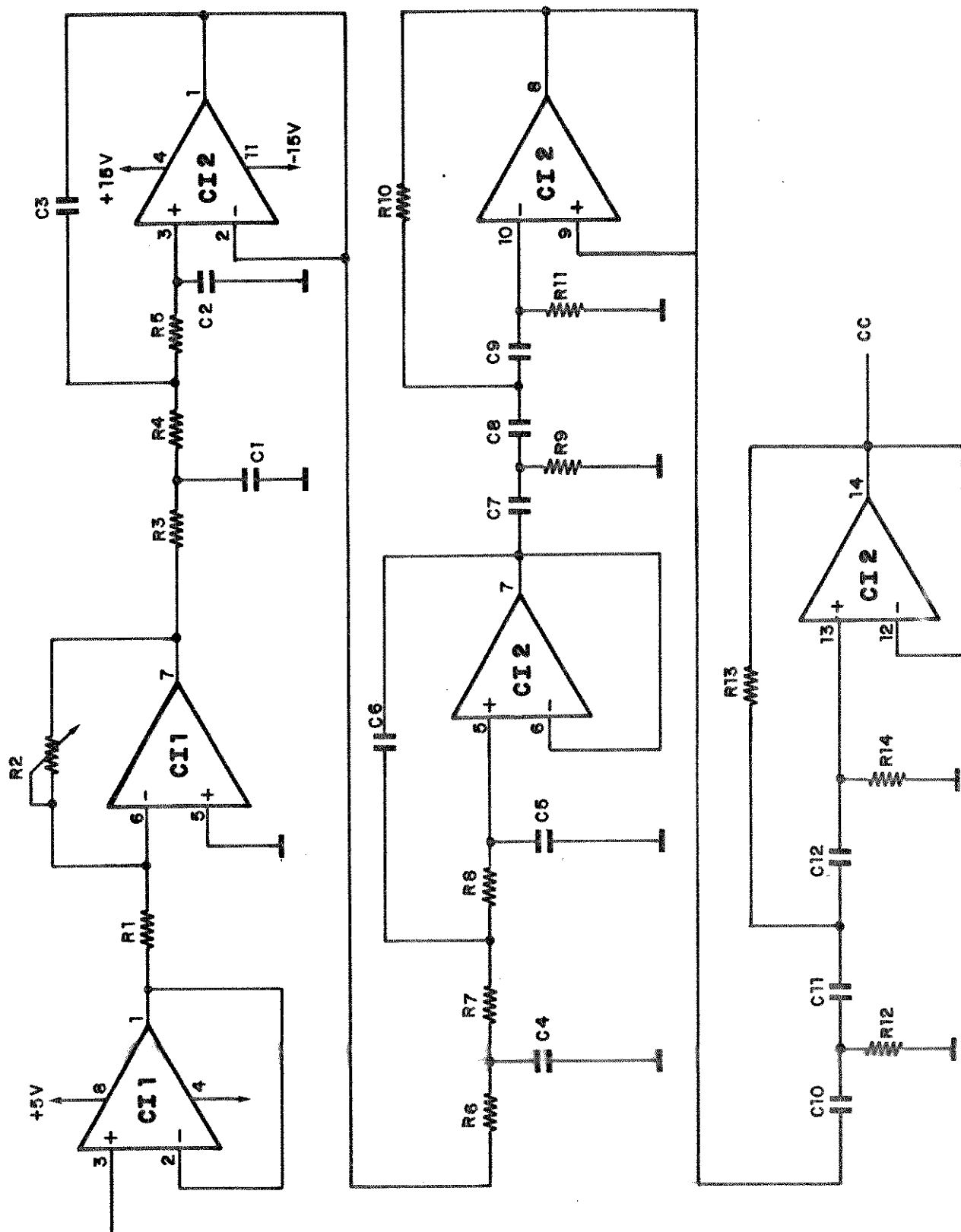
Os custos do sistema podem ser reduzidos, se além dos circuitos desenvolvidos, o equipamento Doppler ultra-sônico, a saída em vídeo e a interface para impressora fossem desenvolvidos com a utilização de circuitos dedicados, e implementado o programa de aquisição e apresentação em EPROM. Porém, dada a atual disponibilidade de processadores de alto desempenho no mercado

nacional, antes que seja dada continuidade a esta técnica, sugere-se um estudo da relação custo x benefício entre o sistema desenvolvido e técnicas de processamento matemático, implementado nestes microcomputadores.

Embora atualmente seja questionável o emprego da técnica de "Time-compression" e filtro de varredura pela razão citada no parágrafo anterior, o desenvolvimento deste trabalho foi de grande valia, pois os objetivos iniciais foram alcançados. A precariedade do mercado nacional em componentes eletrônicos levou à busca de soluções alternativas para o desenvolvimento dos circuitos necessários ao equipamento, incentivando a capacidade de adaptação, aproximando o desenvolvimento deste trabalho à realidade nacional.

A N E X O - I

Esquema Elétrico do Módulo Condicionador de Sinais



ANEXO I

Relação de Componentes - Anexo I

CI₁ - TL072

CI₂ - TL074

R₁ = 10 kΩ

R₂ = 22 kΩ / 1 volta

R₃ a R₈ = 18,2 kΩ 1%

R₉ = R₁₂ = 7,68 kΩ 1%

R₁₀ = R₁₃ = 3,3 kΩ 5%

R₁₁ = R₁₄ = 52,3 kΩ 1%

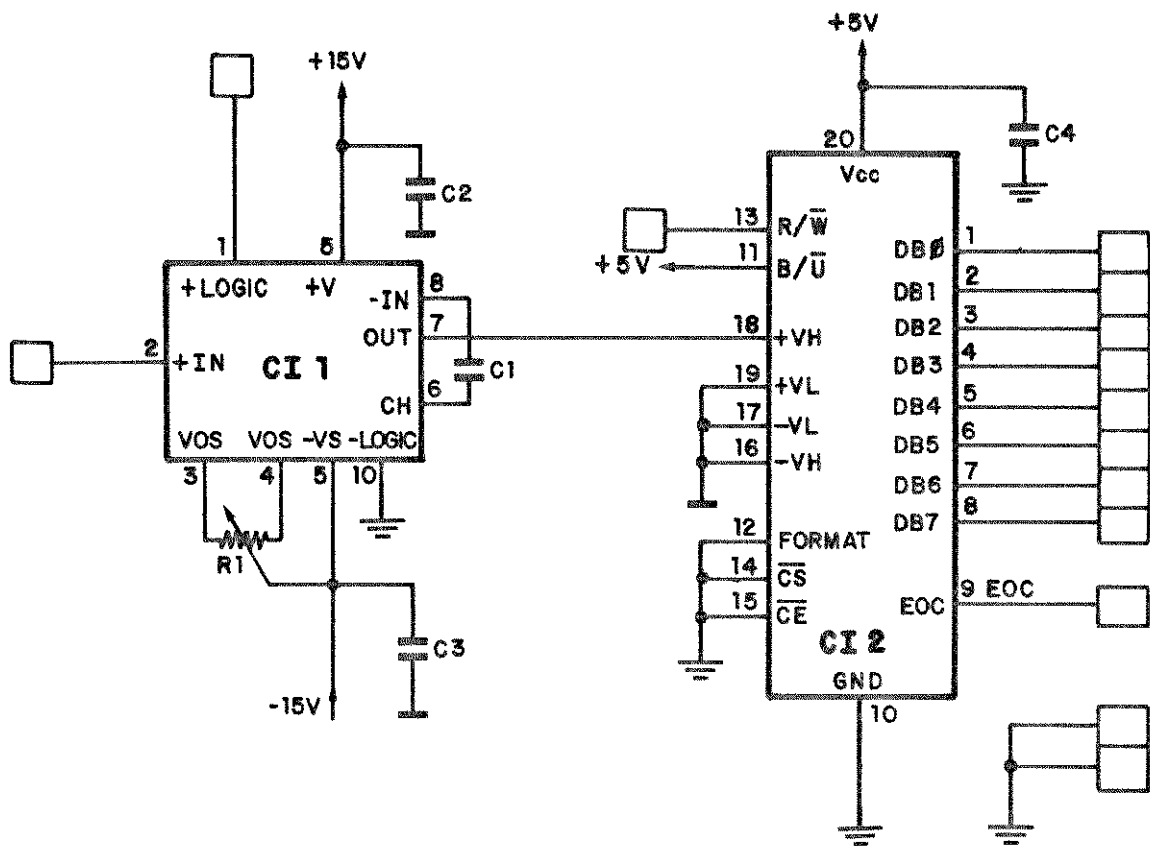
C₁ = C₄ = 3,9 nF / 250 V pol. met.

C₂ = C₅ = 560 pF / 50 V cer.

C₃ = C₆ = 10 nF / 250 V pol. met.

A N E X O - II

Esquema Elétrico do Módulo de Conversão Analógico-Digital



ANEXO II

Relação de Componentes - Anexo II

$C1_1 = AD\ 582\ KH$

$C1_2 = AD\ 670\ JN$

$R_1 = 10\ k\Omega / 15\ voltas$

$C_1 = 100\ pF, 500\ V, teflon$

$C_2\ a\ C_4 = 100\ nF, 50\ V, cer.$

A N E X O - I I I

Mapa do Conteúdo da PROM

Endereço	Dado
00	03
01	04
02	05
03	06
04	07
05	08
06	09
07	0A
08	0B
09	0C
0A	0D
0B	0E
0C	0F
0D	28
0F	29
10	2A
11	2B
12	2C
13	2D
14	2E
15	2F
16	48
17	49
18	4A
19	4B
1A	4C
1B	4D
1C	4E
1D	4F
1E	FF

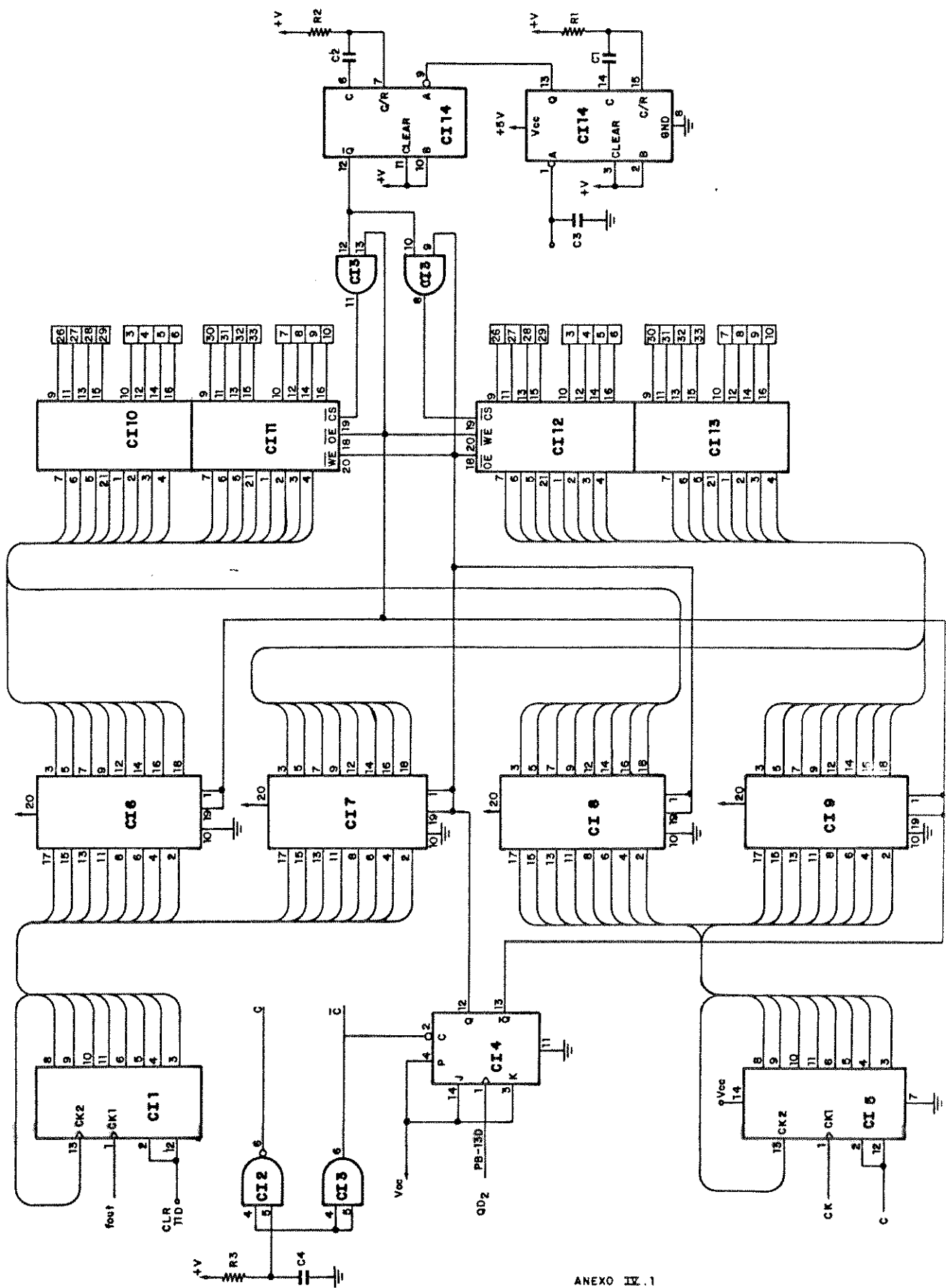
A N E X O - I V

Esquema Elétrico do Módulo de Processamento Digital de Sinais

IV.1 - Banco de Memória.

IV.2 - Gerador de Sinais de Aquisição.

IV.3 - Gerador de Sinais de Recuperação e Controle.



ANEXO IX.1

Relação de Componentes - Anexo IV.1

CI₁ = CI₅ = 74LS393

CI₂ = 74LS08

CI₃ = 74LS00

CI₄ = 7473

CI₆ a CI₉ = 74F244

CI₁₀ a CI₁₃ = 93L422

CI₁₄ = 74LS221

R₁ = 4,7 k Ω

R₂ = 3,3 k Ω

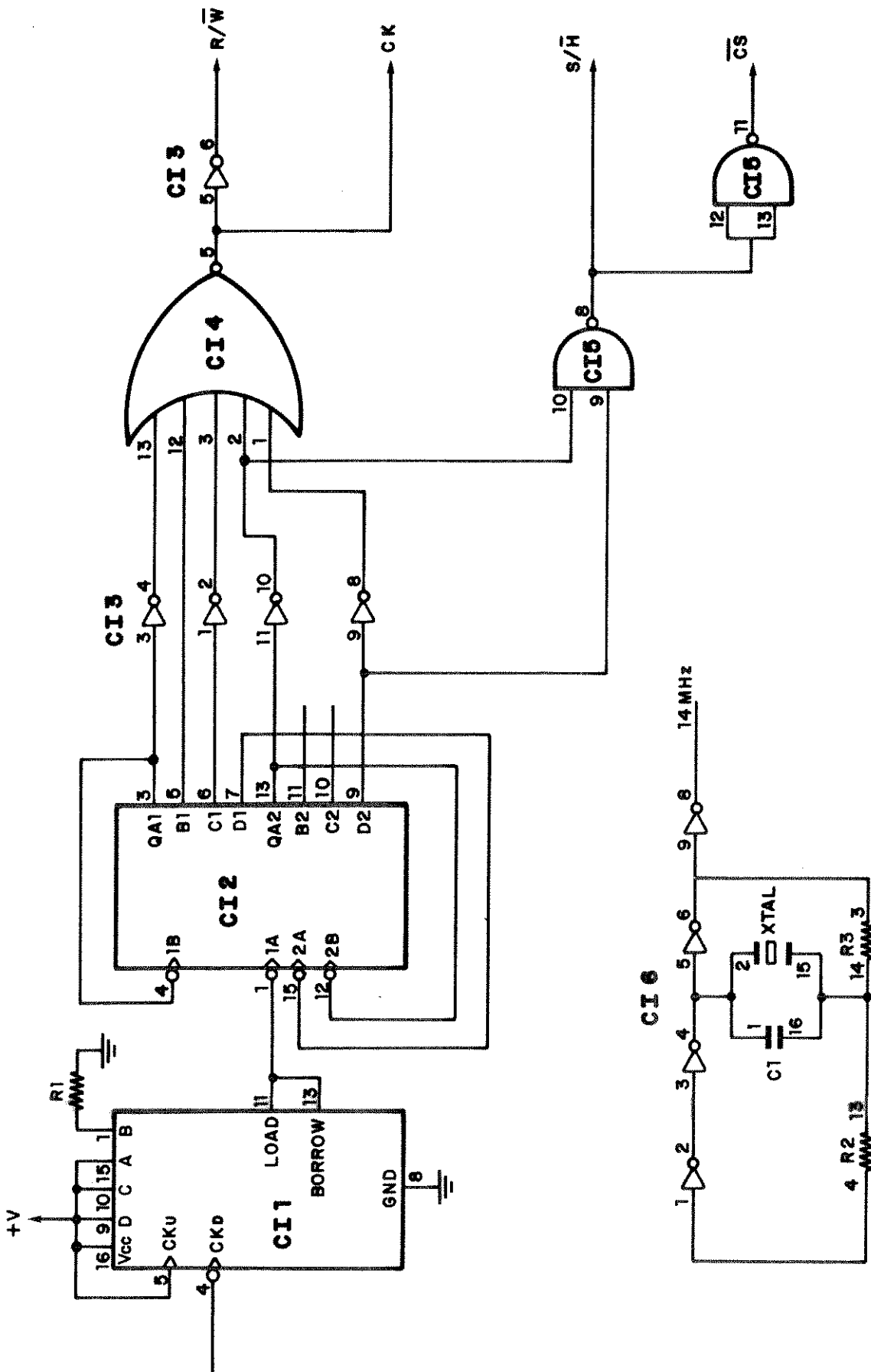
R₃ = 10 k Ω

C₁ = 1 nF, 250 V, pol. met.

C₂ = 220 pF, 60 V, cerâmico

C₃ = 150 pF, 50 V, plate

C₄ = 10 nF, 250 V, pol.met.



ANEXO IV.2

Relação de Componentes - Anexo IV.2

CI₁ = 74LS193

CI₂ = 74LS390

CI₃ = 74F04

CI₄ = 74LS260

CI₅ = 74LS08

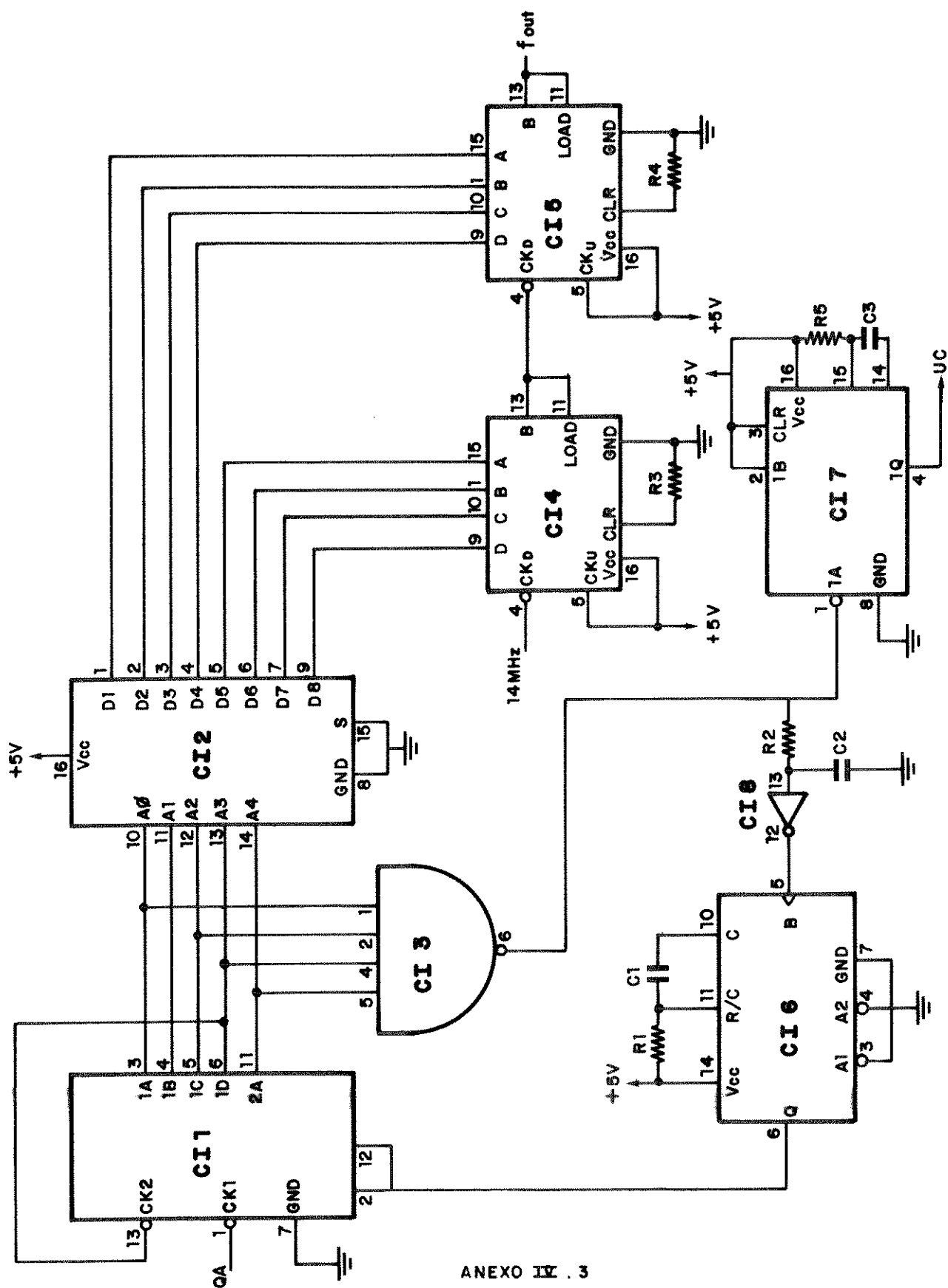
CI₆ = 74LS04

R₁ = 1 k Ω , 5%

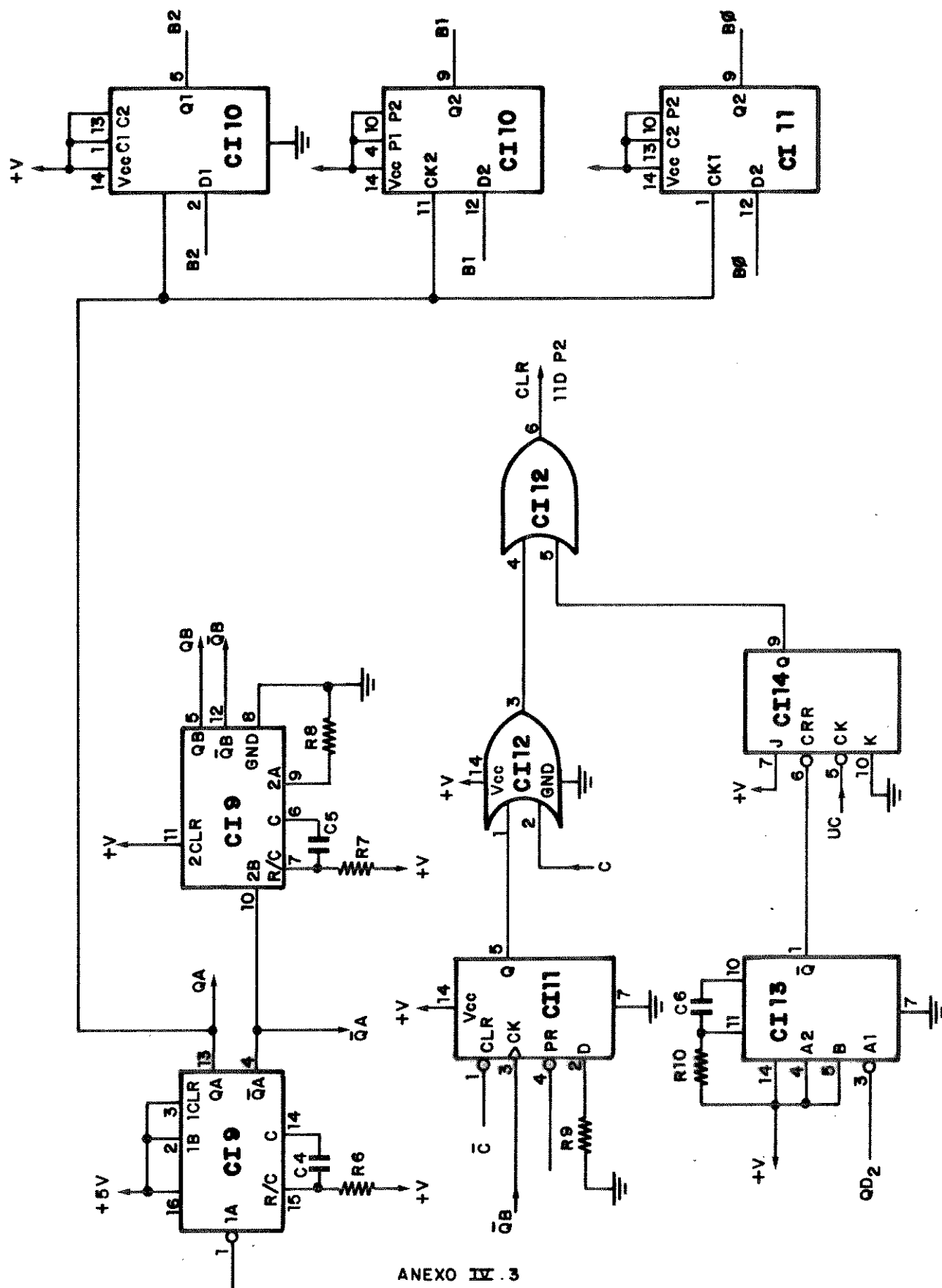
R₂ = 390 Ω

R₃ = 430 Ω

XTAL = Cristal oscilador de 14 MHz



ANEXO IV . 3



ANEXO IV.3

Relação de Componentes - Anexo IV.3

CI₁ = 74LS393

CI₂ = 745288

CI₃ = 74LS20

CI₄ = CIS = 74F193

CI₆ = 74LS121

CI₇ = CI₉ = 74LS221

CI₈ = 74LS04

CI₁₀ = CI₁₁

CI₁₂ = 74LS32

CI₁₃ = 74LS121

CI₁₄ = 7473

R₁ = 10 kΩ

R₂ = 1 kΩ

R₃ = R₄ = R₈ = R₉ = R₁₀ = 1 kΩ

R₅ = 5,6 kΩ

R₆ = R₇ = 4,3 kΩ

R₈ = R₉ = 1 kΩ

R₁₀ = 2,2 kΩ

$C_1 = 1 \text{ nF, } 250 \text{ V, pol. met.}$

$C_2 = 270 \text{ pF, } 50 \text{ V, plate}$

$C_3 = 10 \text{ nF, } 250 \text{ V, pol. met.}$

$C_4 = C_5 = 1 \text{ nF, } 250 \text{ V, pol. met.}$

$C_6 = 100 \text{ pF, } 50 \text{ V, plate}$

A N E X O - V

Esquema Elétrico do Módulo de Processamento Analógico de Sinais

Relação de Componentes - Anexo V

CI₁ = AD581 JH

CI₂ = AD DAC D8 CD

CI₃ = CI₄ = CI₆ = LF 353N

CI₅ = TL074

CI₇ = LM339N

Q₁ = MOS FET IRF 510

D₁ = FDH300

R₂ = 47 k Ω 1%

R₃ = R₄ = 10 k Ω 1%

R₅ = 50 k Ω , 15 voltas

R₆ = R₇ = 30 k Ω , 1%

R₈ = R₉ = 10 k Ω , 1%

R₁₁ = R₁₂ = R₁₃ = 10 k Ω , 1%

R₁₄ = 500 k Ω , 15 voltas

R₁₇ = R₁₈ = R₁₉ = 10 k Ω , 5%

R₂₀ = 47 k Ω 5%

R₂₁ = 4,7 k Ω , 5%

R₂₂ = R₂₅ = R₂₈ = R₃₁ = 48,4 k Ω , 1%

R₂₃ = R₂₆ = R₂₉ = R₃₂ = 249 [Ω], 1%

R₂₄ = R₂₇ = R₃₀ = R₃₃ = 98,7 k Ω , 1%

C₃ = 100 nF, 25 V, cerâmico

$C_6 = 10 \text{ nF}$, 25 V, cerâmico

$C_7 \text{ a } C_{10} = 100 \text{ nF}$ 25 V, cerâmico

$C_{12} = 4700 \text{ pF}$, styroflex

$C_{13} = C_{14} = C_{15} = 100 \text{ nF}$, 25 V, cerâmico

$C_{16} = 470 \text{ pF}$, styroflex

$C_{17} \text{ a } C_{24} = 470 \text{ pF}$, 25 V

ANEXO - VI

Módulo de Interface Paralela

ANEXO VI - INTERFACE PARALELA

A interface paralela desenvolvida será ligada a um conector de expansão do IBM PC-XT. A configuração do conector está representada na figura VI.1, e os pinos utilizados pela interface são descritos a seguir:

-RESET DRV: sinal que fica em nível 1 durante a inicialização do sistema, utilizado para resetar interfaces ao se ligar o micro.

-A0-A19: sinais de saída usados no endereçamento de memória e acesso às portas de I/O. Para as portas, apenas os sinais de A0 a A9 são usados.

-D0-D7: sinais bidirecionais usados para transmissão de dados entre o microprocessador, memória e I/O.

-IRQ2-IRQ7 (Interrupt ReQuest): internamente ligados ao controlador de interrupções 8259A. IRQ2 é a mais prioritária e IRQ7 a menos. IRQ2, IRQ3 e IRQ5 são normalmente disponíveis ao usuário.

-IOW (I/O Write): sinal de saída ativo em nível 0. Vai de nível 1 para nível 0 quando da execução de um OUT pela CPU.

-IOR (I/O Read): também é um sinal de saída ativo em nível 0 que assume este estado quando da execução de um IN pela CPU.

-AEN (Address ENable): sinal de saída que vai a nível 1 indicando que os barramentos estão sendo usados para uma operação de DMA.

-+5 V DC: utilizado na alimentação de interfaces.

-GND: Terra comum do micro.

O controlador de interrupções 8259A tem por função expandir

para 8 o número de interrupções mascaráveis do microprocessador 8088, que possui apenas uma (INTR).

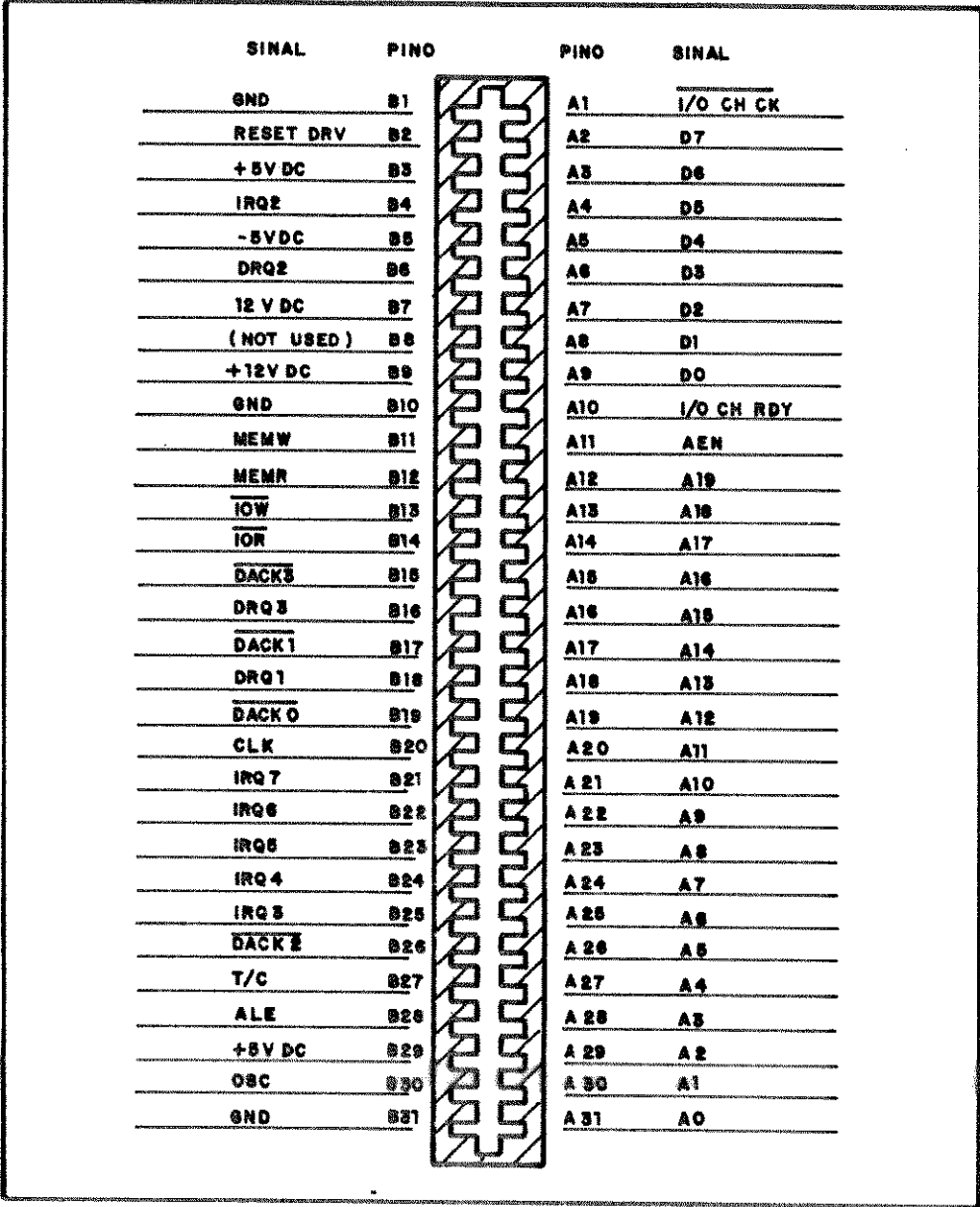


Fig. VI.1 - Conector de expansao do PC-XT

A figura VI.2 mostra o diagrama de blocos do sistema de interrupções do IBM-PC:

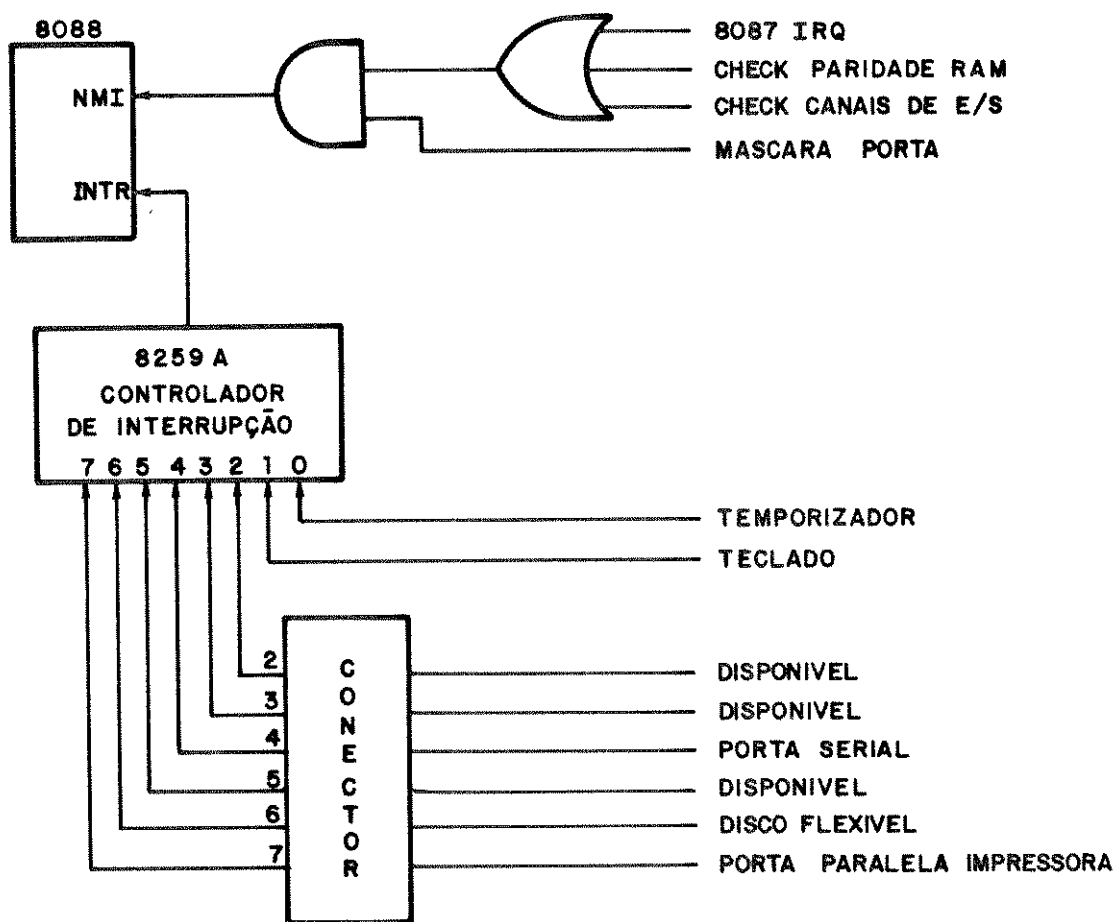


Fig. VI.2 - Diagrama de blocos do sistema de interrupção do PC

Conforme mostrado no figura acima, as entradas de requisição de interrupções de 2 a 7 estão disponíveis no slot do

micro através dos pino IRQ2 a IRQ7.

Por "default", IRQ2 tem mais prioridade que IRQ3, que tem mais prioridade que IRQ4 e assim por diante.

Dependendo da aplicação, pode-se alterar a configuração inicial do 8259A alterando-se os registradores internos do controlador chamados de "Operation Control Mode Word" ou OCW.

No caso deste do projeto foi necessário o uso de OCW1 e de OCW2:

-OCW1 : cada bit do registrador OCW1 está associado à máscara de uma entrada de interrupção. Pode-se habilitar ou desabilitar o reconhecimento das 8 interrupções através da máscara da seguinte forma:

OCW1: IRQ7 IRQ6 IRQ5 IRQ4 IRQ3 IRQ2 IRQ1 IRQ0

Bit n = 0 --> reconhecimento de IRQn habilitado

Bit n = 1 --> reconhecimento de IRQn desabilitado

OCW2: registrador utilizado para se alterar as prioridades e para indicar ao controlador que uma rotina de serviço de interrupção chegou ao final. No projeto foi utilizado para indicar o final da rotina de serviço. Para isto deve-se colocar no registrador OCW2 a palavra 6nh, onde "n" deve ser o número da interrupção que ocasionou o desvio da execução para a rotina de serviço. Por exemplo, se for feita uma interrupção através de IRQ2 deve-se enviar ao 8259A o byte 62h ao final da rotina de serviço.

No IBM-PC tem-se a porta de endereço 0021h associada a OCW1 e a porta 0020h associada a OCW2, portanto para se alterar um destes registradores deve-se executar uma instrução OUT end, novo

valor do registrador (end = 020h ou 021h).

VI.1 - Análise das formas de interfaceamento.

Nesta etapa foram analisadas as diversas possibilidades para a aquisição de dados e escolheu-se o tipo de comunicação mais adequada ao analisador.

O interfaceamento entre um dispositivo e um microcomputador pode ser representado por um transmissor controlando e/ou enviando dados a um receptor. No caso do sistema em questão, o transmissor representa o Analisador de Espectro e o receptor o microcomputador.

Basicamente, temos as seguintes possibilidades para comunicação com um microcomputador:

-Por "pooling", onde o receptor fica esperando que o transmissor envie um sinal avisando que há um dado para se lido. O micro deve ficar em "looping" lendo um bit que está ligado a este sinal. A vantagem deste modo é que a execução do programa é sequencial, eliminando-se a necessidade de saltos para outras rotinas. Isto torna a aquisição mais rápida, porém apresenta a desvantagem do micro ficar preso ao transmissor, ficando inteiramente a seu serviço, impossibilitando-o de realizar outras tarefas enquanto não houver dado para leitura.

-Por interrupção, cada vez que o transmissor indicar que há um dado para leitura, o microcomputador interrompe o processo em

execução (programa principal) e salta para uma determinada rotina que deve ler o dado e, se necessário, processá-lo. Ao terminar esta rotina a execução volta ao programa principal. Esta técnica possui a vantagem do micro processar outras informações até o surgimento de um dado, portanto, não ficando totalmente preso ao transmissor. Sua desvantagem é a perda de tempo, pois a cada interrupção do micro, ocorre um desvio na execução do programa e deve-se salvar os registradores para o retorno à rotina principal, no final da rotina de serviço de interrupção.

-Através de Acesso Direto à Memória (DMA), o dado é colocado diretamente na memória sem que o microprocessador seja utilizado para tal comunicação. A vantagem deste processo é apresentar-se como mais rápido dentre os citados acima, em razão do micro não ficar preso ao transmissor, sendo indicado para a transferência de blocos de memória entre um transmissor e o micro. Esta técnica apresenta uma desvantagem quando utilizada para a aquisição de um único dado, pois para cada leitura é necessária uma nova programação da interface, tornando o processo lento.

Após a análise destes 3 métodos de aquisição, foi escolhido neste projeto o modo de interrupção, por apresentar a vantagem de permitir um processamento enquanto não houver dados para leitura. Entre as aquisições, o microcomputador realiza a apresentação dos dados em vídeo

VI.2 - Projeto e construção da placa de interfaceamento.

A placa de interface construída foi projetada com as seguintes características:

- Dado disponível na forma paralela de 8 bits;
- Necessidade de interromper o micro quando um dado estiver disponível;
- Não interferir em nenhum outro processo do micro (interrupções internas);

Escolheu-se o circuito integrado 8255A como o elemento principal para o interfaceamento.

Este circuito é uma PPI (Programmable Peripheral Interface) com as seguintes características principais:

- trabalha com níveis TTL;
- 3 portas independentes e programáveis de 8 bits cada;
- 3 modos de operação (0,1 e 2);
- projetada para ser usada com microprocessadores da INTEL, dentre os quais se inclui o 8086/8088;

A PPI possui 3 portas (A, B e C) que são programadas para funcionarem como entrada ou saída de dados:

As portas A e B são de 8 bits e podem ser programadas independentemente para entrada ou saída de dados.

A porta C é utilizada, no modo 0, como duas portas de 4 bits cada (C Low e C High). Pode-se programar CL para funcionar como porta de entrada e CH como saída, ou ambas como entrada, ou ambas como saída, ou ainda CL como saída e CH como entrada. Nos

modos 1 e 2 a porta C é utilizada para entrada e saída de sinais de controle. Em qualquer modo a porta C pode ser acessada bit a bit, isto é, pode-se setar ou resetar qualquer, ou quaisquer, dos 8 bits desta porta.

O esquema de pinagem e diagrama de blocos da PPI está representado na figura VI.3.

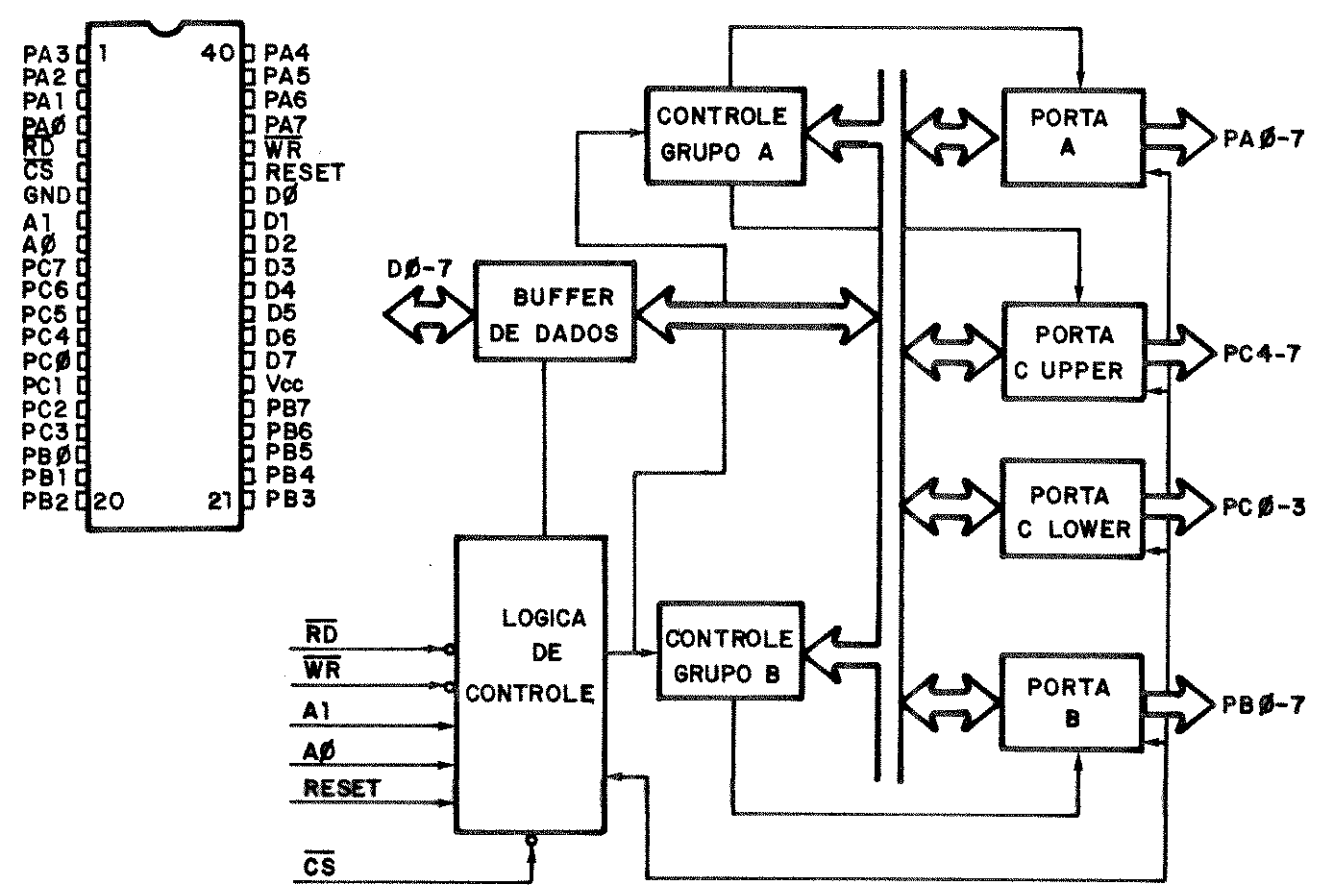


Fig. VI.3 - (a) Pinagem, (b) diagrama em blocos da 8255

Dentre os 3 modos de operação da PPI, decidiu-se utilizar o modo 1 (descrito abaixo) com a porta A programada para entrada e a porta B como saída. Não havia necessidade de uma porta para saída, isto para o caso de alguma futura comunicação entre o micro e outro dispositivo.

Modo 1: Entrada/Saída com strobe: neste modo, quando a operação for de entrada de dados, a PPI deve receber um sinal do dispositivo que os está enviando, indicando que há um dado disponível. A PPI armazena este dado e envia ao microcomputador um sinal para interrupção e consequente leitura.

A seguir é feita a descrição dos sinais de controle usados para a entrada de dados no modo 1. Esta descrição justifica a escolha deste modo.

-STBa (STrobe Input): Quando vai a nível 0, o dado é lido via pinos PA0-PA7 e colocado a disposição para leitura via D0-D7 (D0-D7 ficam em "3-state" enquanto o micro não fizer a leitura).

-IBFa (Input Buffer Full): Um nível 1 nesta saída indica que o dado já foi lido pela porta A:

-INTR (INTerrupt Request): Um nível 1 nesta saída pode ser utilizado para interromper o micro. Para que isto ocorra é necessário "setar" PC4 da PPI, conforme será descrito posteriormente. Feito isto, sempre que um dado for lido pela PPI e estiver disponível para a leitura pelo micro, este sinal vai a nível 1. Se o micro estiver preparado para ser interrompido, o dado deverá ser lido numa rotina de serviço de interrupção.

A PPI possui alguns pinos que devem ser devidamente

conectados ao micro para que o funcionamento seja o desejado:

-CS (Chip Select): Um nível 0 nesta entrada habilita a comunicação entre a PPI e o micro.

-RD (Read): Um nível 0 nesta entrada habilita a PPI a enviar o dado ou informação de status ao micro. Indica que a CPU está realizando uma leitura na PPI. Deve ser ligado ao IOR do micro, o qual vai a nível 0 quando é dado um IN numa porta de I/O.

-WR (Write): Um nível 0 nesta entrada habilita a CPU a escrever um dado ou uma palavra de controle na PPI. Deve ser ligado ao IOW do micro que vai a 0, quando da execução de um OUT para uma porta de I/O.

-A0 e A1 (Port select): Em conjunto com RD e WR, seleciona qual das 3 portas ou a palavra de controle estão sendo acessada. São conectadas às linhas do barramento de endereço A0 e A1. Abaixo é apresentada a tabela de como A1, A0, RD, WR e CS determinam qual porta está sendo acessada.

AO	A1	RD	WR	CS	efeito
0	0	0	1	0	porta A => barramento de dados
0	1	0	1	0	porta B => barramento de dados
1	0	0	1	0	porta C => barramento de dados
0	0	1	0	0	barramento de dados => porta A
0	1	1	0	0	barramento de dados => porta B
1	0	1	0	0	barramento de dados => porta C
1	1	1	0	0	barramento de dados => controle
x	x	x	x	1	barramento de dados em 3-state
1	1	0	1	0	ilegal
x	x	1	1	0	barramento de dados em 3-state

Tabela VI.1 - Palavras de Controle da 8255.

-RESET: Um nível 1 nesta entrada limpa o registrador de controle e a PPI é levada ao modo 0 com as 3 portas funcionando como entradas. Deve ser ligado ao RESET DVR do micro.

Programação da PPI: Para que a PPI trabalhe no modo desejado deve-se programá-la indicando o modo e a configuração das portas.

A PPI reconhece que uma palavra de controle está sendo enviada a ela quando AO=A1=1 e WR=0. A figura VI.4 apresenta as possibilidades de programação da PPI:

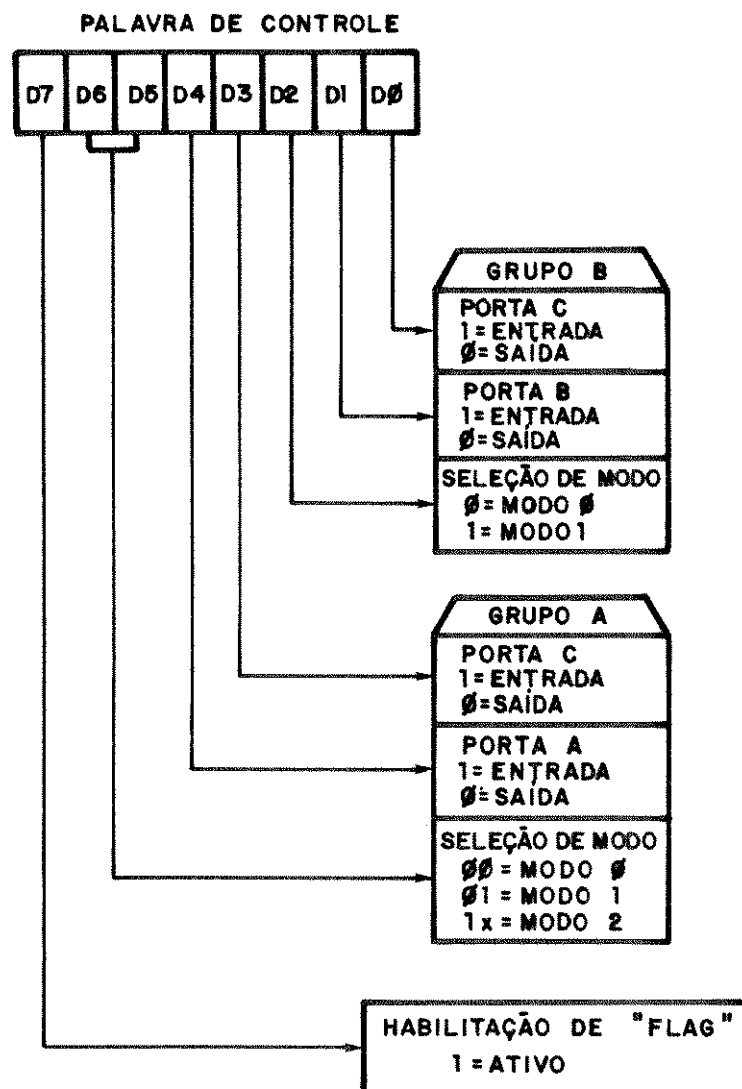


Fig. VI.4 - Palavras de Programação da 8255.

No caso da interface, conforme descrito anteriormente, optou-se pelo modo 1, porta A como entrada e porta B como saída. Escolheu-se também PC6 como entrada para que o sinal de sincronismo UC seja aplicado a ela. Sendo assim, utilizou-se a palavra de controle 10111101b = BDh para programar a PPI.

Pode-se habilitar ou desabilitar cada bit da porta C. Isto é feito usando-se a palavra de controle representada na figura VI.5

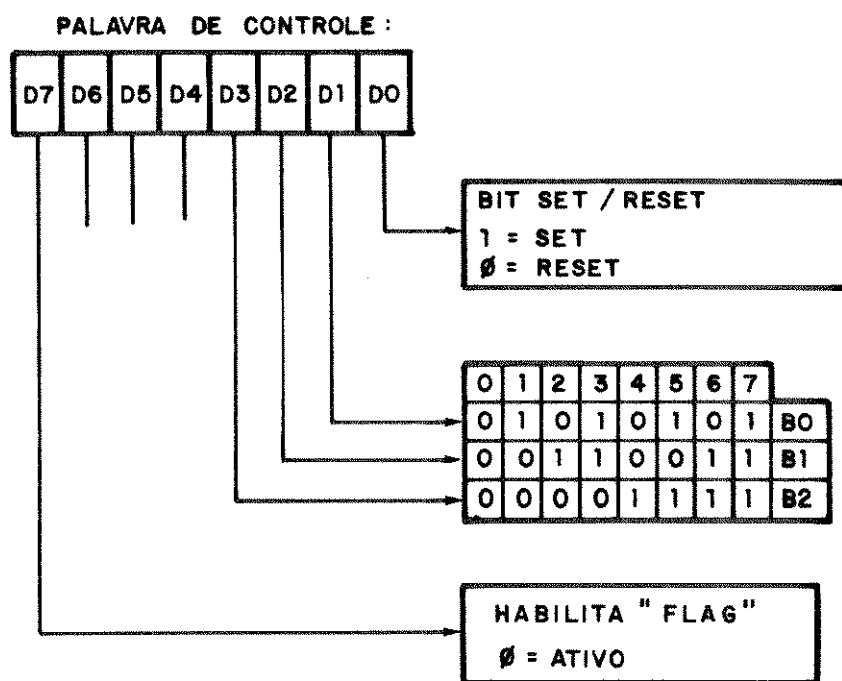


Fig. VI.5 - Programação da porta C da 8255

Esta operação habilitar/desabilitar a porta C deve ser enviada à PPI como uma palavra de controle.

Observando a tabela de A0 e A1, nota-se que os endereços de I/O devem ser convenientemente escolhidos, dentre os disponíveis, para que se possa acessar todas as portas da PPI. Os endereços escolhidos foram:

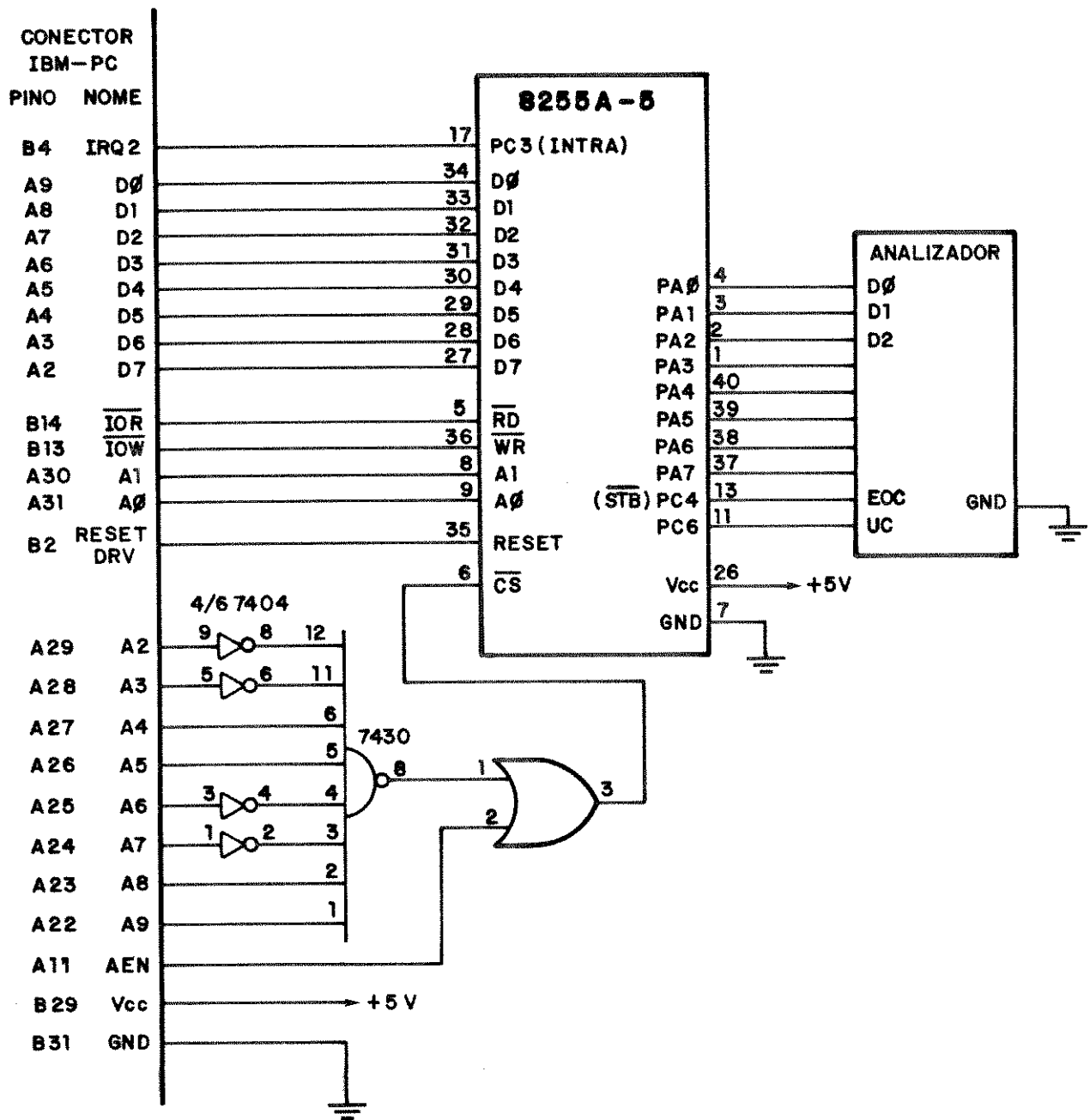
Porta A - end. 0330h
Porta B - end. 0331h
Porta C - end. 0332h
Controle - end. 0333h

Um circuito de "Chip Select" foi projetado para habilitar a interface somente quando o micro estivesse acessando uma destas 4 portas.

Optou-se por ligar o sinal INTRa no pino IRQ2 do conector, já que esta requisição de interrupções está disponível ao usuário.

Conforme dito anteriormente, para habilitar a transmissão do sinal de interrupção pela PPI, deve-se habilitar o "bit" 4 da porta C (PC4). Para tal usou-se a palavra de controle 00001001b = 09h.

O diagrama esquemático da placa de interfaceamento está representado na figura VI.6



ANEXO - VII

Programa de Aquisição e Apresentação

-Listagens e Fluxogramas-

```

#include <dos.h>
#include <stdio.h>
#include <graphics.h>
#include <conio.h>
#pragma inline

/* GRAF1.C 17:00:09 10/24/1990 */

#define linmax 29 /* numero de componentes de freq. lidas. */
#define colmax 80 /* numero de colunas na tela. */
#define increndarm linmax*colmax /* incremento no endereco p/ armazenar. */
#define lcentr 97 /* linha que divide a tela em 1/4. */

main()
{
    int graphdriver = DETECT,graphmode; /* auto deteccao do modo grafico */

    /* variaveis da entrada de dados e armazenamento na memoria: */
    unsigned int aux, /* -variavel auxiliar usada no programa. */
    endarm, /* -ender.p/ armazenamento dos dados lidos; */
    ntelas, /* -ntelas armazenadas. */

    /* variaveis do scroll de telas armazenadas: */
    endcor, /* -ender. inicial da tela armazenada; */
    telacor, /* -tela sendo mostrada; */
    ntelamax, /* -numero de telas armazenadas (1 a 16); */
    nmsgtela; /* -numero da tela sendo mostrada; */
    char msgtela[3], /* -mensagem do num. da tela sendo mostrada. */
    c; /* -usada na leitura do teclado. */

    extern void prtela(); /* rotina em Assembly para preencher tela e
                           armazena dados lidos. */
    extern void telaarm(); /* rotina em Assembly para mostrar tela
                           armazenada. */

    if (registerbgidriver(CGA_driver)<0)
        cprintf("CGA nao esta presente \n");
    if (registerbgidriver(EGAVGA_driver)<0)
        cprintf("EGAVGA nao esta presente \n");
    if (registerbgifont(small_font)<0)
        cprintf("LITT nao esta presente \n");

    initgraph(&graphdriver,&graphmode,""); /* inicializa modo grafico */
    aux=getgraphmode(); /* aux=modo grafico */
    restorecrtmode(); /* volta ao modo texto */

    normvideo(); /* mensagem inicial */
    cprintf("----- \n");
    cprintf("Utilize ");
    highvideo();
    cprintf("ESC ");
    normvideo();

```

```

cprintf("para sair da entrada de dados. \n");
highvideo();
cprintf("Tecle <ENTER> para comecar \n");
getch();

ntelamax=endarm=ntelas=0;          /* inicializa variaveis
                                     para entrada de dados. */

setgraphmode(aux);                  /* modo grafico */
settextstyle(0,0,1);                /* caracteres tipo normal */

m4:
outportb (0x0333,0x0bd);            /* programa PPI: modo 1,
                                     porta A=input,
                                     porta B=output,
                                     PC6 e PC7=input. */
outportb (0x0333,0x09);             /* inicia emissao do sinal de
                                     interrupcao pela PPI. */

m1:
_BX=endarm;                         /* BX=offset inicial para armazenar dados lidos:
                                     parametro para o programa em Assembly. */
prtela();                           /* prtela()-> prog. em Assembly que le e
                                     armazena os dados. */

endarm += increndarm;               /* endarm=endereco inicial para armazenar
                                     proxima tela. */
ntelas++;                          /* incrementa numero de telas armazenadas. */

if (ntelas >= 16) { endarm=0; /* ntelas esta sempre entre 0 e 16. */
                    ntelas=0; } /* se ntelas chegar a 16, recomeca a
                                   armazenar do endereco 0. */
if (ntelamax < 16) ntelamax++; /* ntelamax nao ultrapassa 16. */

aux=64;                             /* se nao houver caracter sendo */
asm mov ah,01;                       /* teclado faz aux=64, */
asm int 016h;                        /* caso contrario, aux=codigo */
asm jz m6;                           /* ASCII do caracter teclado. */
asm mov ah,07;
asm int 021h;
aux=_AL;

m6:                                  /* se tecla 'ESC' nao estiver */
if (aux!=27) goto m1;                /* pressionada, volta a m1. */

outportb (0x0333,0x08);              /* interrompe emissao do sinal de
                                     interrupcao pela PPI. */

/*-----*/
/* INICIO DO SCROLL DE TELAS */
setlinestyle(0,0,1);

setviewport(10,180,365,199,0); /* mensagem para scroll de telas. */
clearviewport();
rectangle(0,0,355,19);
outtextxy(4,2,"Utilize: <- e -> para vizualizar as telas;");
outtextxy(4,10,"<BARRA ESPACO> para finalizar.");

```

```

nmsgtela = ntelamax;      /* inicializa variaveis p/ mostrar telas. */
telacor = ntelas;
endcor = endarm-increndarm;

setviewport(530,189,588,199,0); /* mensagem de numero da tela */
clearviewport();               /* sendo mostrada. */
rectangle(0,0,58,10);
outtextxy(2,2,"Tela");
itoa(nmsgtela,msgtela,10);
outtextxy(41,2,msgtela);

m2:
c = getch();                  /* c = caracter teclado */
if (c == 'P') goto m3;        /* se c='P', vai para m3. */

else
if ((c == '6') || (c == 'M')) { nmsgtela++;          /* seta -> */
telacor++;
endcor += increndarm; }

else
if ((c == '4') || (c == 'K')) { nmsgtela--;          /* seta <- */
telacor--;
endcor -= increndarm; }

else goto m2;

if (telacor < 1) { telacor = ntelamax;
endcor = (ntelamax-1)*increndarm; }

else
if (telacor > ntelamax) { telacor = 1;
endcor = 0; }

if (nmsgtela < 1) nmsgtela = ntelamax;
else if (nmsgtela > ntelamax) nmsgtela = 1;

_BX = endcor;                /* BX = offset inicial dos dados armazenados;
parametro para a rotina em Assembly. */
telaarm();                   /* preenche tela grafica com caracteres atribuidos
aos dados armazenados a partir de endcor. */

setviewport(0,0,639,199,0);   /* desenha linha em lcentr. */
/* line(0,lcentr,639,lcentr);
/* for (aux=0; aux<630; aux+=10)
/* line(aux,lcentr-1,aux,lcentr+1); /* aux=coluna */

setviewport(530,189,588,199,0); /* mensagem do numero da tela */
clearviewport();               /* sendo mostrada. */
rectangle(0,0,58,10);
outtextxy(2,2,"Tela");
itoa(nmsgtela,msgtela,10);
outtextxy(41,2,msgtela);

goto m2;                      /* volta ao inicio do scroll de telas. */

m3: setviewport(10,180,385,199,0); /* opcoes para reiniciar aquisicao */

```

```

clearviewport();          /* dados ou finalizar.      */
rectangle(0,0,375,19);
outtextxy(2,2,"Tecla: 'R ' para reiniciar entrada de dados;");
outtextxy(2,10,"      <BARRA ESPACO> para finalizar processo.");
m5:  c=getch();
if ((c=='r')||(c=='R')) goto m4;  /* se opcao for 'r', reinicia. */
else if (c!=' ') goto m5;         /* se for ESPACO, termina.   */

restorecrtmode();          /* modo texto */

highvideo();              /* mensagem de fim do processo. */
cprintf("Fim do processo. \n");

closegraph();

}

```

name grafal
title GRAFA1.ASM 10:12:47 17/11/1988
page 60,132

;Este programa contem 3 funcoes:
;-PRTELA: utilizada pelo programa em C;
;-RSIN2 : utilizada pela funcao PRTELA;
;-TELAARM: utilizada pelo programa em C.

;%%%%%%%%%%%
%%%%%%%%%

;COMANDO PRTELA USADO PELO PROGRAMA GRAF1.C

;Faz tabela de Pointers de Interrupcao IRQ2 apontar para _RSIN2
;-Programa em C fornece atraves de BX o offset inicial para armazenar dados
;-Inicializa registradores para apontarem para enderecos de tela grafica
;-Habilita reconhecimento somente de IRQ2
;-Fica em looping esperando uma requisicao de interrupcao IRQ2
;-Quando interrompido: .le dado da porta A da PPI
;
; .dependendo do valor do dado, atribui a ele um caracter
;
; .coloca este caracter na tela grafica
;
; .atualiza registradores p/ apontarem p/ o proximo end.
;
; .retorna ao programa principal
;-Desabilita IRQ2
;-Volta ao programa em C

; ** Atribuicao de valores:
OFIRQ2 EQU 0028H ;offset do pointer de IRQ2 na tabela de inter.
CONTRPPI EQU 0333H ;porta de controle da PPI.
LERPA EQU 0330H ;porta A.
LERPC EQU 0332H ;porta C.
SGGRAF EQU 0B800H ;segmento memoria de video.
OFGRAF EQU 01EF0H ;offset memoria de video (em baixo, esquerda).
LINMAX EQU 29 ;numero de linhas em uma coluna.
PULACOL EQU LINMAX*160+1 ;usado para iniciar nova coluna.
OCW1 EQU 021H ;usado para mascarar interrupcoes pelo 8259.
OCW2 EQU 020H ;usado para enviar EOI para o 8259.
COLMAX EQU 80 ;numero de colunas na tela.
C0 EQU 0000 ;caracteres para serem colocados na tela.
C1 EQU 00CCH
C2 EQU 0CC33H
C3 EQU 0FFFFH
SGARM EQU 05000H ;segmento da memoria para armazenar dados

; ** Definicao de segmentos para o MASM, compativel com o TURBOC

_TEXT SEGMENT BYTE PUBLIC 'CODE'
_DGROUP GROUP _DATA, _BSS
ASSUME CS:_TEXT, DS:_DGROUP

; ** Inicio da funcao.
; ** O programa em C chamara' a funcao pelo nome prtela

```

PUBLIC      _prtela      ;definida como PUBLIC p/ ser acessada pelo C.
_prtela PROC NEAR

    PUSH AX      ;salva registradores.
    PUSH BX
    PUSH CX
    PUSH DX
    PUSH ES
    PUSH DS
    PUSH SI
    PUSH DI
    PUSH BP

    MOV  DI,BX      ;DI=offset inicial para armazenar dados lidos.

; ** Atribui ao pointer de interrupcao IRQ2 (interrupcao numero 10) o
; ** endereco de _RSIN2
    MOV  AX,0000      ;DS=0000 =segmento da tabela de
    MOV  DS,AX        ; pointers.
    MOV  BX,OFIRQ2    ;BX=0028h = offset da tabela
                        ; de pointers para IRQ2.
    MOV  AX,OFFSET _RSIN2 ;AX=offset _RSIN2.
    MOV  WORD PTR DS:[BX],AX ;pointer de offset=AX.
    MOV  WORD PTR DS:[BX+2],CS ;pointer de segment=CS (segmento de
                                ; IRQ2 e'o mesmo que o de prtela).

; ** Inicializa registradores
    MOV  AX,SGARM
    MOV  DS,AX        ;DS=segmento memoria de armazenamento de dados.

    MOV  AX,SGGRAF
    MOV  ES,AX        ;ES=segmento memoria video de alta resolucao.
    MOV  BX,OFGRAF    ;BX=offset inicial memoria video, apontando
                        ; para o ponto inferior esquerdo da tela.
    MOV  CL,0         ;CL=linha inicial.
    MOV  CH,0         ;CH=coluna inicial.

; ** Espera sinal de sincronismo que devera' ser enviado quando a ultima
; componente de freq. estiver sendo transmitida.
; O sincronismo estara ligado 'a porta PC6 da PPI.
; O programa ficara em looping, esperando que este sinal de sincronismo
; seja recebido. As interrupcoes serao reconhecidas apos a recepcao da
; borda de subida deste sinal.
; Com isto, o progr. iniciara' o preenchimento da tela a partir da
; interrupcao subsequente 'a ultima componente enviada, ou seja, a
; partir da primeira componente de frequencia transmitida.
; Nota-se que o programa perdera' uma coluna enquanto estiver esperando
; o sinal de sincronismo.
; Isto dara' um tempo para que o programa em C faca as atualizacoes
; necessarias nas variaveis, sem que o sincronismo seja perdido.

    PUSH DX      ;salva DX
    MOV  DX,LERPC ;DX=endereco porta C da PPI

```

```

S0:    IN     AL,DX           ;fica em looping ate' que sinal de
      AND    AL,040H         ;sincronismo esteja em nivel zero.
      JNZ    S0

```

```

S1:    IN     AL,DX           ;fica em looping ate' que sinal de
      AND    AL,040H         ;sincronismo va' a nivel 1.
      JZ     S1

```

```

      POP    DX              ;recupera DX.

```

;** Inicia preenchimento de tela e armazenamento de dados lidos.

;** Habilita somente reconhecimento de IRQ2

```

      CLI                      ;desabilita reconhecimento de interrupcoes.
      IN     AL,OCW1           ;AL=valor da mascara de interrupcoes atual.
      PUSH   AX                ;salva AL na pilha.
      MOV    AL,0FBH           ;faz mascara habilitar somente IRQ2.
      OUT    OCW1,AL           ;envia mascara para o 8259.
      STI                      ;habilita reconhecimento de interrupcoes.

```

;** Looping de espera

```

M0:    CMP    CH,80            ;se coluna for <80, fica em looping.
      JB     M0

```

;** Habilita interrupcoes que estavam inicialmente habilitadas

```

      CLI
      POP    AX                ;AL=mascara original de interrupcoes.
      OUT    OCW1,AL           ;envia mascara para o 8259.
      STI

```

```

      POP    BP                ;recupera registradores.

```

```

      POP    DI

```

```

      POP    SI

```

```

      POP    DS

```

```

      POP    ES

```

```

      POP    DX

```

```

      POP    CX

```

```

      POP    BX

```

```

      POP    AX

```

```

      RET                    ;volta ao programa em C

```

;** Rotina de servico de interrupcao

_RSIN2:

```

      CLI                      ;desabilita reconhecimento de interrupcoes.
      INC    CL                ;incrementa linha.
      MOV    DX,LERPA          ;AL=dado lido da PPI.
      IN     AL,DX

```

```

      MOV    BYTE PTR DS:[DI],AL ;armazena dado lido na memoria.

```

```

      INC    DI                ;incrementa offset da memoria p/ armazenar.

```


;** O programa em C chamara' a funcao pelo nome telaarm

```
        PUBLIC _telaarm      ;definida como PUBLIC p/ ser acessada pelo C
_telaarm PROC NEAR

        PUSH    AX           ;salva registradores
        PUSH    BX
        PUSH    CX
        PUSH    DX
        PUSH    DS
        PUSH    ES
        PUSH    DI
        PUSH    SI
        PUSH    BP

        MOV     SI,BX        ;SI= end. inicial dados armazenados fornecido
                                ; pelo prog. em C.
        MOV     CH,0         ;CH= coluna inicial.
        MOV     CL,0         ;CL= linha inicial.

        MOV     AX,$GARM
        MOV     DS,AX        ;DS= segmento onde os dados estao armazenados.

        MOV     AX,$GGRAF
        MOV     ES,AX        ;ES= segmento memoria de video.
        MOV     DI,OFGRAF    ;DI= offset inicial memoria de video, apontando
                                ; para o canto inferior esquerdo.

TA5:    INC     CL            ;incrementa linha.
        MOV     AL,BYTE PTR DS:[SI] ;AL= dado armazenado em prtela.
        INC     SI           ;incrementa apontador de dado armazenado.

        CMP     AL,0         ;se dado<>0, vai para TA0,
        JNE     TA0         ;caso contrario:
        MOV     DX,C0        ;DX= caracter C0
        JMP     SHORT TA3    ;pula para TA3.

TA0:    CMP     AL,1         ;se dado<>1, vai para TA1,
        JNE     TA1         ;caso contrario:
        MOV     DX,C1        ;DX= caracter C1
        JMP     SHORT TA3    ;pula para TA3.

TA1:    CMP     AL,3         ;se dado<>3, vai para TA2,
        JNE     TA2         ;caso contrario:
        MOV     DX,C2        ;DX= caracter C2
        JMP     SHORT TA3    ;pula para TA3.

TA2:    MOV     DX,C3        ;DX= caracter C3.

TA3:    MOV     BYTE PTR ES:[DI],DL ;DL na linha par
        MOV     BYTE PTR ES:[DI-80],DL ;DL na proxima linha par
        MOV     BYTE PTR ES:[DI+02000H],DH ;DH na linha impar
        MOV     BYTE PTR ES:[DI+02000H-80],DH ;DH na proxima linha impar.
        SUB     DI,160       ;DI= offset do proximo
```

; caracter a ser colocado.

```
CMP CL,LINMAX ;se linha diferente de linmax, volta a TA5.
JNE TA5 ;caso contrario:
ADD DI,PULACOL ;DI=offset do inicio da proxima coluna.
MOV CL,0 ;linha=0.
INC CH ;incrementa coluna.
CMP CH,COLMAX ;se coluna diferente de colmax, volta a TA5.
JB TA5
```

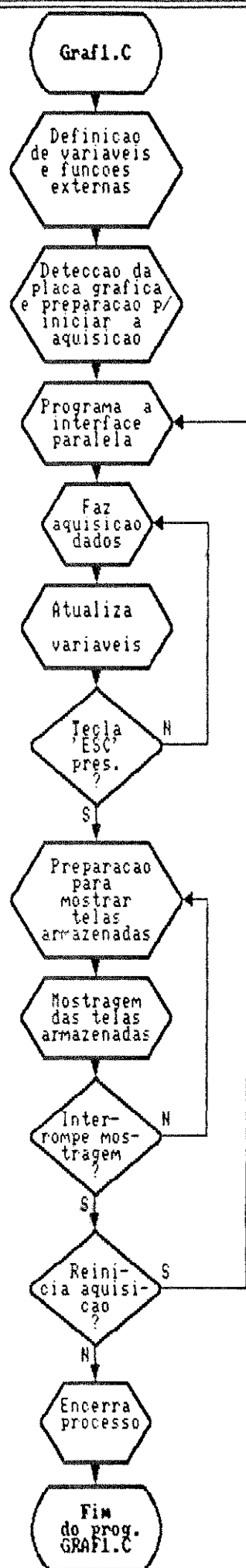
```
POP BP ;recupera registradores salvos.
POP SI
POP DI
POP ES
POP DS
POP DX
POP CX
POP BX
POP AX
RET ;volta ao programa em C.
```

```
_telaarm ENDP ;fim da funcao telaarm.
;~~~~~
;*****
;
_TEXT ENDS

_DATA SEGMENT WORD PUBLIC 'DATA'
_DATA ENDS

_BSS SEGMENT WORD PUBLIC 'BSS'
_BSS ENDS

; ** Final da definicao de segmentos
;*****
;
END
```



Definicao de variaveis e de funcoes externas

Definicao de variaveis

Nome da variavel	Uso no programa	Valor Inicial
.GRAPHDRIVER (int)	.utilizada na deteccao da placa grafica pelo TURBOC 1.5	-
.AUX (int)	.utilizada para ir do modo grafico para o modo texto no inicio do programa	-
.ENDARM (int)	.offset inicial para armazenar na memoria os dados lidos	0
.NTELAS (int)	.usada no controle de ENDARM	0
.NTELAMAX (int)	.numero maximo de telas armazenadas. Seu valor nao ultrapassa 16	0
.ENDCOR (int)	.offset inicial p/ mostrar os dados armazenados	0
.TELCOR (int)	.usada no controle de ENDCOR	=ntelas
.NMSGTELA (int)	.numero da tela armazenada que esta sendo mostrada	=ntelamax
.NMSGTELA[3] (char)	.usada para mostrar o valor de MSGTELA na tela de alta resolucao	=nmsgtela
.C (char)	.usada para entrada de opcoes via teclado	-

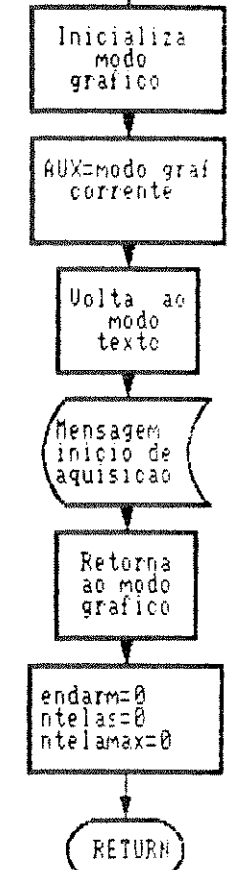
Funcoes Externas

.PRIELA -> Funcao contida no prog. em Assembly GRAFA1.ASM; Faz a aquisicao de dados via placa projetada, atribui ao dado um caracter da escala cinza, mostra o caracter na tela e armazena o dado no segmento 5000h e offset fornecido pelo programa GRAFI.C; Repete enquanto a tela nao for preenchida.

.TELAARM -> Funcao contida no prog. em Assembly GRAFA1.ASM; Recupera dados armazenados no segmento 5000h e offset fornecido pelo programa GRAFI.C, colocando-os em forma de escala cinza na tela; Repete ate preencher uma tela.

Deteccao da placa grafica e preparacao para iniciar a aquisicao

Detecca que placa grafica esta presente atraves do comando do IC 1.5 REGISTERBGIDRIVER



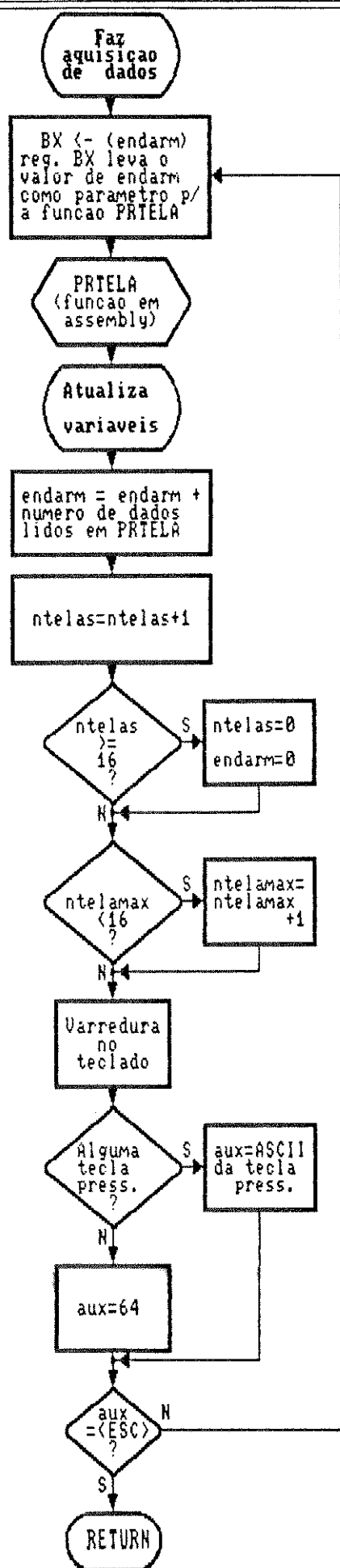
Programa a interface paralela

Programa PPI 8255A-5 da seguinte forma:
 .modo 1
 .porta A=input
 .porta B=output
 .pC6 e pC7=input

Habilita emissao do sinal de interrupcao pela PPI

RETURN

Obs: AL, BX, CL, CH, DX (DL+DH), DI, ES, ES
são registradores do 8088/8086



Funcao PRIELA
(contida no pr. GRAF1.ASM em Assembly)

salva registradores

DI = BX = offset p/ arm. dados (forn. pelo pr. GRAF1.C)

Tabela de pointers de interrupcoes:
int 10 (IRQ2)
aponta p/ rotina RSIN2

DS = 5000h (segm. p/ arm. dados)
ES = B800h (segm. memoria video)

BX = offset do canto inferior esquerdo da tela na mem. de video

CL = 0 → linha e
CH = 0 → coluna iniciais

Espera sinal de sincronismo

Habilita somente interrupcoes via int 10 (IRQ2)

Houve interrupcao por hardware via IRQ2 ?

RSIN2

CH > 80 (preencheu tela) ?

Habilita interrupcoes inicial/habilitadas

Recupera registr. salvos no inicio da funcao

RETURN

Espera sinal de sincronismo vindo c/a ultima componente de freq. (F50)

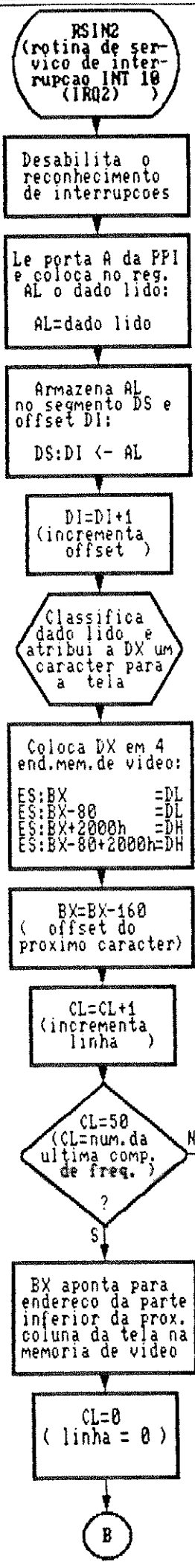
Le bit 6 da porta C (PC6) da PPI (sinal de sincron. esta ligado a PC6)

PC6 = 0 ?

Le PC6

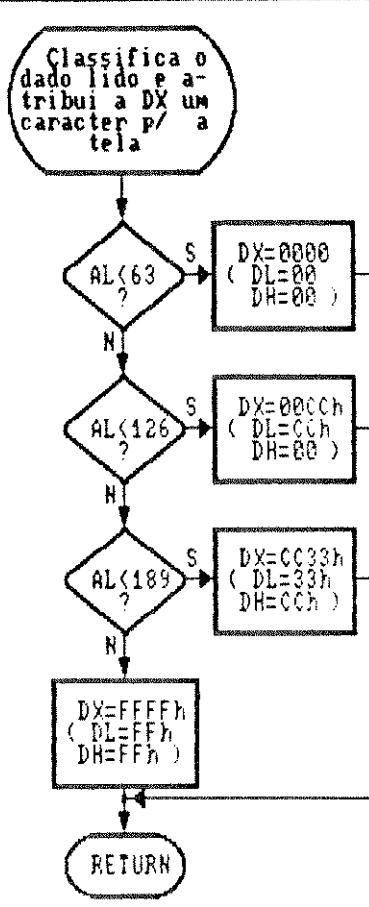
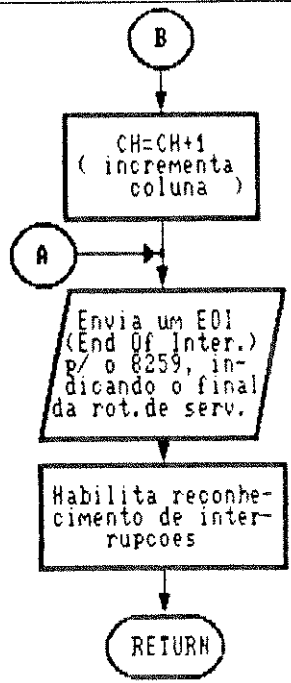
PC6 = 1 ?

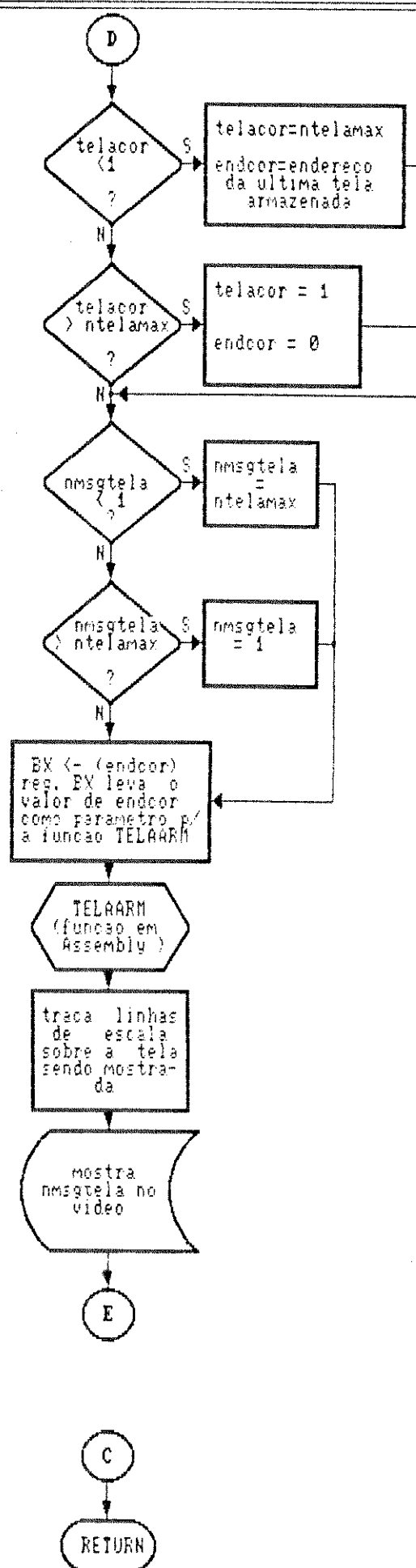
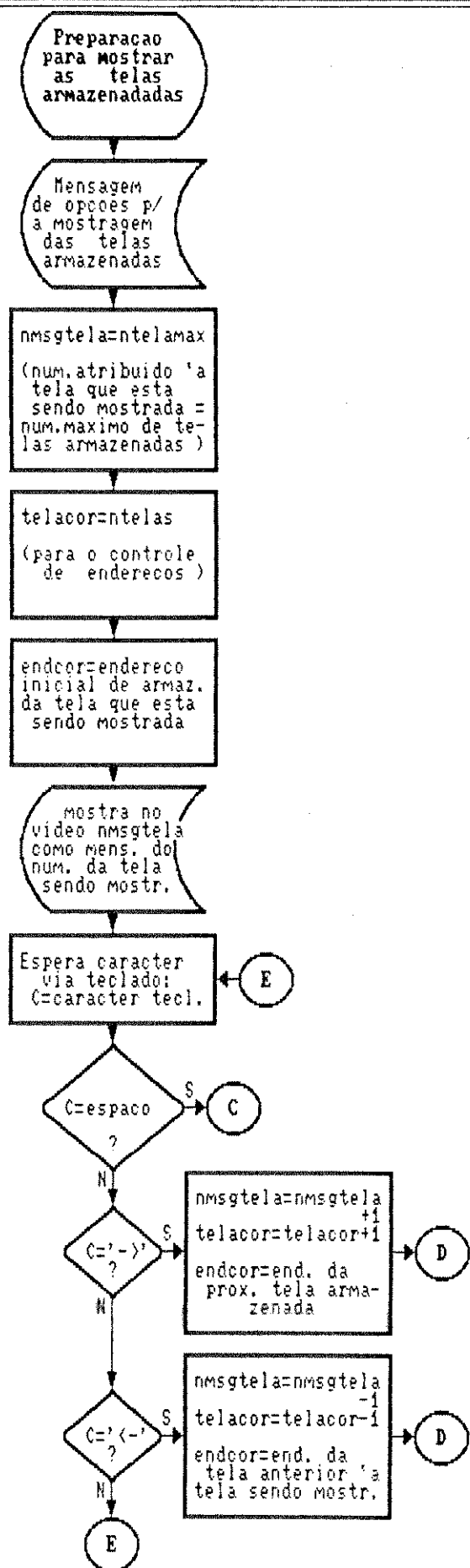
RETURN



memoria de video:

conteudo	endereço
	BX - 160
	BX-160 +2000h
DHL	BX-80
DH	BX-80 +2000h
DL	BX
DH	BX +2000h





Funcao
TELAARM
(contida no pr.
GRAFI.ASM em
Assembly)

salva
registra-
dores

SI=BX =offset
inicial dos dados
armazenados, for-
necido como para-
metro pelo progr.
GRAFI.C

CL=0-> linha e
CH=0-> coluna
iniciais

DS=5000h (segm.
onde os dados
estao armaz.)
ES=8000h (segm.
memoria video)

DI= offset do
canto inferior
esquerdo da
tela na mem.
de video

AL= conteudo do
segm. DS e
offset SI
AL <- ES:[SI]

G

SI=SI+1
(incr.
offset)

Classifica
dado lido (AL)
e atribui a DX
um caracter
p/ a tela

Coloca DX em 4
end.mem.de video:
ES:DI =DL
ES:DI-80 =DL
ES:DI+2000h =DH
ES:DI-80+2000h=DH

DI=DI-160
(offset do
proximo caracter)

CL=CL+1
(incrementa
linha)

F

memoria de video:
conteudo|endereço

	DI-160 +2000h
DL	DI-80
DH	DI-80 +2000h
CL	DI
CH	DI +2000h

caracter

F
CL=50
(CL=num.da
ultima comp.
de freq.)
N
G

DI aponta para
endereço da parte
inferior da prox.
coluna da tela na
memoria de video

CL=0
(linha = 0)

CH=CH+1
(incrementa
coluna)

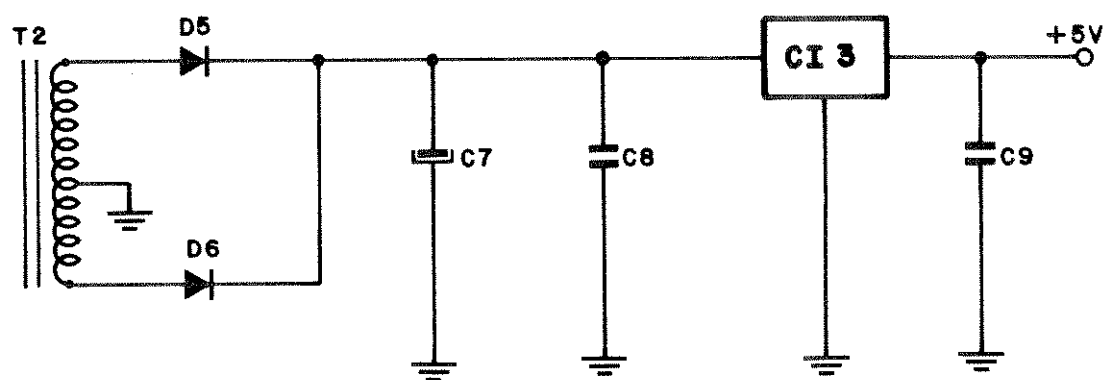
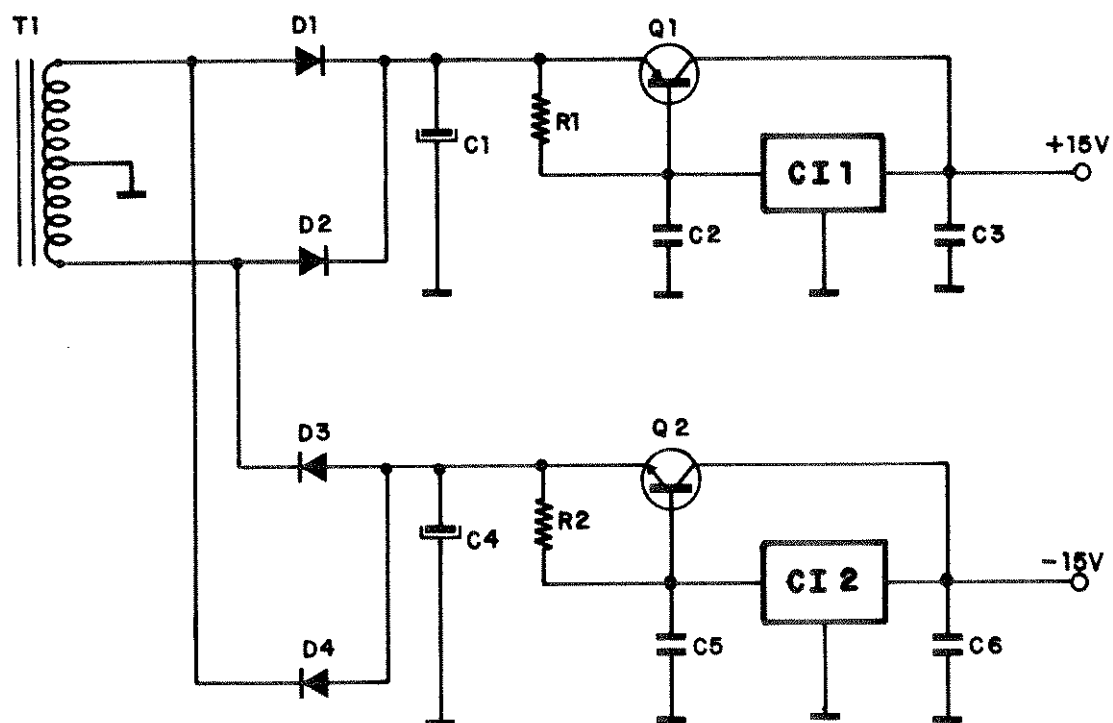
CH<80
(ha, espaco
na tela)
S
G
N

recupera registr.
salvos no inicio
da funcao

RETURN

ANEXO - VIII

Esquema Elétrico do Módulo da Fonte de Alimentação



ANEXO VIII

Relação de Componentes - Anexo VIII

CI₁ = 7805

CI₂ = 7905

CI₃ = LM 323 K

Q₁ = TIP 29

Q₂ = TIP 30

D₁ a D₄ = 1N4001

D₅ = D₆ = SKE 3/02

R₁ = R₂ = 56 Ω , 1/2 W

C₁ = C₄ = 2200 μ F, 40 V, eletrolítico

C₂ = C₃ = C₅ = C₆ = C₈ = C₉ = 100 nF, 25 V, cerâmico

C₇ = 4700 μ F, 25 V, eletrolítico

T₁ = Transformador 18-0-18 @ 2 A

T₂ = Transformador 12-0-12 @ 3 A

ANEXO - IX

Trabalhos Publicados

- HERMINI, A.H. & CALIL, S.J.

"Analisador de Espectro para Resposta Doppler de Sinal Ultra-sônico de Onda Contínua"

Resumos do I Congresso Brasileiro de Informática em Saúde, 1986, p. 1989.

- HERMINI, A.H. & CALIL, S.J.

"Analisador de Espectro para Resposta Doppler de Sinais Ultra-sônicos de Onda Contínua".

Resumo de VI Congresso Argentino de Bioengenharia Tucumán - Argentina, p. 30

- HERMINI, A.H. & CALIL, S.J.

"Técnica para Análise Espectral da Resposta Doppler de Sinais Ultra-sônicos"

RBE - Caderno de Engenharia Biomédica - XII Congresso Brasileiro de Engenharia Biomédica.

Ribeirão Preto SP - Outubro de 1990 - pp. 518-525

- HERMINI, A.H. & CALIL, S.J.

"Spectral Analysis Technique for the Ultrasonic Doppler Shift".

World Congress on Medical Physics and Biomedical Engineering.

16th International Conference on Medical and Biomedical Engineering.

International Federation for Medical and Biological Engineering

Kioto, Japan (07 a 12 de julho de 1991), op-3-D-1-06.

- HERMINI, A.H. & CALIL, S.J.

"Spectral Analysis Techique for the Ultrasonic Doppler Shift"

IV International Symposium on Biomedical Engineering

Peniscola, Spain 1991, pp. 460-461

BIBLIOGRAFIA

- ANALOG DEVICES (1984) - "Integrated Circuits Data Book - vol. I".
- ATKINSON, P.; Woodcock, J.P. (1982) - "Doppler Ultrasound and its use in Clinical Measurement", Academic Press.
- BROWN, P.M.; Johnston, K.W.; Kassan, M.; Cobbold, R.S.C. (1982) - "A Critical Study of Ultrasound Doppler Spectral Analysis for Detecting Carotid Disease", Ultrasound in Medicine and Biology, vol. 8, N.5, Pergamon Press, pp. 512-523.
- COGHLAN, B.A.; Taylor, M.G.; King, D.H. (1973) - "On-line Display of Doppler - shift Spectra by a New Time Compression Analyser", Proceedings of an International Symposium held at Janssen Pharmaceutica, Beerse Belgium, May 29 and 30.
- Di GIULIOMARIA, C.; Capponi M.; D. Alessio, T.; Sacco, R.; Zanette, E. (1990) - " Real - Time System for Robust Spectral Parameter Estimation in Doppler Signal Analysis", Medical & Biological Engineering & Computing, January, pp. 54 - 59.

- EGGBRECHT, L. C. (1983) - "Interfacing to the IBM Personal Computer", Howard W. Sams & Co., First Edition.

- FELIX, Jr. W.R.; Sigel, B.; Popky, G.L. (1975) - "Doppler Ultrasound in the Diagnosis of Peripheral Vascular Disease", Seminars in Roentgenology, vol. x n.4, October.

- FOLMER - JOHNSON, T.N.O. (1988) - "Acústica", Livraria Nobel S.A.

- GOSLING, R.G. (1976) - "Extraction of Physiological Information from Spectrum Analysed Doppler-shifted Continuous - Wave Ultrasound Signal Obtained non Invasively from the Arterial System", IEE Monographs (18-22), pp. 73 - 125.

- GAMBA, H.R. (1989) - "Projeto de um Sistema Ultra-sônico Doppler Pulsátil para Avaliação da Vascularização Periférica" - Tese de Mestrado FEE - UNICAMP.

- HERMINI, A.H.; Calil, S.J. (1990) - "Técnica para Análise Espectral da Resposta Doppler de Sinais Ultra-Sônicos", Revista Brasileira de Engenharia, Caderno de Engenharia Biomédica, vd. 7, n 1, outubro, pp. 518 - 525.

- INTEL (1980) - "Component Data Catalog", Intel Corp.

- LATHI, B.P. (1979) - "Sistemas de Comunicação" Guanabara Dois.

- LUNT, M.J. (1975) - "Accuracy and Limitations of the Ultrasonic Doppler Blood Velocimeter and Zero Crossing Detector" Ultrasound in Medicine and Biology, vol. 2, Pergamon Press pp. 1 - 10.

- MAGNIN, P.A. (1986) - "Doppler Effect: History and Theory", Hewlett Packard Journal, June, pp. 26 - 31.

- NATIONAL (1978) - "Voltage Regulator Handdbook"; National Semiconductor Corp.

- NATIONAL (1982) - "Linear Handbook", National Semiconductor Corp.

- NCRP (1986) - "Biological Effects of Ultrasound: Mechanisms and Clinical Implications", National Council on Radiation Protection and Measurements, Report N 74.

- PINARDI, G.; Sainz, A. (1977) - "El uso de Ultrasonidos Doppler en la Evaluation Cardiovascular", Acta Cientifica Venezolana, n. 28, pp. 308 - 312.

- PRYTHERCH, D.R.; Evans, D.H. (1985) - "Versatile Microcomputer-based System for the Capture, Storage and Processing of Spectral-analysed Doppler Ultrasound Blood Flow Signals", Medical & Biological Engineering & Computing September, pp. 445 - 452.

- RICHARDSON, P.C.A.; Stevens A.L.; Cowan, D.; Calil S.J.; Roberts, V.C. (1987) "Design of a Continuous-wave Doppler Ultrasonic Flowmeter for Perivascular Application - Part 1 - Probe Design", Medical & Biological Engineering & Computing, November, pp. 661 - 666.

- RICHARDSON, P.C.A.; Stevens A.L., Cowan, P.; Calil S.J.; Roberts, V.C. (1988) "Design of a Continuous-wave Doppler Ultrasonic Flowmeter for Perivascular Application - Part 2 - Signal Processing System", Medical & Biological Engineering & Computing, March, pp. 153 - 159.

- ROBERTS, V.C. (1974) - "A Review of Non-Invasive Measurement of Blood Flow" - Biomedical Engineering, August, pp. 332 - 335.

- SCHLINDWEIN, F.S.; Evans, D. H. (1989) - "A Real-Time Autoregressive Spectrum Analyser for Doppler Ultrasound Signals", Ultrasound in Medicine and Biology, vol. 15, n.3, Pergamon Press, pp. 263 - 272.

- SCHEU, H.D. (1972) - "Pulsatile Velocity of Central Arterial Blood", S. Krager AG.

- SILLESEN, H.; Schroeder, T. (1988) - "Changes in Doppler Waveforms Can Predict Pressure Reduction Across Internal Carotid Artery Stenosis", Ultrasound in Medicine and Biology, vol. 14, n.8 Pergamon Press, pp. 649 - 655.

- STOUT, D.F.; Kaufman, M. (1976) - "Handbook of Operational Amplifier Circuit Design" - Mc Graw-Hill

- STEVENS, A.L. (1982) - "Measurement of Blood Flow Using Doppler-shifted Ultrasound", Trans Int M.C., vol. 4, n.2, April-June, pp. 79-91.

- TEXAS (1976) - "The TTL Data Book", Texas Instruments Inc., second edition.

- TEXAS (1980) - "The Linear Control Circuits Data Book", Texas Instruments Inc., second edition.

- THOMPSON, R.S.; Aldis, G.K.; Linnetti, I.W. (1990) "Doppler Ultrasound Spectral Power Density Distribution: Measurements Artifacts in Steady Flow, Medical & Biological Engineering & Computing, January, pp. 60 - 66.

- WIJN, P.F.; Van der Sar, P.; Gootsen, T.H.J.M.; Tilmans, M.H.J.; Skotricbi, S.H. (1987) - "Value of the Spectral Broadening Index in Continuous Wave Doppler Measurements", Medical & Biological Engineering & Computing, July, pp. 377 - 385.
- ZUECH, P.E.; Cobbald, R.S.C.; Johnston, K.W.; Kassan, M. (1984) - "Spectral Analysis of Doppler Flow Velocity Signals: Assessment of Objectives, Methods and Interpretation", Annals of Biomedical Engineering, vol.12, pp. 103 - 116.