ESCOLA FEDERAL DE ENGENHARIA DE ITAJUBÁ

Escola Federal de Engenharia
Biblioteca
MAUÁ
Itajubá

DISSERTAÇÃO DE MESTRADO

IMPLEMENTAÇÃO DE UM MODULADOR PWM PARA MICROCOMPUTADOR

JOSÉ ALBERTO FERREIRA FILHO

ORIENTADORES:

PROF. VALBERTO FERREIRA DA SILVA
PROF. CARLOS ALBERTO MURARI PINHEIRO

ITAJUBÁ - MG
1992

CLASS. 681.31:621.38(043.2)
CUTT. F383i
TOMBO. 691

EFEI / BIBLIOTECA

ESTE LIVRO DEVE SER DEVOLVIDO NA ÚLTIMA DATA CARIMBADA.

12/07/95		
19/9/95		
25/10/95		
01/03/96		
19.4.96		
10 9 96		
27.11.96		
09/08/97		
12/08/97		
22/08/97		
29-8-97		
04/07/02		
02.10.2001		
10.11.2001		
~~[illegible]~~		
08/03/2.004		
[illegible]		
[illegible]		
15/05/06		
26/08/08		

INDICE

Indice i

Dedicatória iii

Resumo iv

Abstract v

Cap.1 Introdução 1

Cap.2 Controle por microprocessador de inversores PWM 4
2.1 Introdução 4
2.2 Resumo das funções do sistema 6
2.3 Implementação com microprocessador do controle PWM 13

Cap. 3 Programas 24
3.1 Introdução 24
3.2 Programa DISP 24
3.3 Programa ANALISE 36

Cap. 4 Hardware 43
4.1 Introdução 43
4.2 Descrição do circuito de disparo 43
4.3 Descrição de funcionamento 46
4.4 Analisador lógico e de espectro 50

Cap.5 Resultados Obtidos 60
5.1 Introdução 60
5.2 Simulação 1 60
5.3 Simulação 2 70
5.4 Simulação 3 71
5.5 Validação dos resultados 72
5.6 Aplicação prática 77

Cap.6 Conclusões 78
6.1 Análise do circuíto 78
6.2 Análise dos resultados 78

Apêndice I 80
Apêndice II 87
Apêndice III 110
Apêndice IV 116

Referências Bibliográficas 130

Dedico este trabalho às mulheres maravilhosas de minha vida : D. Divina , Silvana e Mariana .

RESUMO

O desenvolvimento de técnicas de controle para inversores utilizando modulação de largura de pulso é algo trabalhoso devido ao tempo gasto para se fazer as avaliações e os testes adequados. O modulador por largura de pulso apresentado nesta dissertação foi projetado para atenuar este problema. O modulador, que utiliza um microcomputador, é capaz de gerar os sinais de gatilho para várias técnicas de modulação por largura de pulso (PWM) na condição de operação em malha aberta. Também podem ser geradas e apresentadas as formas de onda das tensões de saída do modulador e o espectro de frequência destas ondas para avaliação da técnica selecionada. Tarefas como a seleção da técnica PWM, das frequências de chaveamento e de curvas tensão/frequência (para acionamento de motores de indução (MIT) com velocidade variável e em malha aberta) são fáceis de serem executadas, com este sistema desenvolvido. Esta ferramenta de pesquisa e desenvolvimento difere de outros dispositivos similares por ser versátil e fácil de usar.

ABSTRACT

Developments in control techniques for pulse- width modulator inverters are so hard because of lack of time for adequate testing and evaluation. The programable PC-based pulse-width modulator described in this tese is designed to alliviate this problem. The modulator is capable of generating gating signals for most types of programmable pulse-width modulator (PWM) schemes under open-loop operating conditions. It can also generate and display associate modulator voltage and their respective frequency spectra for further evaluation of selected PWM schemes. Tasks such as the selection of PWM strategy, switching frequencies and voltage/frequency curves (for variable-speed AC drives under open loop control) are easier to perform. Moreover, the proposed research and development toil differ from similar devices in that it is user friendly, multipurpose and versatile.

CAPÍTULO I
INTRODUÇÃO

As vantagens da modulação de largura de pulso (PWM) em conversão de potência, utilizando dispositivos de estado sólido, são numerosas e bem documentadas [1]-[5]. Essas vantagens tem gerado inúmeras pesquisas trazendo novas técnicas de PWM. A avaliação destas técnicas pode ser facilitada utilizando um modulador flexível e fácil de usar . Este modulador pode incorporar facilmente novas técnicas PWM sendo capaz de gerar e apresentar os sinais de gatilhamento do conversor. Tal modulador é descrito nesta tese.

Este modulador implementa técnicas de PWM que utilizam portadoras. Os sinais gerados poderão ser utilizados em inversores de potência. Um computador pessoal (PC) permite ao usuário selecionar rapidamente qualquer técnica PWM existente em um menu. O usuário define a frequência de chaveamento, o índice de modulação e a frequência de saída. Os sinais gerados poderão ser então aplicados em um conversor. Com algumas pequenas mudanças no software, o usuário poderá adicionar outras técnicas de modulação assim como o disparo para outras topologias de conversores (como por exemplo retificadores com modulação em largura de pulso ou inversores fonte de corrente). Isto é possível devido à programação estruturada e ao emprego de linguagem de alto nível na interface homem-máquina.

Na técnica PWM utilizada neste trabalho a definição dos pontos de chaveamento é feita a partir da interseção entre uma forma de onda portadora e uma forma de onda de referência [1]-[3]. Outras técnicas, como por exemplo, a técnica de eliminação seletiva de harmônicos [4],[5] podem ser incorporadas ao programa com grande facilidade.

As técnicas aqui empregadas ignoram as propriedades da carga ou interação entre o conversor e a carga, tratando os conversores

como uma fonte ideal de potência. No entanto, existem vários procedimentos que definem estratégias que levam em conta as características da carga [6]-[8]. Esses procedimentos estão fora do objetivo deste trabalho.

O modulador é capaz de comportar técnicas de modulação mais avançadas. Isto é possível pois os algorítmos que implementam estas técnicas exigem grande poder computacional e tal poder está disponível nos computadores pessoais hoje exitentes. Esses computadores possuem grande capacidade de memória, aceitam linguagem avançadas de programação(como a linguagem C e PASCAL) além de possuirem circuítos poderosos como co-processadores aritméticos, controladores de acesso direto à memória e gerenciadores de interrupções.

A validação do modulador foi comprovada por análise de espectro dos sinais PWM, utilizando um analisador de espectro. Essa análise é feita em tempo real a partir dos sinais gerados pelo sistema. Para isto foi utilizado um módulo [9] que colhe 2048 amostras dos sinais de gatilhamento gerados. A partir disto um programa executa a Tansformada Rápida De Fourier (FFT) dos sinais calculando o espectro de frequência. O analisador de espectro é sem dúvida uma ferramenta indispensável para estudar os sinais PWM.

Embora não tenha sido desenvolvido nesta tese, exite um conjunto de programas de simulação [10] que permite ao usuário simular as técnicas PWM abordadas no modulador. Os programas de simulação mais o modulador formam um conjunto completo para pesquisa e desenvolvimento. Além disto, este conjunto poderá ser utilisado em cursos de eletrônica de potência oferecidos em disciplinas de graduaçao e pós-graduação.

O módulo de aquisição de dados também possui a capacidade de funcionar como um analisador lógico. Neste modo de operação o equipamento apresenta como estão as formas de onda no domínio do

tempo. A operação neste modo é importante pois é possível conferir se a lógica de disparo está correta.

No Capítulo II será dado uma visão geral das técnicas de modulação PWM, sua evolução no decorrer dos anos e as várias maneiras de serem implementadas utilizando circuítos analógicos e digitais. No Capitulo III será feita uma descrição dos programas (Software) que foram utilizados para a confecção do modulador. O Capítulo IV irá descrever o circuíto que implementa o sistema de modulação. Várias técnicas de disparo PWM serão estudadas no Capítulo V.

CAPÍTULO II
CONTROLE POR MICROPROCESSADOR DE INVERSORES PWM

2.1 INTRODUÇÃO:

Atualmente as vantagens operacionais dos inversores PWM são bem conhecidas e aceitas. Essas vantagens geralmente são conseguidas através de circuítos de controle e de potência muito complexos. O potencial e a versatilidade das técnicas de PWM ainda não foram totalmente exploradas. No passado essas técnicas foram limitadas pelos circuítos analógicos e digitais disponíveis. O grande avanço na tecnologia dos circuítos digitais, no desenvolvimento de tiristores de chaveamento mais rápidos e o desenvolvimento de transistores de alta potência , tem eliminado muitas dessas restrições práticas e agora tornam possível explorar ao máximo as técnicas de controle PWM.

Controlar equipamentos de eletrônica de potência por meio de microprocessadores está se tornando comercialmente atrativo por oferecer a possibilidade de aumentar a confiabilidade e a flexibilidade, principalmente em acionadores (drivers) onde haverá controle de velocidade. Essas vantagens resultam da redução da complexidade do circuíto de controle que pode ser repassado para o software do microprocessador. Deste modo é possível mudar as características do driver sem alterar o hardware. O objetivo de todo projeto utilizando microprocessador deve ser implementar o máximo possível de funções de controle através de software, aumentando deste modo a confiabilidade e reduzindo os custos de hardware e de produção. A manutenção dos equipamentos de potência é beneficiada quando se usa microprocessador. Muitas funções de monitoramento de falhas e de diagnose podem ser implementadas por meio de software. Por exemplo, sistemas com grandezas variáveis tais como tensão, corrente, e velocidade podem ser monitorados e essas grandezas armazenadas na memória antes de uma falha. Estas informações

podem então serem utilizadas para alarmes, bloqueios e diagnósticos.

A implementação do circuito de controle pode ser feita utilizando microprocessadores ou circuítos analógicos. Contudo, para se obter todas as vantagens que o projeto em software tem sobre o projeto em hardware e ainda fazer um uso efetivo do poder computacional do microprocessador, é necessário uma nova abordagem do projeto do sistema.

As limitações de projeto dependerão do tipo de implementação realizada: se analógica ou com microprocessador. Deste modo, são necessárias cuidadosas considerações sobre as implicações de várias limitações de hardware ou software e seus efeitos na velocidade,precisão e temporização das malhas de controle. Isto tudo para obter a solução ótima de projeto. As novas tecnologias em microprocesadores tem renovado as filosofias de controle. Microprocessadores mais velozes tem permitido reguladores de alta velocidade de resposta aumentando assim o desempenho do sistema.

Em seguida serão analizadas as considerações essenciais envolvidas no projeto de um inversor PWM controlado por microprocessador, relacionando-as com as tecnologias de micro atuais e futuras. É dada ênfase na diminuição dos requisitos de hardware, com a maioria das funções sendo implementadas em software. Um critério adicional é que uma estrutura de programa eficiente deve ser usada para minimizar o tempo computacional requerido, aumentando deste modo a velocidade operacional do sistema e liberando o microprocessador para realizar outras operações.

Antes de considerar o projeto de um sistema baseado em microprocessador, é necessário resumir os requisitos mínimos para se implementar um controlador PWM.

2.2 RESUMO DAS FUNÇÕES DO SISTEMA:

Antes de se fazer as considerações detalhadas das técnicas PWM implementadas por microprocessador, é importante neste estágio rever rapidamente os requisitos funcionais mais importantes ligados aos acionadores.

A Fig. 2.1 fornece uma visão geral de uma hierarquia de controle por microprocessador e distingue algumas funções necessárias para um sistema PWM [1]-[5]. Dependendo do tipo de microprocessador utilizado, apenas um número pequeno de funções poderão ser realizadas. Para sistemas mais complexos, onde várias funções serão realizadas, deverá ser construído um sistema com vários processadores, cada um realizando uma função diferente. A seguir serão descritas as principais funções indicadas na Fig. 2.1.

Funções de Gerênciamento do Sistema

Monitorando os parâmetros que definem o estado atual do sistema e implementando funções de gerênciamento adequadas é possível incorporar através de microprocessadores vários requisitos funcionais do acionador. Por exemplo, podem ser incoporados procedimentos de inicialização e bloqueio. Também podem ser implementadas aceleração e desaceleração controladas, com a possibilidade de acrescentar controle de gerenciamento de energia para minimizar, por exemplo, a demanda máxima ou operar com fator de potência máximo. As funções de gerenciamento devem acomodar protocolos de operações de modo que um sistema possa ser interligado com outros sistemas controlados por computadores. A operação por meio de estações remotas também pode ser incluída.

A especificação das funções de gerenciamento dependem muito do tamanho do sistema, da natureza da aplicação e do desempenho necessário.

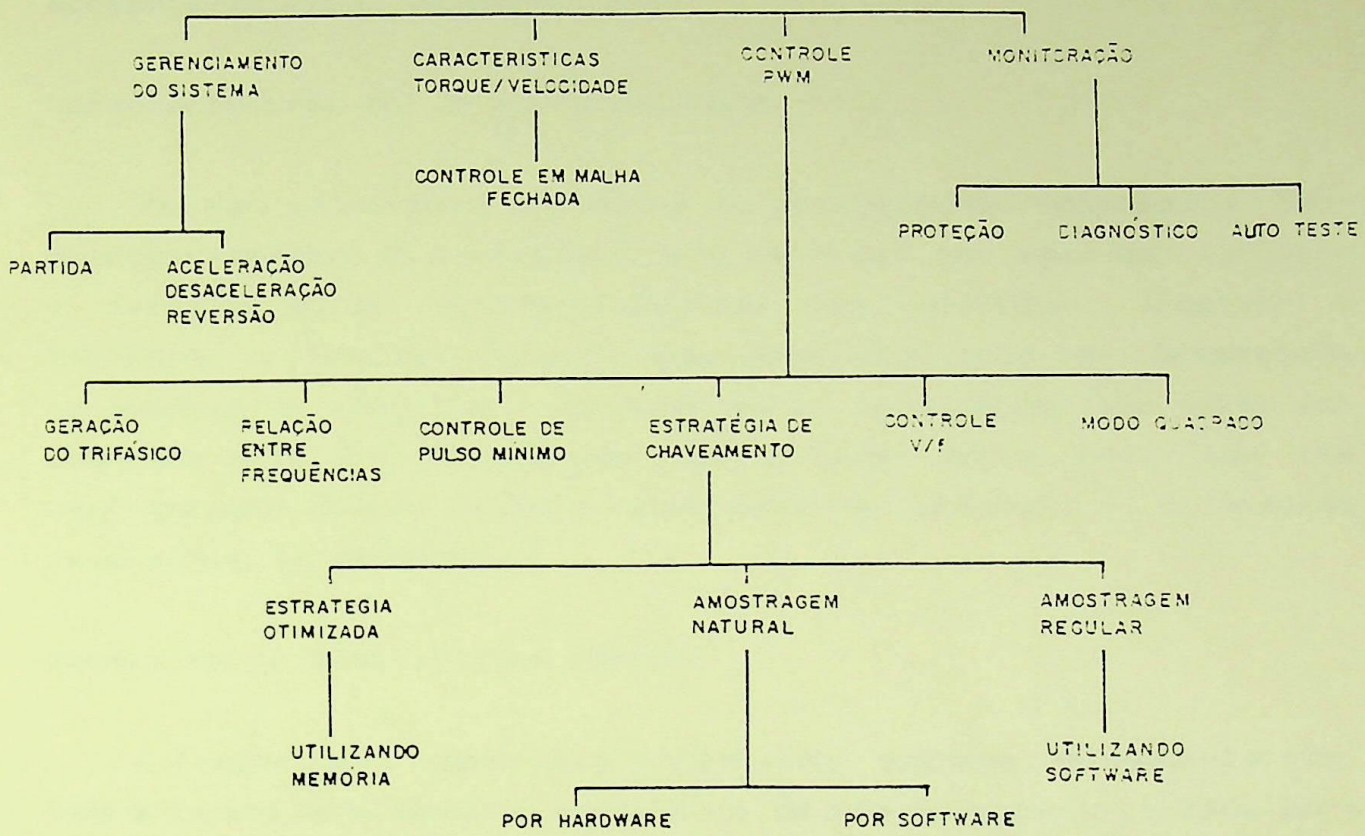


Fig. 2.1 Hierarquia de um sistema PWM de controle

Funções de Monitoração e Diagnóstico [20]

Uma das vantagens principais de um sistema que utiliza microprocessador é a capacidade de amostrar variáveis e a facilidade para armazenar essas variáveis na memória para recuperá - las mais tarde. Assim é possível identificar condições apropriadas para alarme ou bloqueio do controle. Por meio de um banco de dados é possível recuperar informações logo após o bloqueio do sistema de modo a serem utilizadas para finalidades de diagnose. É possível também utilizar as funções de monitoração para detectar deterioração no desempenho medindo ,por exemplo, a

tensão, corrente, temperatura, etc., em várias partes do sistema. O microprocessador pode ter a capacidade de monitorar a deterioração no desempenho dos componentes e tomar decisões apropriadas para compensar isto.

Características de Torque/Velocidade

Uma das principais vantagens de acionadores controlados por microprocessador é a possibilidade de mudar por meio de software a característica Torque/Velocidade sem precisar alterar o hardware. A característica Torque/Velocidade pode ser armazenada na memória de modo que a característica mais apropriada possa ser acessada pelo microprocessador, permitindo deste modo que uma configuração básica tenha a capacidade de atender a diferentes requisitos de desempenho.

Requisito do Controle com PWM[20]

A função mais importante em qualquer esquema de controle com PWM é indubitávelmente a estratégia de chaveamento utilizada para gerar as bordas dos pulsos. Deve-se portanto explorar as várias opções de hardware/software associadas com a implementação de uma estratégia de modulação e avaliar essas opções em termos da velocidade de chaveamento e resposta do sistema. Além dos requisitos para uma implantação eficiente com microprocessador da estratégia de chaveamento, é necessário que o acionador incorpore certas funções que o tornem viável praticamente.

Operação com três Fases

Este requisito pode ser facilmente obtido usando um microprocessador. Por exemplo, um quadrante de uma onda senoidal pode ser armazenado na forma de uma tabela na memória de modo a ser lida utilizando 3 ponteiros. A mudança no sentido de rotação pode ser obtida simplesmente trocando dois ponteiros ou alterando a direção na qual os 3 ponteiros percorrem a tabela. O software

tensão, corrente, temperatura, etc., em várias partes do sistema. O microprocessador pode ter a capacidade de monitorar a deterioração no desempenho dos componentes e tomar decisões apropriadas para compensar isto.

Características de Torque/Velocidade

Uma das principais vantagens de acionadores controlados por microprocessador é a possibilidade de mudar por meio de software a característica Torque/Velocidade sem precisar alterar o hardware. A característica Torque/Velocidade pode ser armazenada na memória de modo que a característica mais apropriada possa ser acessada pelo microprocessador, permitindo deste modo que uma configuração básica tenha a capacidade de atender a diferentes requisitos de desempenho.

Requisito do Controle com PWM[20]

A função mais importante em qualquer esquema de controle com PWM é indubitávelmente a estratégia de chaveamento utilizada para gerar as bordas dos pulsos. Deve-se portanto explorar as várias opções de hardware/software associadas com a implementação de uma estratégia de modulação e avaliar essas opções em termos da velocidade de chaveamento e resposta do sistema. Além dos requisitos para uma implantação eficiente com microprocessador da estratégia de chaveamento, é necessário que o acionador incorpore certas funções que o tornem viável praticamente.

Operação com três Fases

Este requisito pode ser facilmente obtido usando um microprocessador. Por exemplo, um quadrante de uma onda senoidal pode ser armazenado na forma de uma tabela na memória de modo a ser lida utilizando 3 ponteiros. A mudança no sentido de rotação pode ser obtida simplesmente trocando dois ponteiros ou alterando a direção na qual os 3 ponteiros percorrem a tabela. O software

de controle pode ser simplificado aumentando a velocidade de processamento, caso a tabela completa da onda senoidal seja armazenada na memória. Isto ocasionará um maior consumo de memória.

As formas de onda trifásicas e de frequência variável, tambem podem ser implementadas de modo analógico. O circuito analógico necessário para obter este requisito é geralmente mais oneroso e complexo,por envolver um grande número de componentes discretos além de apresentar problemas nas precisões da fase e da frequência. Além disto este circuito se torna menos confiável do que a implementação equivalente em microprocessador.

Requisito V/F

Muitos acionadores de frequência variável operam com torque constante em uma faixa de frequência abaixo de 50/60 Hz. Isto é geralmente obtido mantendo constante a relação V/F (fluxo constante) dentro de grande parte da faixa de frequência, com a tensão sendo aumentada adicionalmente a relação V/F nas baixas frequências (5Hz a 10 Hz)para compensar o aumento nas perdas da tensão do estator. É frequentemente necessário incorporar uma saída com tensão constante em frequências maiores que 50/60 Hz, a fim de produzir a operação em potência constante. As características V/F como estas podem facilmente ser incorporadas por intermédio de software no microprocessador.

Em alguns casos é desejado uma característica V/F não linear. Isto pode se tornar muito difícil de se obter utilizando circuítos analógicos. Com o uso de microprocessador, estas características especiais do acionador podem ser facilmente programadas utilizando operações aritméticas ou tabelas na memória.

A capacidade de programação do microprocessador de modo a produzir diferentes características de acionamento é

comercialmente atrativa, pois permite que a mesma configuração seja usada para vários tipos de aplicações diferentes.

Controle da Frequência de Chaveamento [6]-[15]

Para utilizar completamente as caracteristcas de eliminação de harmônicos do controle PWM, é desejável operar o inversor na máxima frequência de chaveamento permitida, na maior faixa de frequência de saída possível. A frequência máxima de chaveamento é geralmente restringida pelas limitações do circuíto de potência e deste modo é inevitável um compromísso entre a eficiência do circuíto de potência e a minimização dos harmônicos.

A frequência máxima de chaveamento (frequência da portadora) deve ser coerente com as limitações do circuíto de potência. Ela é determinada para o pior caso , ou seja , quando o inversor estiver operando na frequência máxima de saída. Deve-se aumentar a relação entre as frequências (frequência de chaveamento / frequência de saída) a medida que a frequência de saída do inversor é reduzida . Isto assegura que a distorção harmônica na tensão do inversor seja reduzida a medida que diminui a frequência de saída, mantendo-se os efeitos dos harmônicos no motor dentro de limites aceitáveis. A relação entre a frequência de chaveamento e a frequência de saída será denominada a partir de agora de " relação p " .

No passado eram usados dois métodos de controle da relação p (ou controle de harmônico). O primeiro método usa uma única frequência de chaveamento dentro de toda faixa de variação da frequência de saída do inversor. Este método de controle faz com que a relação p aumente continuamente a medida que a frequência de saída reduz. Este método é conhecido como modo de controle de relação "assíncrono" ou "não inteiro", porque a frequência de chaveamento não é sincronizada com a frequência de saída [6],[14] Embora o modo de controle assíncrono possa ser implementado facilmente utilizando circuítos analógicos, é importante notar

que quando a relação p fica abaixo de nove ocorre o aparecimento de sub-harmônicos [14] ou "frequência de batimento" nos casos em que não há um sincronismo entre as frequências modulante e portadora.

O segundo método de controle conhecido como " modo da relação inteira e discreta " [12],[13] consiste em manter um sincronismo entre a frequência de chaveamento e a frequência de saída. Este modo de controle requer que a relação p seja mudada em passos inteiros e discretos dentro da faixa de variação da frequência de saída .Deste modo a frequência de chaveamento correspondente terá valores máximos e mínimos para cada valor de p.É importante notar que com este modo de controle é necessário o cuidado em manter a mesma tensão PWM antes e depois da transição para evitar transitórios de chaveamento no acionador . O modo síncrono de controle é mais adequado para implementação digital e pode ser programado no software do microprocessador usando uma tabela para armazenar as relações p necessárias de modo a produzir as frequências de saída necessárias .

Controle da Largura Mínima do Pulso

Na prática é necessário assegurar que o tempo mínimo de atraso das chaves de potência [11],[17] seja mantido de modo a atender os requisitos de comutação do inversor. A medida que o índice de modulação chega perto da unidade e durante a operação multi-modo quando é usado , por exemplo , sobremodulação , aparecem pulsos na tensão PWM com larguras menores que um certo valor mínimo .Esses pulsos devem ser eliminados da forma de onda PWM .Este processo é conhecido como " eliminação de pulsos " e a largura dos pulsos que serão eliminados dependem da configuração do circuíto de potência utilizada .

Para operações confiáveis do inversor é essencial que qualquer estratégia teórica de controle incorpore este requisito.Neste caso, a característica tensão/frequência deve ser

modificada adequadamente levando em consideração a eliminação dos pulsos.

É possível implementar no software a restrição da largura mínima de pulso para atender a um circuíto de potência particular e aos requisitos do acionador.

Estratégia de Chaveamento

O coração de qualquer controle de inversor PWM é sem dúvida a estratégia de chaveamento usada para gerar as bordas dos pulsos da forma de onda de controle. É possível, analisando a literatura nas últimas décadas , traçar o desenvolvimento histórico das técnicas de chaveamento e relacionar este desenvolvimento com as mudanças na tecnologia. Começando com os sistemas analógicos [11]-[16] progrediu-se para os sistemas digitais discretos [18]-[20], passando pelos sistemas que utilizam ROM [21]-[25] terminando nos esquemas implementados com microprocessador [26]-[39].

As estratégias de chaveamento podem ser agrupadas em três conjuntos distintos , apresentados na Fig. 2.1 .

A estratégia conhecida como " Estratégia ótima De Chaveamento " é baseada na minimização de certos critérios de desempenho [40]-[54] , como por exemplo , eliminação ou minimização de harmônicos particulares, minimização da distorção harmônica de corrente ,do ripple de torque e do pico de corrente. Esta estratégia representa a melhor abordagem que pode ser obtida para o controle do inversor . A estratégia otimizada envolve o uso de grande esforço computacional e algoritmos de minimização para determinar os ângulos otimizados de chaveamento.Estes ângulos são posteriormentes armazenados na memória do microprocessador e usados para gerar em tempo real a estratégia PWM de chaveamento. Essas estratégias de chaveamento otimizadas são particularmente atrativas para valores baixo de p,onde

produzem um efeito espetacular na composição de harmônicos e oferece a possibilidade de aumentar a faixa viável da frequência de saída do acionador de velocidade variável . Embora as vantagens de se usar a estratégia otimizada sejam bem conhecidas há muito tempo, foi apenas com o desenvolvimento do microprocessador que esta técnica tornou-se possível .

Infelizmente, a implementação da estratégia otimizada requer um considerável gerenciamento de memória. Vários estudos tem sido feitos no sentido de desenvolver algorítmos mais simples para gerar a estratégia otimizada em tempo real, por meio de microprocesasador.

Muitas implementações de controle PWM são baseadas na estratégia de chaveamento com amostragem "natural". Outra estratégia importante conhecida como "amostragem regular" [55] tem sido proposta contendo várias vantagens, quando implementada utilizando técnicas digitais. As técnicas de amostragem natural e de amostragem regular podem ser implementadas utilizando microprocessadores .Estas duas técnicas formam a base da maioria dos inversores PWM .

É muito importante comparar estas duas técnicas para obter informações de grande valor na escolha da técnica a ser escolhida.

A seção seguinte irá se preocupar com uma investigação detalhada de implementações com microprocessador associadas com cada método de amostragem. Serão avaliadas as principais vantagens e desvantagens de cada técnica de modo a estabelecer as melhores configurações de hardware/software mais adequadas para cada caso.

2.3 IMPLEMENTAÇÃO COM MICROPROCESSADOR DO CONTROLE PWM

2.3.1 Amostragem Natural

Uma técnica para se fazer o controle PWM de inversores é a técnica conhecida na literatura como " técnica de amostragem natural " [55], [56] .Uma implementação prática mostrando as características gerais deste modo de amostragem está ilustrada na Fig. 2.2 . Desta figura , pode-se ver que o sinal de amostragem é a forma de onda portadora triangular. Esta forma de onda é comparada diretamente com a forma de onda modulante senoidal para determinar os instantes de chaveamento e deste modo as larguras dos pulsos . Este método de amostragem foi muito empregado no passado por causa de sua simplicidade e facilidade de implementação quando se utilizavam técnicas analógicas .

É importante notar que a largura do pulso resultante é proporcional à amplitude da onda modulante no instante em que ocorre o chaveamento. Isto ocorre porque a transição do pulso é determinada pela interseção instantânea de duas ondas .Logo existem duas consequências importantes : a primeira é que os centros dos pulsos na onda PWM resultante não são equidistantes ou uniformemente espaçados e segundo , não é possível definir a largura dos pulsos usando expressões analíticas lineares.Entretanto , é possível mostrar [55],[56] que as larguras dos pulsos só podem ser definidas usando equações transcedentais da forma :

$$t2-t1 = \frac{T}{2}\left[1 + \frac{M}{2}\left[\text{Sen } W_n\, t1 + \text{Sen } W_n\, t2 \right]\right] \quad (2.1)$$

Por causa do relacionamento transcendental existente entre os instantes de chaveamento torna-se muito difícil usar o microprocessador para calcular a largura dos pulsos modulados. Naturalmente é possível simular o processo analógico de amostragem natural em um microprocessador, e , usando as funções lógicas determinar as larguras dos pulsos ; esta abordagem será explicada mais detalhadamente nas seções seguintes .

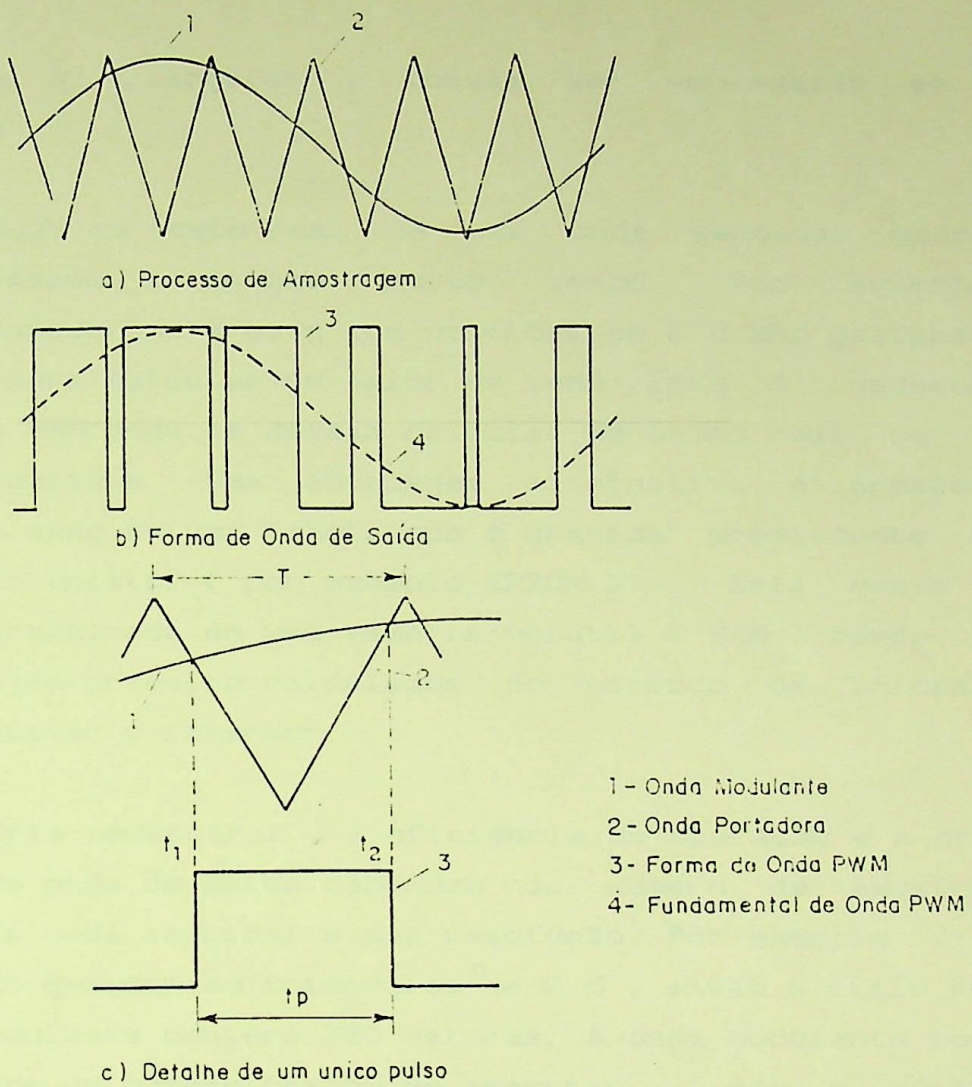


Figura 2.2 PWM com amostragem natural

Nas sub-seções seguintes será dado uma atenção especial para explorar as várias opções com microprocessador, começando com os esquemas que utilizam em predominancia o software . Isto fornecerá as bases para comparar as várias opções .Serão estudados também os meios para se gerar por software as ondas modulantes para o sistema trifásico .

Geração de Onda Modulante por meio de Microprocesador [8],[26]

Ondas de modulação senoidal são muito usadas em técnicas PWM, embora outros tipos de ondas modulantes, tais como

trapezoidal ou triangular , possam ser empregadas em certas aplicações .

A geração em tempo real de uma onda senoidal usando um microprocessador consome muito tempo. Por exemplo, um microprocessador de 8 bits com um clock de 2.5 Mhz gastará cerca de 800 μs para calcular um valor de seno .Isto é inadequado em aplicações PWM onde se deseja calcular em tempo real os valores da onda modulante. Uma abordagem alternativa é armazenar os valores do seno em uma tabela que é gravada previamente em uma memória não volátil (por exemplo EPROM) . Esta mesma tabela pode ser armazenada em uma memória volátil (RAM) desde que os valores sejam primeiro calculados no periodo de inicialização antes de ativar o inversor .

A memória necessária , a eficiência de operação e a precisão da forma de onda de saída dependem do número de amostras que definirão a onda senoidal e sua resolução. Por exemplo , se os valores são gerados em intervalos ^{o}de 0.5 , então o ciclo completo de onda modulante conterá 720 valores. A onda modulante pode ser definida com um número grande de amostras , mas o consumo de memória aumentará proporcionalmente. Se a estratégia de mudança de modos é utilizada , particularmente na amostragem regular , é conveniente fazer com que a quantidade de pontos da tabela seja múltiplo da relação p .Se isto ocorrer, o instante no qual ocorrerá a amostragem para o processo de modulação corresponderá diretamente a um valor na tabela. Se é gerada uma portadora de uma única fase, então cada relação de frequência deverá ser um múltiplo de três para eliminar no inversor os harmônicos da frequência portadora. Por exemplo , se 720 valores definem um ciclo de modulação , então a relação entre frequências multiplas de 3 deverão ser utilizadas , por exemplo , 180 , 120 , 90 , 60 , 45 , etc ...

A resolução de cada tabela deverá ser especificada tendo como base o tipo de modulação a ser realizada e a precisão

necessária na largura dos pulsos modulados . Para muitas aplicações uma precisão de 8 bits é suficiente onde pode-se usar então um microprocessador de 8 bits .

Os requisitos de memória são reduzidos se apenas uma metade ou um quarto do ciclo modulante é armazenado na tabela . Quando se utiliza um quarto do ciclo aumentam-se os bits de resolução de 1 pois não é necessário indicar o sinal de cada amostra armazenada .

Os valores que serão acessados na tabela serão indicados utilizando-se um ponteiro . Se for utilizado um ciclo na memória. então o ponteiro circulará por toda tabela . Um procedimento similar ocorrerá se apenas metade do ciclo da onda modulante é armazenado na tabela ; embora neste caso seja necessário incluir um flag para indicar se o valor do seno obtido é positivo ou negativo. Este flag também é necessário se apenas um quarto de onda modulante é utilizado . Neste caso o ponteiro se movimentará para cima e para baixo ao invés de se movimentar em uma única direção .

A operação em três fases pode ser obtida facilmente tendo uma única tabela com três ponteiros , cada um apontando para um terço do ciclo .

Sistemas Utilizando Hardware [21]

Neste sistema , o microprocessador é usado apenas para gerar a onda modulante de frequência variável ; o processo de amostragem natural e a geração do PWM são realizados por circuítos externos ao microprocessador. Uma configuração típica deste sistema está ilustrada na Fig. 2.3 . Neste caso , o microprocessador é interrompido por um clock de frequência variável e os valores amostrados da onda de modulação são armazenados na memória na forma de uma tabela . Durante cada

interrupção , a memória é acessada e o valor amostrado é enviado ao hardware externo . Esta operação leva tipicamente 40 μs para ser processada usando um microprocessador Z-80 da Zilog com um clock de 2.5 Mhz . Se, por exemplo, um ciclo completo da onda de modulação ocupa 256 localizações na tabela , então a frequência máxima de modulação obtida do microprocessador será aproximadamente de 100Hz .

A frequência de chaveamento PWM pode ser controlada usando um contador crescente/decrescente para gerar uma onda triangular portadora . Esta onda é comparada com a onda modulante amostrada num comparador digital produzindo a forma de onda PWM . Se o contador crescente / decrescente é chaveado por um clock separado, então é gerado o PWM assíncrono e a frequência máxima de chaveamento será limitada pela capacidade do circuíto de potência do inversor .

Se é necessário operar o PWM no modo síncrono , então isto poderá ser feito reinicializando o contador crescente / decrescente no inicio de cada ciclo modulante ou obtendo o sinal de interrupção para o microprocessador a partir do clock do contador crescente / decrescente .

O controle de tensão pode ser obtido através de mudanças no índice de modulação da forma de onda PWM . Isto envolve uma multiplicação que , se realizada em software , levará tipicamente 200 μs , diminuindo de modo drástico a frequência máxima de saída. Esta multiplicação pode ser implementada em hardware reduzindo o tempo de cálculo para 5 μs tipicamente , como mostrado na Fig. 2.3 .O índice de modulação pode ser fornecido para o multiplicador diretamente pelo microprocessador ou por uma outra fonte independente como o sinal de realimentação em um sistema de tensão regulada .

O sistema monofásico mostrado na Fig. 2.3 pode ser extendido para operação em sistemas trifásico , simplesmente multiplexando

o multiplicador e adicionando outros comparadores .Contudo, no caso trifásico é necessário acessar três valores de amostras na tabela , usando para isto três ponteiros como discutido anteriormente. Se , por exemplo , é usado um microprocessador Z-80 (Zilog) com um clock de 2.5 Mhz gastando tipicamente 40μs/amostra, e um número inteiro de amostras multiplos de três é armazenado na memória, digamos 255, então a frequência máxima de modulação estará limitada em 30 Hz . Esta frequência pode ser estendida para 50 Hz se é utilizado um clock de 4 Mhz, levando o tempo de acesso das amostras para 25μs/amostra .Pode-se aumentar a frequência máxima de saída diminuindo o número de amostras por ciclo armazenadas na tabela ; contudo isto irá aumentar o erro de quantização associado com o valor amostrado .

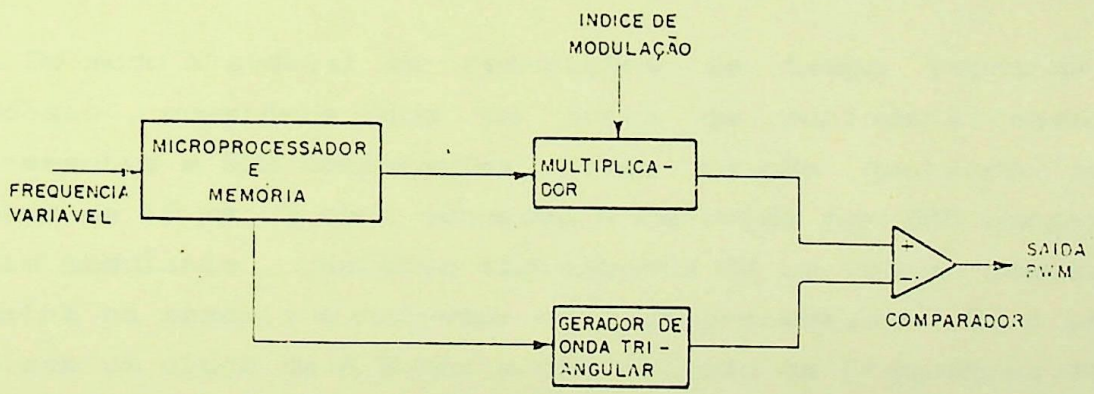


Fig. 2.3 Geração de onda PWM utilizando hardware

A geração dos sinais PWM utilizando hardware não é flexível, uma vez que a configuração do circuito precisa ser mudada se sua operação necessita de modificações . Uma abordagem alternativa é implementar a geração dos sinais por meio de software . Os

problemas associados com este método de implementação são discutidos na próxima seção .

Sistemas Baseados Em Software

Antes de mais nada é necessário considerar as restrições de tempo impostas quando a geração de sinais PWM em tempo real são implementadas por meio de software . A implementação por software da onda modulante já foi descrita na seção anterior. A onda portadora triangular pode ser gerada por meio de software usando um contador crescente / decrescente . A taxa pela qual o contador será incrementado (ou decrementado) determina a frequência da onda portadora e a precisão do processo de amostragem . Cada vez que o contador é incrementado (ou decrementado) , sua saída é comparada com o valor amostrado da onda modulante para determinar os instantes de chaveamento da forma de onda PWM . Este processo está ilustrado na Fig.2.4 .

De modo a avaliar as restrições de tempo impostas neste processo , considere que um ciclo da portadora envolva 256 incrementos e 512 comparações , cada função gastando um tempo típico de 12 μs. A onda senoidal é definida por 255 amostras por ciclo modulante , gastando tipicamente 32 μs para acessar cada amostra na memória e carregar no microprocessador . Se é usado um Z80 com um clock de 2.5 Mhz e uma relação de frequência de 12 , então a frequência de modulação ficará limitada a 12 Hz.

No caso do sistema trifásico é necessário um esforço computacional adicional , reduzindo substancialmente a frequência de saida . É possível aumentar a frequência utilizando um clock de 4Mhz e um número reduzido de amostras na onda portadora. Contudo, são impostas várias restrições na temporização e precisão da forma de onda PWM .

2.3.2 - Amostragem Regular

Uma abordagem alternativa pode ser usada baseada no cálculo dos ângulos de chaveamento [61] ao invés dos tempos de chaveamento, de modo a não involver uma geração em tempo real da onda PWM . A largura dos pulsos de modulação correspondente a um ciclo podem ser determinados para um índice de modulação e uma determinada frequência , sendo então armazenados em memória . As larguras dos pulsos podem ser determinadas usando o processo descrito anteriormente para a geração em tempo real, sendo que neste caso o cálculo destas larguras não são feitos em tempo real. Durante o tempo em que este cálculo está sendo realizado , uma outra tabela deve conter os ângulos atuais para gerar a forma de onda PWM . Uma vez que a tabela é totalmente calculada , correspondente a uma nova frequência ou índice de modulação , os conteúdos da tabela são trocados em um ponto apropriado ,normalmente no fim do ciclo de modulação para evitar transitórios de corrente [13]. O conteúdo das tabelas podem ser trocados ou haver uma simples troca de ponteiros o que será mais fácil e levará menos tempo .

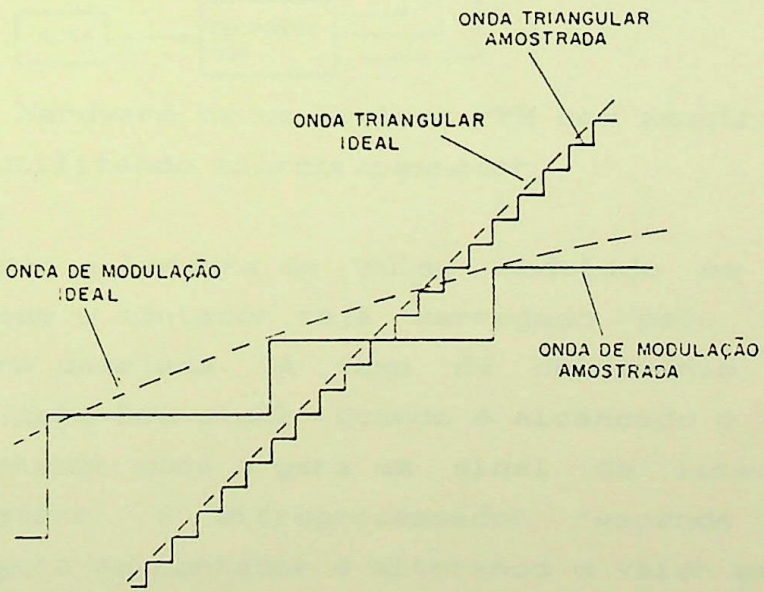


Fig. 2.4 Ilustração do processo de amostragem

Neste esquema [59] o microprocessador lê o novo valor de frequência por meio de um recenseamento em determinada porta de entrada de dados. A nova frequência é então combinada com o índice de modulação de modo a determinar a largura dos pulsos para serem gerados em tempo real . Neste caso deverá ser usado um temporizador juntamente com o microprocessador . Um esquema adequado está mostrado na Fig. 2.5 , onde um temporizador 8253 da INTEL foi utilizado para gerar os pulsos de gatilhamento . O temporizador contém três contadores de 16 bits , que podem ser programados para operarem em seis modos diferentes .

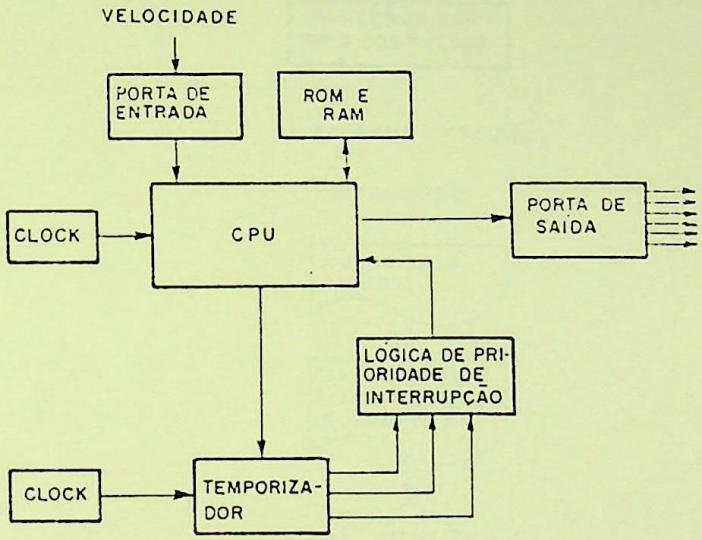


Fig. 2.5 Hardware de um Sistema PWM com amostragem regular utilizando microprocessador

Para gerar a largura do pulso modulado em tempo real é necessário que o contador seja carregado pelo microprocessador com a largura desejada . A taxa de decremento do contador é determinada pelo seu clock . Quando é alcançado o valor zero , a saída do contador muda e gera um sinal de interrupção para o microprocessador . O microprocessador responde armazenando a próxima largura no contador e alterando o valor em uma porta de saída gerando deste modo a forma de onda PWM . A rotina de interrupção por software usada para servir a interrupção do temporizador é mostrada no fluxograma da Fig. 2.6 , junto com as funções do programa principal usadas para calcular a largura dos pulsos.

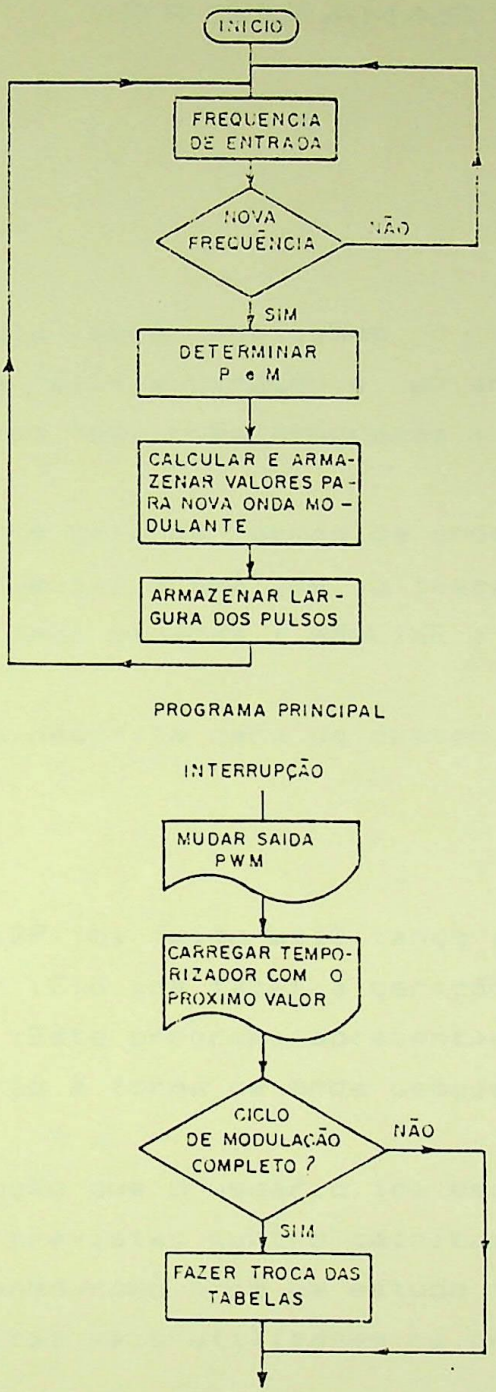


Fig. 2.6 Fluxograma de um sistema baseado em software

É importante notar que , usando esta abordagem , o tempo necessário para calcular os ângulos de chaveamento não deve ser maior que a frequência de saída . Contudo, um atraso em tempo é inevitavelmente introduzido entre a leitura do valor de uma nova frequência ou índice de modulação e esta nova condição ser transferida efetivamente para o conversor .Isto limita a resposta do sistema .

CAPÍTULO III
PROGRAMAS

3.1 INTRODUÇÃO:

Neste capítulo será analisado o conjunto de programas utilizados para se fazer a geração e a analise das ondas PWM. Basicamente temos os seguintes programas :

a) Programa que gera as formas de onda em tempo real - DISP;
b) Programa que faz a análise no tempo e na frequência das formas de onda geradas - ANALISE ;

A seguir será descrito cada um destes programas .

3.2 PROGRAMA DISP

O programa DISP foi feito utilizando a linguagem PASCAL e linguagem ASSEMBLY .Ele irá fazer a geração das formas de onda PWM em tempo real .Este programa apresentará um menu de opções que guiará o usuário à forma de onda desejada .

A primeira opção que o usuário irá escolher será a técnica PWM desejada .São previstas quatro técnicas .Estas técnicas foram pré-programadas tendo como base um estudo feito onde se procurou verificar as técnicas mais utilizadas na prática .

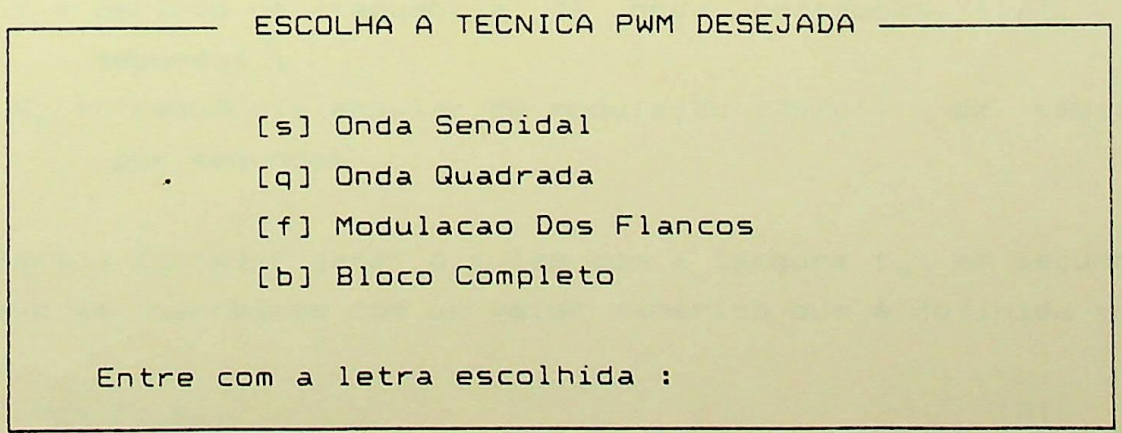


Fig. 3.1 Primeiro Menu de opções do programa DISP .

A Fig. 3.1 apresenta o primeiro menu de opções do programa DISP.Como mostrado no menu , o programa irá apresentar quatro técnicas .

3.2.1 - Técnica De Modulação Senoidal

Na técnica denominada "Onda Senoidal" tem-se uma onda portadora triangular sendo modulada por uma onda senoidal sincronizada.

Nesta técnica , a largura do n-ésimo pulso é dada por [61]:

$$t_n = \frac{T}{2}\left\{1 + (-1)^{n+1}\frac{M}{2}\left\{\mathrm{Sen}\left[(n - 1.5)\frac{T\,W_n}{2}\right] + \mathrm{Sen}\left[(n-0.5)\frac{T\,W_n}{2}\right]\right\}\right\} \qquad (1)$$

Onde:

$n = 2,3,4,\ldots \frac{(p+1)}{2}$

f_r = frequência do clock do contador, em hertz ;
p = relação entre frequências , f_c/f_m ;
M = índice de modulação ;
f_c = frequência da onda portadora , em hertz ;
f_m = frequência da onda modulante , em hertz ;
T = periodo da frequência da onda portadora $(1/f_c)$, em segundos ;
W_n = frequência angular de modulação $(2\pi/pT)$, em radianos por segundos .

Para o contador gerar o pulso com a largura t_n (em segundos) ele deve ser carregado com um valor numérico que é definido como:

$$N_n = f_r\, t_n. \qquad (2)$$

Se o valor de t_n é substituído de (1) em (2) obtem-se:

Se o valor de t_n é substituido de (1) em (2) obtem-se:

$$N_n = \frac{1}{f_m} valor_o + (-1)^{n+1} \frac{M}{f_m} valor_n \quad (3)$$

$$valor_o = \frac{f_r}{2p} \quad (4)$$

$$valor_n = \frac{f_r}{4p} \left\{ Sen\left[(n-1.5)\frac{\pi}{p}\right] + Sen\left[(n-0.5)\frac{\pi}{p}\right] \right\} \quad (5)$$

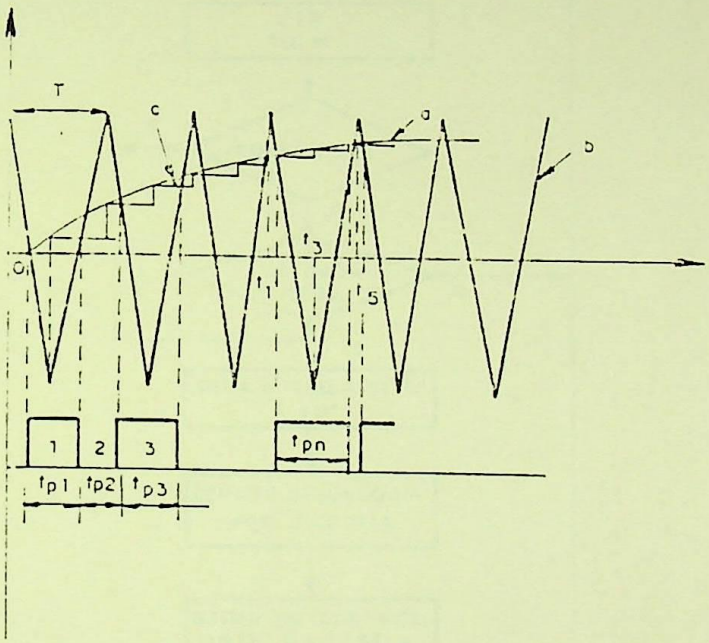


Fig. 3.2 PWM com amostragem regular : (a) sinal modulante (b) sinal da onda portadora (c) sinal amostrado

A largura do primeiro pulso na Fig. 3.2 é :

$$t_{p1} = \frac{T}{2} + \frac{M\ T}{2} Sen\ (\ W_n\ T/4\ ) \quad (6)$$

Deste modo :

$$N_{p1} = \frac{1}{f_r} valor_o + \frac{M}{f_r} valor_1 \quad (7)$$

Onde :

$$valor_1 = \frac{f_r\ Sen\ (\ \pi\ /\ 2p\ )}{2p} \quad (8)$$

Para um valor constante de p , os valores $valor_o$, $valor_1$,

$valor_2$, ...são calculados no inicio do programa .

As equações (3) e (7) mostram que a largura dos pulsos dependem do índice de modulação M e da frequência da onda modulante f_m , considerando um valor constante da relação p.Deste modo , o computador deverá receber os valores de f_m e M para gerar os valores correspondentes das largura dos pulsos .

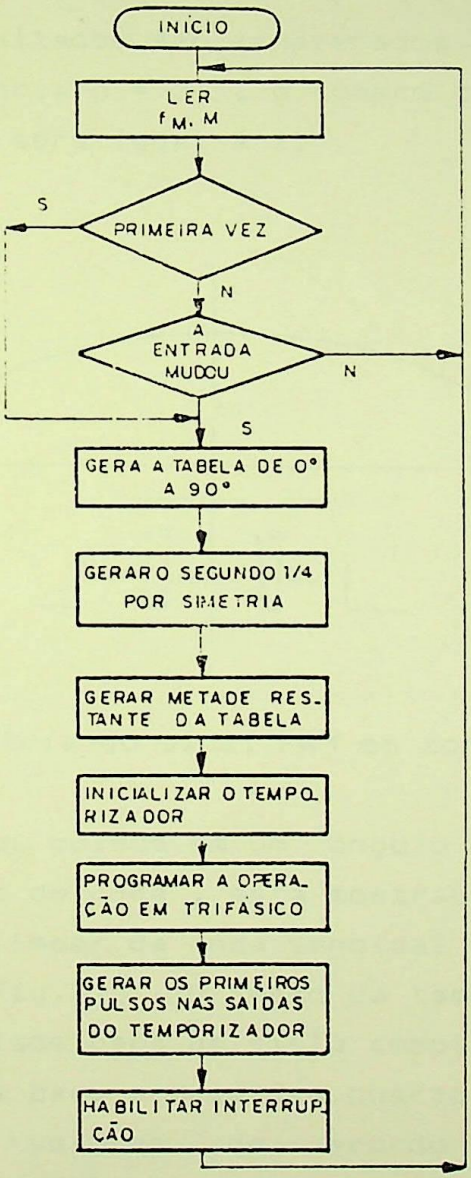


Fig. 3.3 Fluxograma do programa principal

O valor de f_m determina a velocidade do motor . O valor de M, que determina a tensão no estator , depende exclusivamente do metodo de controle usado .

O computador , após receber os valores de entrada f_m , M , e p , faz todos os cálculos necessários para formar uma tabela completa contendo os valores dos pulsos que serão gerados . Um Temporizador Programável 8253 da INTEL é então usado para gerar os sinais PWM em tempo real . O fluxograma para a rotina que gera os sinais está mostrado na Fig. 3.3 .

Esta rotina usa os valores de f_m, M e p para calcular os valores dos pulsos para ângulos entre 0° até 90° (um quarto de um ciclo) .As fórmulas (3) e (7) são usadas para esta finalidade.Os resultados são armazenados em uma tabela .Para uma relação de frequência p = 21 , o número de pulsos para o primeiro um quarto de onda será igual a 11 .

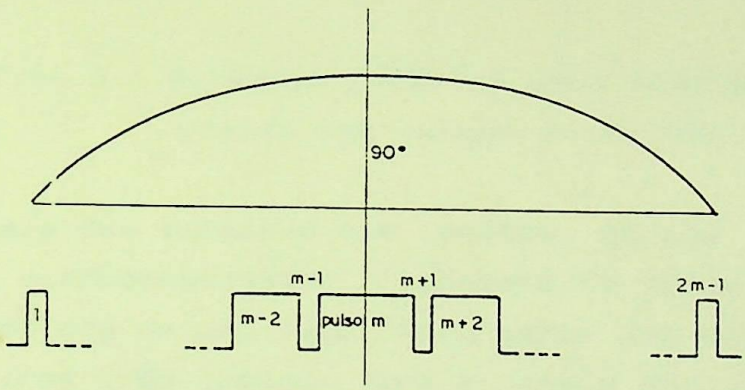


Fig. 3.4 Simetria do sinal PWM em torno do ângulo de 90°

A simetria dos pulsos em um ângulo de 90° , que é uma simetria de quarto de onda , está mostrada na Fig. 3.4 .Do mesmo modo , a simetria ímpar da onda senoidal , em um ângulo de 180° , está mostrado na Fig.3.5 . O resto da tabela , que contém todos os valores dos pulsos para um ciclo completo , é gerado usando os valores computados para o primeiro quarto de onda, simplesmente copiando estes valores de acordo com as simetrias específicas.Isto economiza um bom tempo do microcomputador .

Cada temporizador do 8253 estará ligado a uma fase .Estes contadores serão carregados com valores existentes na tabela .Os valores são obtidos na tabela por meio de três ponteiros .Os sinais gerados pelos temporizadores são sinais em tempo real.Após

a contagem de cada temporizador , a saída muda seu nível , e a rotina de serviço de interrupção transfere o próximo valor do pulso para o latch do contador .

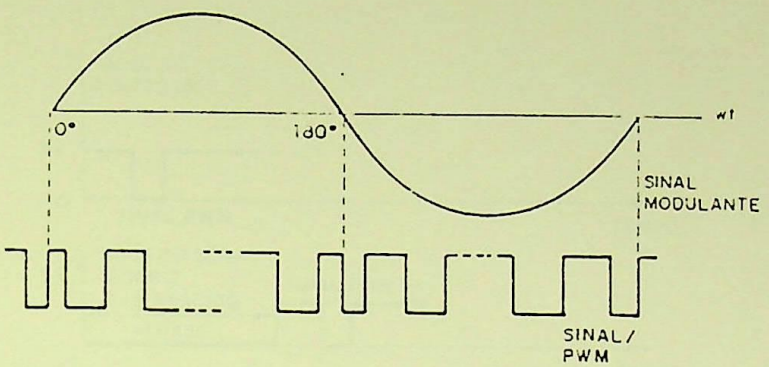


Fig. 3.5 Simetria impar da onda senoidal e seu efeito nos pulsos entre 180° e 360°

A tabela das larguras dos pulsos gerada e armazenada na memória do microprocessador , consiste de todos os valores para o período completo de uma fase .Para gerar uma saída em trifásico , três ponteiros irão apontar para a tabela com uma defasagem de um terço de periodo cada . Cada ponteiro é usado para servir a um temporizador do 8253 .

Para realizar a defasagem de 120° entre as três saídas do 8253 após a inicialização deste componente , a primeira fase começa recebendo o primeiro valor da tabela PWM .A segunda fase é forçada a fazer uma defasagem de um terço de período do sinal modulante e após a defasagem o sinal PWM é transferido para o temporizador .A terceira fase é forçada a fazer uma defasagem de dois terços de período ; após esta defasagem o sinal PWM é gerado, como mostrado na Fig. 3.6 .

Este método de criar a defasagem é um método muito preciso.A resolução chega a 0.5 μs .Esta resolução é maior se a frequência do clock do temporizador também for maior .

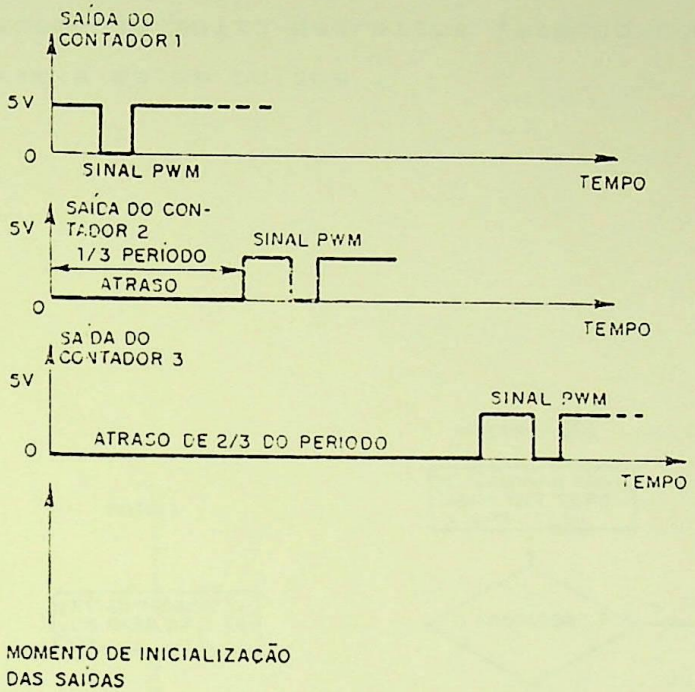


Fig. 3.6 Método de geração do sistema trifásico

A parte do programa que continuamente irá carregar cada contador será a Rotina de Serviço de Interrupção .O fluxograma desta rotina está mostrado na Fig.3.7 .

Após a inicialização dos contadores o programa ficará apenas carregando estes contadores . Grande parte da capacidade do microprocessador permanecerá disponível assim como grande parte de sua memória .Isto indica que outras funções tais como proteção do circuíto e controle de processo poderão ser instaladas .

Na opção "Onda Senoidal" o usuário irá escolher em seguida o índice de modulação M e a relação entre frequências p .O índice de modulação deverá variar entre 0 e 1 ,inclusive .Já a relação p

deverá assumir apenas três valores : 9, 15,e 21 .Estes valores são os valores que possuem um aplicação prática aceitável .As limitações nos valores de p não são puramente teóricas .Há um grande ingrediente prático nesta limitação .Ao diminuir o valor de p ,estaremos criando muitos harmônicos de baixa frequência , o que é ruim para o lado AC. O ideal é que p fosse o maior possível. Isto todavia não é praticamente possível pois geraria pulso de chaveamento muito estreitos fazendo com que o inversor não respondesse a estes pulsos .

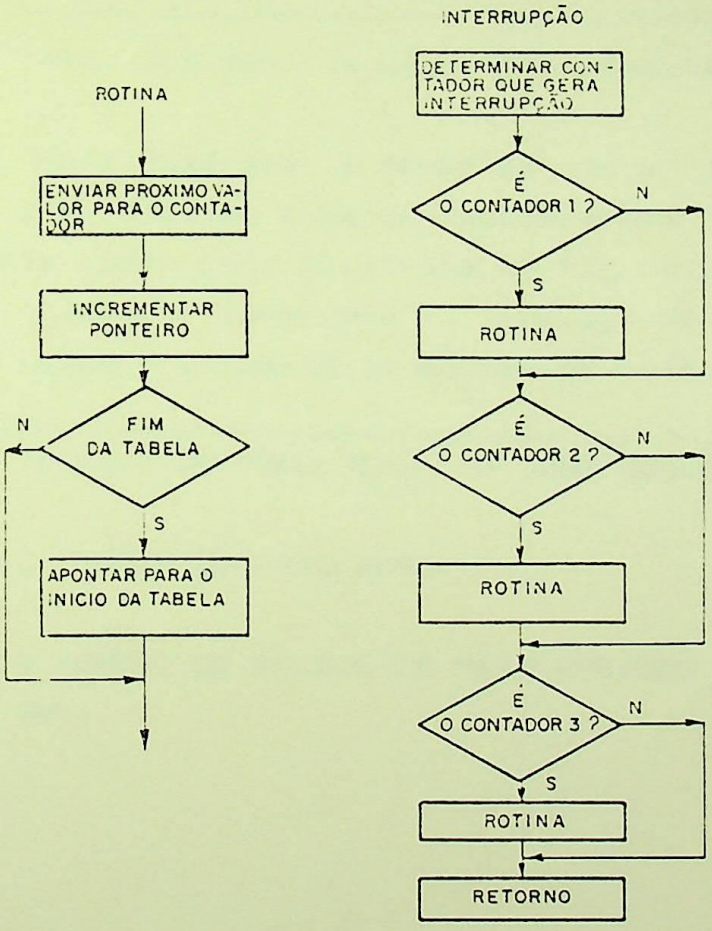


Fig. 3.7 Fluxograma da rotina de serviço de interrupção

```
                TECNICA DE MODULACAO SENOIDAL

ENTRE COM O INDICE DE MODULACAO
1

ENTRE COM O NUMERO DE PULSOS EM MEIO PERIODO
AS OPCOES SAO :
-9 PULSOS
-15 PULSOS
-21 PULSOS
9
ENTRE COM A FREQUENCIA DE SAIDA EM HERTZ:
60
```

Fig. 3.8 Menu da modulação senoidal

Na opção "Onda Quadrada" a forma de onda modulante é uma onda quadrada e a forma de onda portadora é uma onda triangular modificada .Esta opção está ilustrada na Fig. 3.10 , Fig. 3.11 e Fig. 3.12. O usuário fornecerá o índice de modulação , a frequência de saída e o número de pulsos em meio período .

```
            TECNICA DE MODULACAO COM ONDA QUADRADA

ENTRE COM O INDICE DE MODULACAO
0.6
ENTRE COM O NUMERO DE PULSOS EM MEIO PERIODO
AS OPCOES SAO:
-3 PULSOS
-5 PULSOS
-7 PULSOS
 7
ENTRE COM A FREQUENCIA DE SAIDA EM HERTZ:
60
```

Fig. 3.9 Menu da modulação com onda quadrada

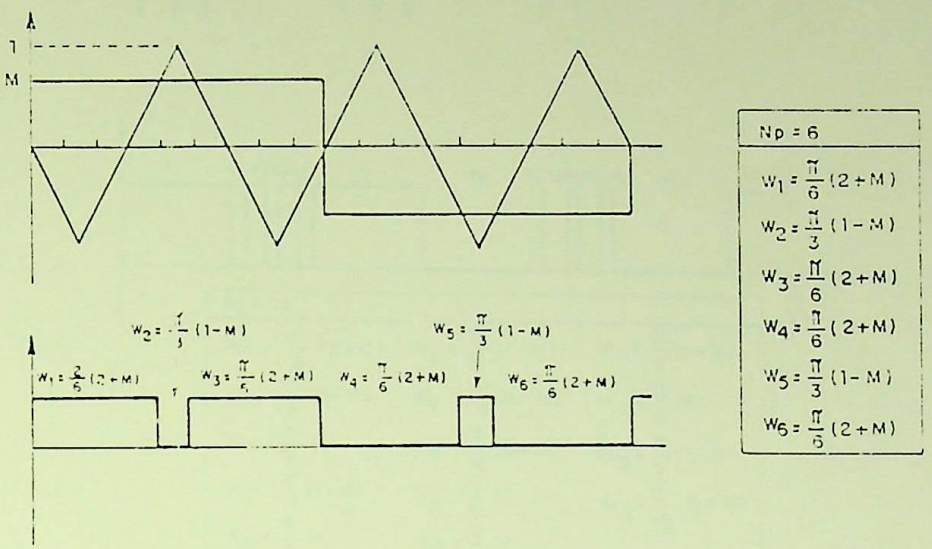


Fig. 3.10 Modulação com onda quadrada tendo 3 pulsos em meio período

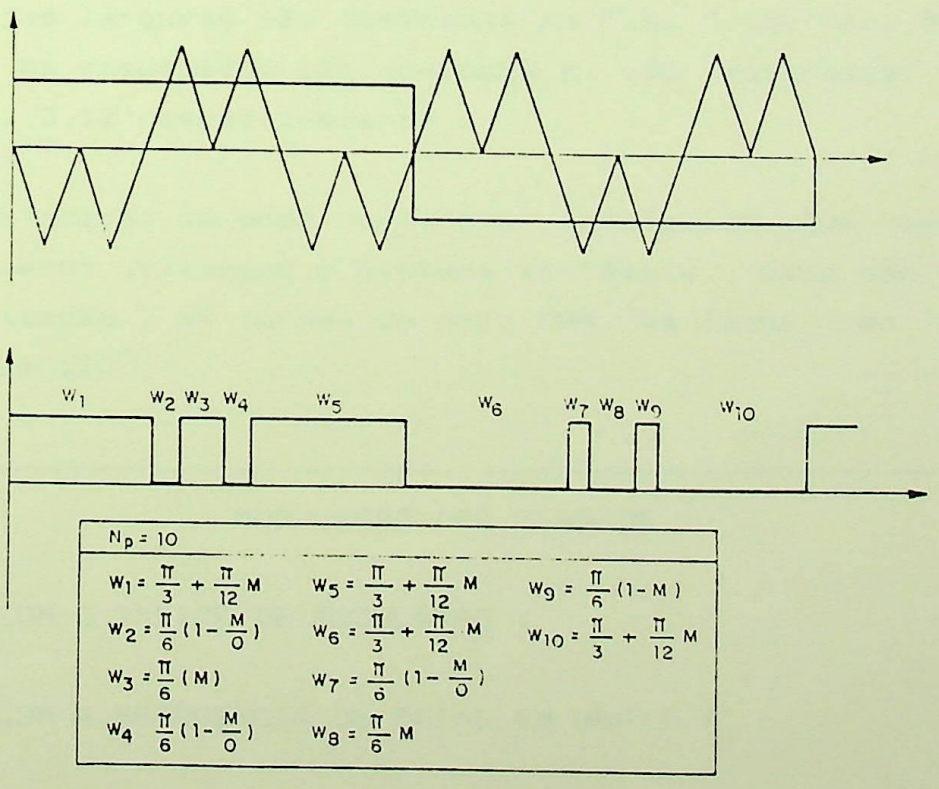
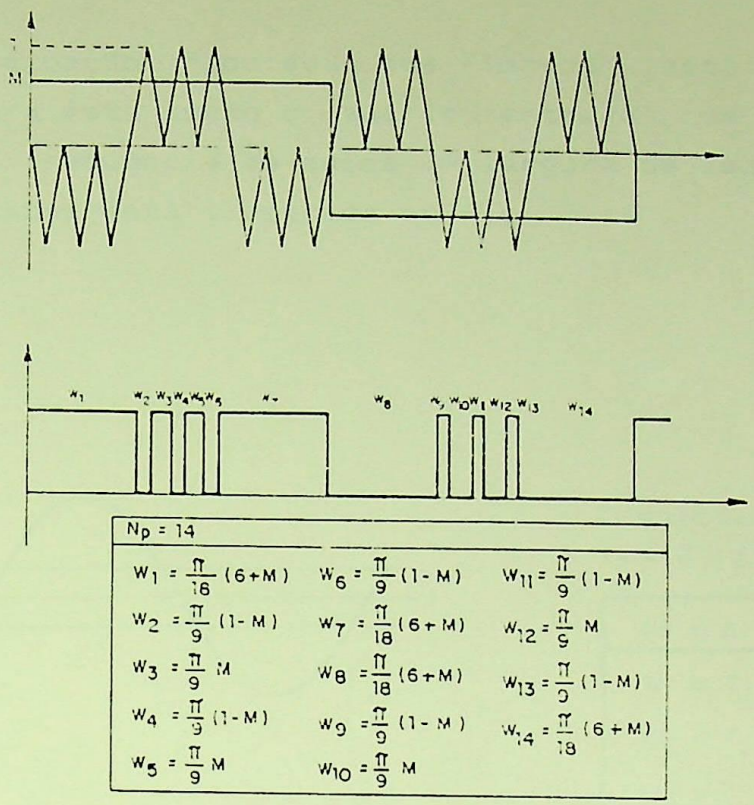


Fig. 3.11 Modulação com onda quadrada tendo 5 pulsos em meio período



$N_p = 14$

$W_1 = \frac{\pi}{18}(6+M)$	$W_6 = \frac{\pi}{9}(1-M)$	$W_{11} = \frac{\pi}{9}(1-M)$
$W_2 = \frac{\pi}{9}(1-M)$	$W_7 = \frac{\pi}{18}(6+M)$	$W_{12} = \frac{\pi}{9}M$
$W_3 = \frac{\pi}{9}M$	$W_8 = \frac{\pi}{18}(6+M)$	$W_{13} = \frac{\pi}{9}(1-M)$
$W_4 = \frac{\pi}{9}(1-M)$	$W_9 = \frac{\pi}{9}(1-M)$	$W_{14} = \frac{\pi}{18}(6+M)$
$W_5 = \frac{\pi}{9}M$	$W_{10} = \frac{\pi}{9}M$	

Fig. 3.12 Modulação com onda quadrada tendo 7 pulsos em meio período

Para 3 pulsos em meio período teremos 6 pulsos no período inteiro cujas larguras são mostradas na Fig. 3.10. Para 5 pulsos e 7 pulsos os resultados são analogos e são mostrados na Fig. 3.11 e Fig. 3.12 respectivamente .

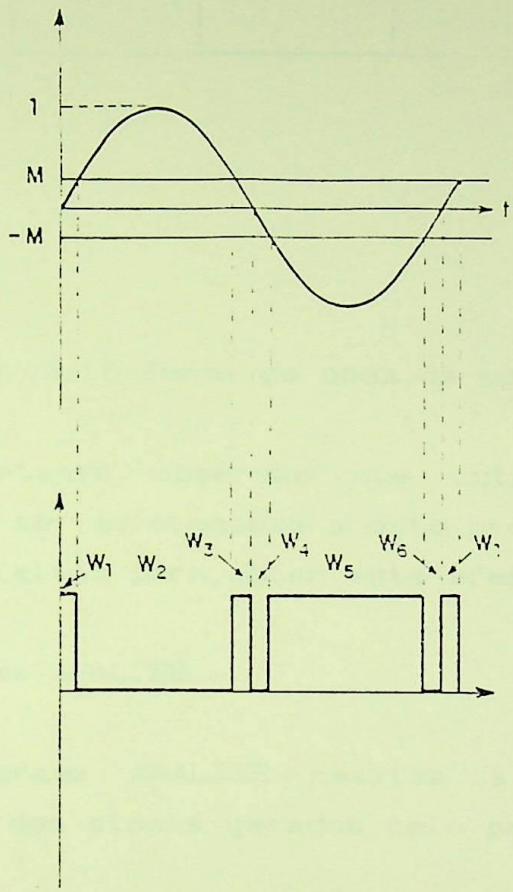
A modificação na onda portadora triangular foi necessária pois está sendo utilizado o sistema trifásico . Caso não houvesse essa modificação , as formas de onda PWM nas fases não estariam defasadas de 120°.

```
                MODULACAO DOS FLANCOS

ENTRE COM O INDICE DE MODULACAO :
0.2
ENTRE COM A FREQUENCIA DE SAIDA EM HERTZ :
60
```

Fig. 3.13 Menu da opção modulação dos flancos

O menu da opção "Modulação dos Flancos" está ilustrado na Fig. 3.13 .Para esta opção o usuário entrará com o índice de modulação e a frequência de saída .A largura de cada pulso para esta opção também está ilustrada na Fig. 3.14 .



$T_1 = \text{arc Sen}(M)$

$T_2 = \pi - 2T_1$

$N_D = 6$
$W_1 = T_1$
$W_2 = T_2$
$W_3 = W_1$
$W_4 = W_1$
$W_5 = W_2$
$W_6 = W_1$

Fig. 3.14 Modulação dos flancos

Na opção "Bloco Completo" o usuário deverá fornecer apenas a frequência de saída .Nesta opção é gerado um sistema trifásico com três ondas quadradas.Para esta opção o menu está ilustrado na Fig. 3.15 e as formas de onda na Fig. 3.16 .

```
                    BLOCO COMPLETO

ENTRE COM A FRQUENCIA DE SAIDA EM HERTZ
60
```

Fig. 3.15 Menu da opção bloco completo

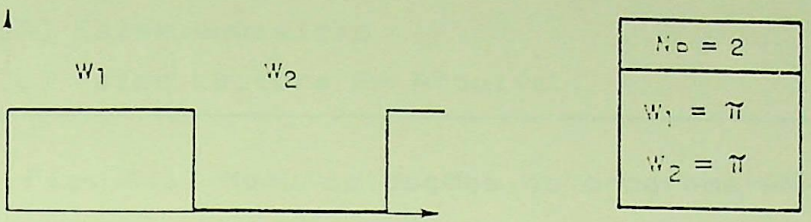


Fig. 3.16 Forma de onda da opção bloco completo

É importante observar que outras estratégias PWM podem facilmente ser adicionadas a este programa . Poucas modificações devem ser feitas para obter este efeito .

3.3 Programa ANALISE .

O programa ANALISE realiza a analise no tempo e na frequência dos sinais gerados pelo programa DISP .

Após fazer o disparo , o usuário poderá fazer em seguida a aquisição dos dados que estão sendo gerados em tempo real. O menu de opções do programa ANALISE está mostrado na Fig. 3.17. Na opção "Fazer Aquisição", o usuário irá processar a coleta dos 2048 pontos que serão usados para calcular o espectro de frequência do sinal PWM , assim como apresentar esta forma de onda no domínio do tempo. Na opção "Fazer Leitura Em Arquivo" , o usuário poderá analizar informações que foram gravadas previamente em um arquivo.

Feita a aquisição dos sinais , o usuário poderá gravar os dados em um arquivo como mostrado na Fig. 3.18 . Se a opção escolhida for a opção "Fazer Leitura Em Arquivo" , o usuário informará o diretório e o arquivo onde se encontra os dados ,

como mostrado na Fig. 3.19 .

```
                    PROGRAMA ANALISE

        [A] Fazer Aquisicao
        [L] Fazer Leitura Em Arquivo
```

Fig. 3.17 Menu de opções do programa ANALISE

```
Deseja Gravar Em Arquivo ? S/N
S
Diretorio Atual :C\PWM
Mudar Diretorio ? S/N
S
Nome Do Diretorio
C:\PWMDAT
Nome Do Arquivo :
TP100
```

Fig. 3.18 Gravação dos dados em arquivo

```
Diretorio Atual :C\PWM
Mudar Diretorio ? S/N
S
Nome Do Diretorio:
C:\PWMDAT
Nome Do Arquivo:
TP100
```

Fig. 3.19 Leitura de dados em arquivo

Em seguida, o usuário poderá escolher a apresentação que desejar :"Apresenta Domínio do Tempo" ou "Apresenta Domínio da Frequência".Caso não deseje nenhuma destas opções , ele poderá fazer uma saída para o DOS ,na opção "Saída Para o DOS", ou estudar um novo caso , na opção "Estudar Um Novo Caso".Estas opções estão mostradas na Fig. 3.20 .

```
[T] Apresenta Dominio Do Tempo
[F] Apresenta Dominio Da Frequencia
[N] Estudar Novo Caso
[S] Saida Para O DOS
```

Fig. 3.20 Menu de apresentação dos resultados

Na opção "Apresenta Domínio de Tempo" o usuário irá ver a forma de onda das tensões Vab , Vbc , Vca .Na opção " Apresenta Domínio da Frequência ", será apresentado o espectro de frequência destas formas de onda .

Juntamente com o programa ANALISE , funciona uma placa que fará a aquisição de 2048 amostras colhidas dos sinais de modulação.É a partir destas amostras que será possível observar o sinal no domínio do tempo e da frequência .

A Fig. 3.21 e Fig. 3.22 mostram como o programa ANALISE representará estes dados .

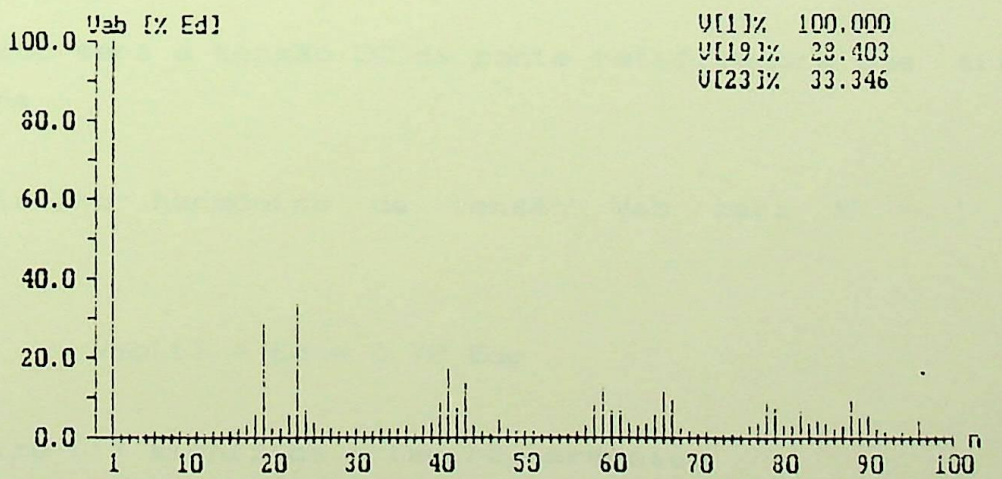


Fig. 3.21 Espectro de frequência

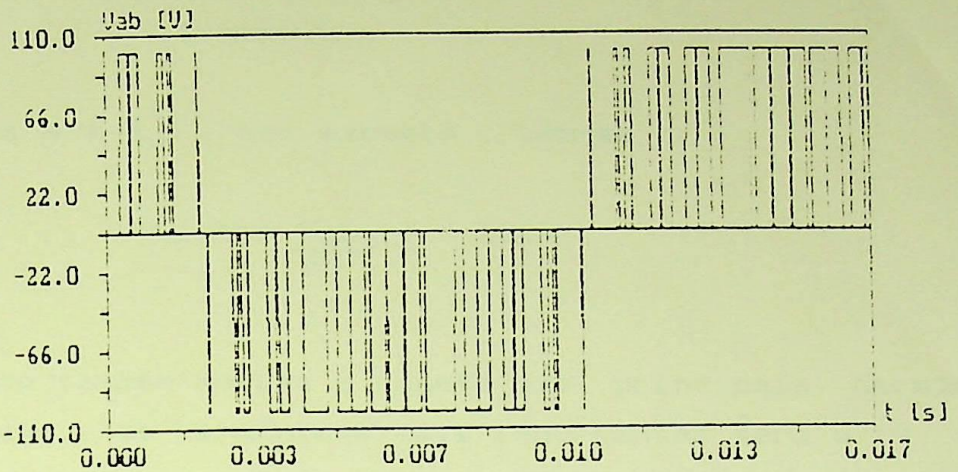


Fig. 3.22 Forma de onda no domínio do tempo

No gráfico do espectro de frequência apresentado pelo programa ANALISE ,são necessários alguns comentários. Vemos no eixo das ordenadas uma escala variando de 0 a 100 . Esta escala foi chamada de [% Ed]. Qual o significado de [% Ed]? Ed é o máximo valor da componente fundamental da tensão de saída . Para este caso , este valor ocorrerá quando M = 1 , e será igual a :

$$Ed = 0.78\ Edc$$

Onde Edc será a tensão DC da ponte retificadora que alimenta a inversora .

O primeiro harmônico da tensão Vab para M = 1 será portanto:

$$Vab[1] = Ed = 0.78\ Edc$$

O índice [1] significa primeiro harmônico .

Para M = 1 obtem-se então :

$$V[1] = Ed$$

$$V[1]\% = \frac{Vab[1]}{Ed} \cdot 100\% = \frac{Vab[1]}{Vab[1]} \cdot 100\% = 100\%$$

$$V[1]\%=100\%$$

Para M = 0.5 , por exemplo , tem-se :

$$V[1]\% = \frac{Vab[1]}{Ed} \cdot 100\%$$

O gráfico também mostra a tensão dos principais harmônicos .No nosso caso , os harmônicos mais importantes será o 19º harmônico e o 23º harmônico . Para estes harmônicos tem-se ;

$$V[19]\% = \frac{Vab[19]}{Ed} \cdot 100\%$$

$$V[23]\% = \frac{Vab[23]}{Ed} \cdot 100\%$$

No eixo das abcissas existe uma graduação que vai de 0 a 100, e a etiqueta " n ".Esta etiqueta significa a ordem do harmônico considerando uma fundamental , que no caso foi de 60 Hz.

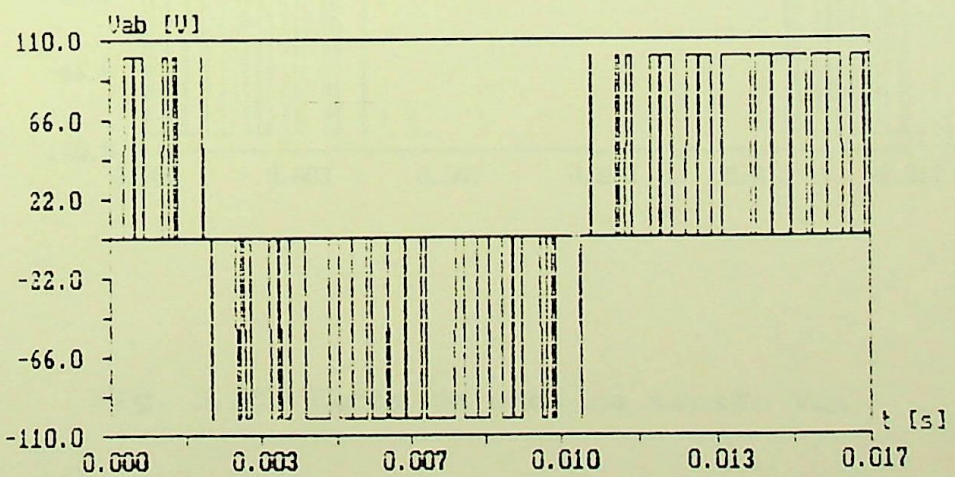


Fig 3.25 Forma de onda da tensão Vab

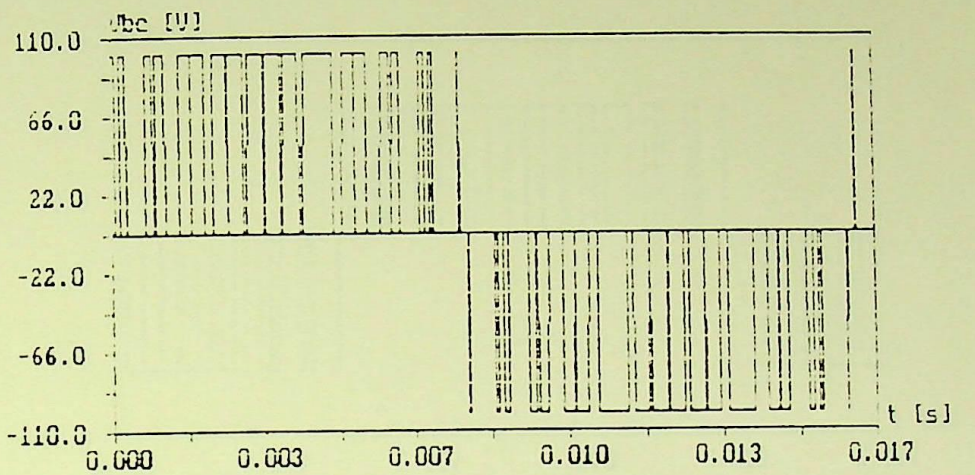


Fig. 3.24 Forma de onda da tensão Vbc

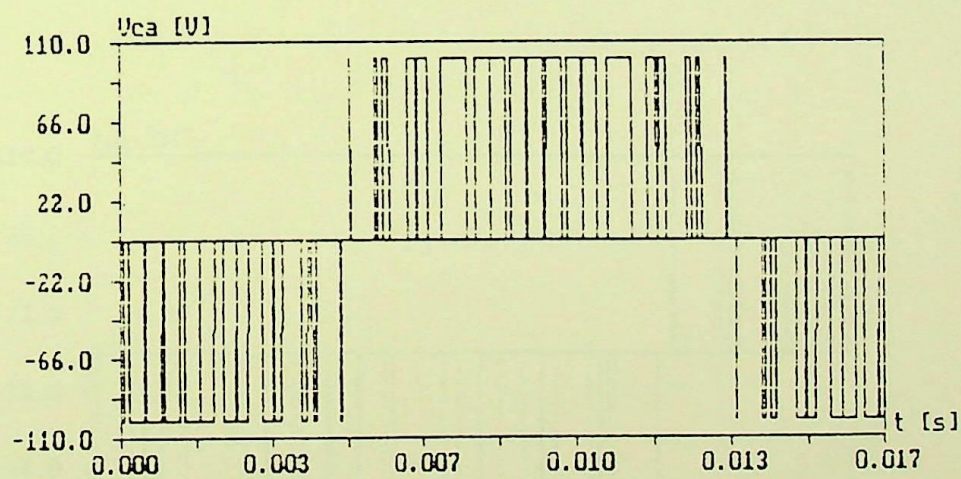


Fig. 3.25 Forma de onda da tensão Vca

No domínio do tempo o programa ANALISE apresentará as tensões Vab , Vbc , e Vca .Isto está mostrados nas Fig. 3.23 , Fig. 3.24 e Fig. 3.25 . O usuário poderá fazer ainda uma

alteração no eixo da tensão (Fig. 3.26) e no eixo do tempo (Fig. 3.27) . No eixo do tempo será possível desta forma verificar com mais detalhes certos trechos da forma de onda .

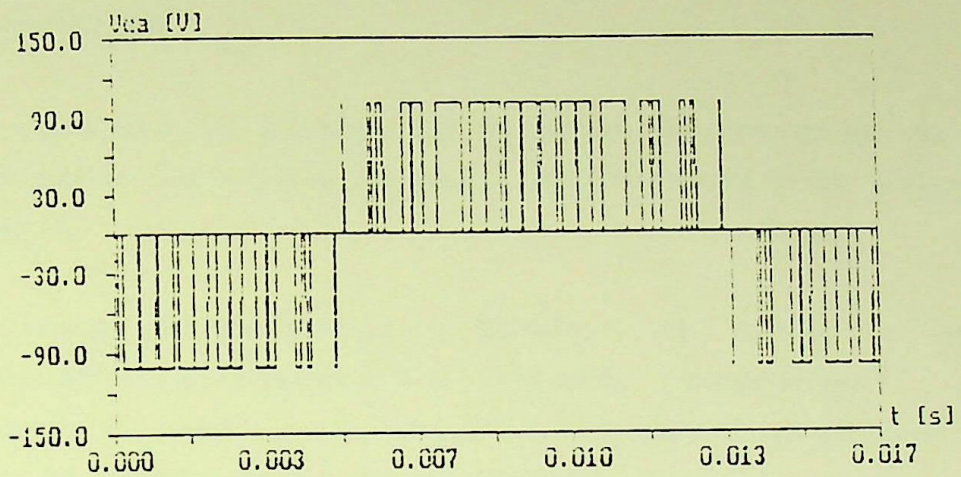


Fig. 3.26 Alteração na escala de tensão

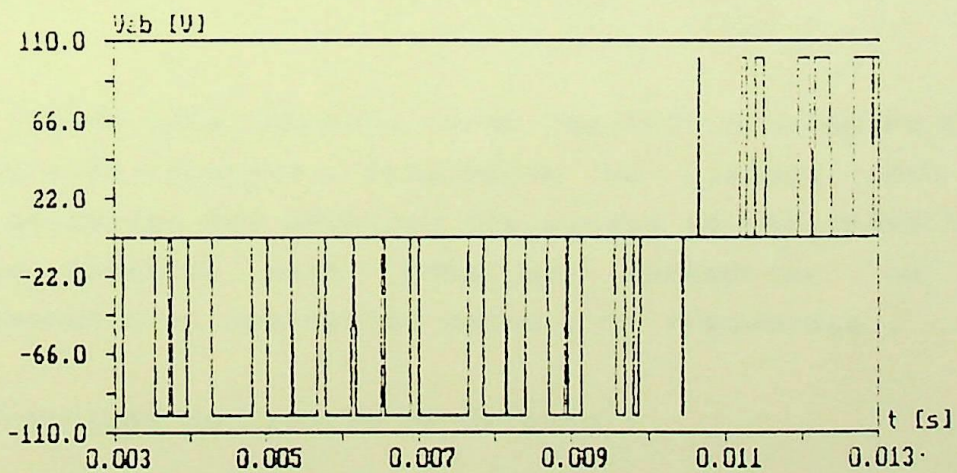


Fig. 3.27 Alteração na escala de tempo

CAPÍTULO IV
HARDWARE

4.1 INTRODUÇÃO:

O hardware do sistema de disparo foi projetado de modo que pudesse gerar os pulsos de gatilhamento dentro da maior precisão possível .

Formado por um circuito gerador de clock , um circuito decodificador de endereço e o circuito temporizador , o sistema de disparo é ligado a um computador pessoal compatível com IBM PC através de um dos slots disponíveis .

Através da combinação proposta foi possível gerar pulsos com uma precisão de 0.5 μs , e ondas de modulação de até 200 Hz para um microprocesador operando em 4.77 MHz . Foi procurado aproveitar as características de baixo custo e grande capacidade de programação dos temporizadores para que o circuito pudesse acomodar várias técnicas de modulação PWM .

Também será abordado neste capítulo o hardware do Analisador Lógico e de Espectro . Esta parte do sistema será responsável pela aquisição das amostras dos sinais de gatilhamento. A partir destas amostras será possível apresentar os sinais de gatilhamento no domínio do tempo e da frequência .

4.2 Descrição Do Cicuito De Disparo :

O circuito de disparo está mostrado na Fig. 4.1 .Este circuito é dividido em vários setores que serão analisados na

seção seguinte .

Serão analisados nesta seção as partes componentes do Circuíto de Disparo .

A) Circuíto Decodificador De Endereço :

Este circuíto é formado pelos integrados CI-1 , CI-2 , CI-3 CI-4 e pelas chaves CH1 A CH6 . Sua finalidade é estabelecer os endereços de entrada e saída (I/O) dos circuítos envolvidos no sistema de disparo .

Com isto o usuário poderá programar em quais endereços estarão localizados os periféricos da placa .

B) Circuíto De Clock :

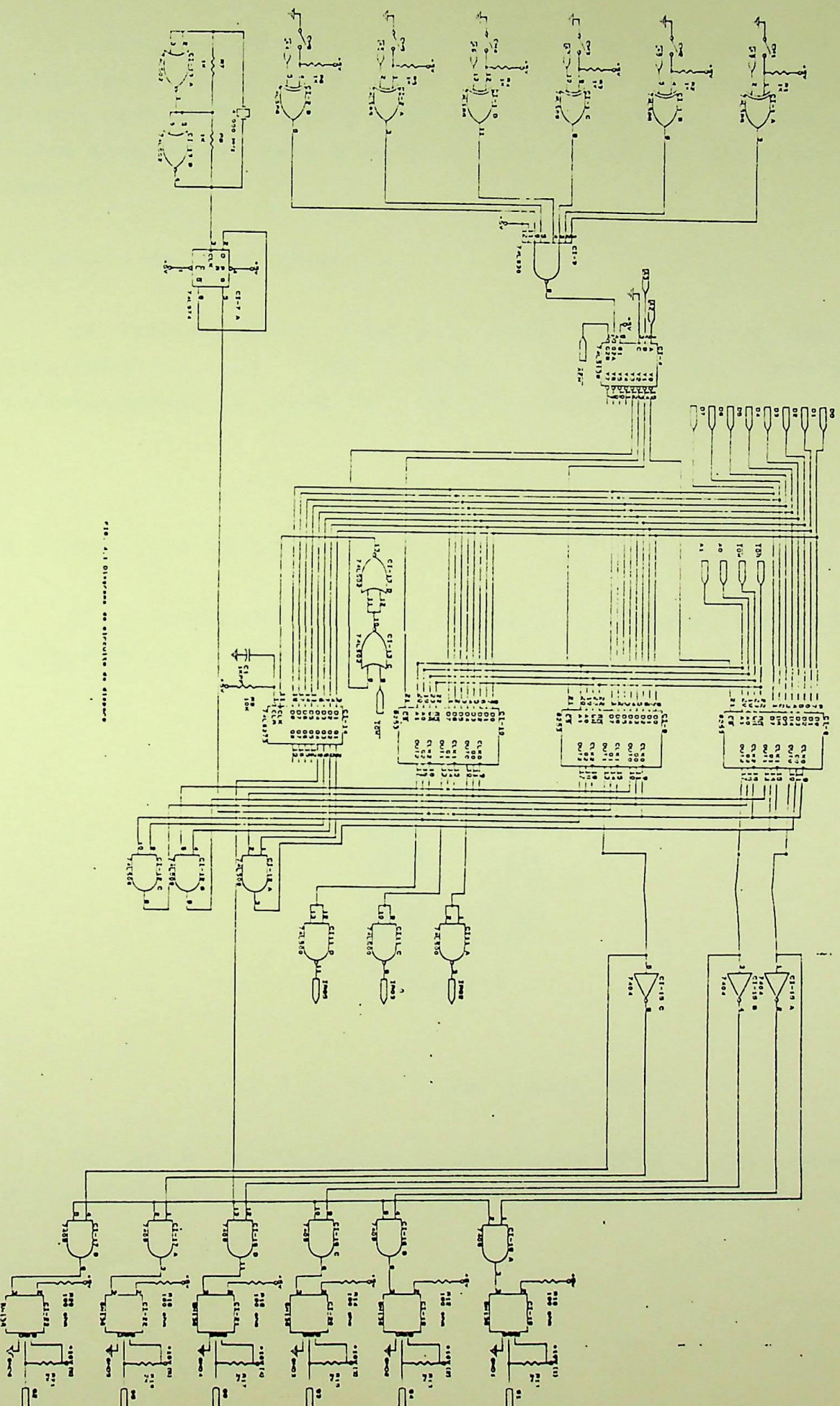
Formado pelo CI-13 e CI-7 , este circuíto tem como finalidade gerar um clock de 2 MHz que será utilizado como base de tempo para os temporizadores .

C) Circuíto Temporizador :

Este circuíto é programado pelo microprocessador e será responsável pela geração dos sinais de gatilhamento . Ele é formado pelos integrados CI-8, CI-9 e CI-10 .

D) Circuíto De Sincronísmo :

Formado pelo CI-12, CI-13 e CI-14 , este circuíto gera o sinal de gatilhamento para estabelecer o sincronismo de 120° entre as formas de onda nas três fases .



E) Circuito Gerador De Interrupção :

O circuito gerador de interrupção é formado pelo CI-11 .A saída deste circuito irá gerar os sinais para as linhas de interrupção IRQ2 , IRQ3 e IRQ5 do microcomputador .Estas linhas foram programadas para atuarem quando houver uma transição de 1 para 0 .

F) Sinais De Gatilhamneto :

As linhas FASE A , FASE B e FASE C , são os sinais de gatilhamento que irão acionar o circuito inversor , correspondendo às fases Va , Vb e Vc respectivamente .

4.3 Descrição do Funcionamento:

O temporizador 8253 é formado por três temporizadores independentes , como mostrado na Fig. 4.2 .

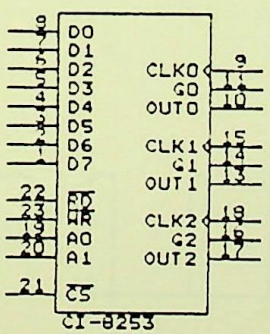


Fig. 4.2 Temporizador 8253

Devido às limitações do temporizador 8253 , foi necessário utilizar alguns artifícios de hardware para realizar a geração

das ondas PWM .

Um sinal PWM basicamente é gerado por meio de uma sequência de pulsos ligados e desligados , como mostrado na Fig. 4.3.

A formação do pulso será feita gerando várias sequências ligado-desligado . Por exemplo , na Fig. 4.3 será gerado em primeiro lugar a sequência $W_1 + W_2$, em seguida $W_3 + W_4$ e assim sucessivamente .

A implementação utiliza a seguinte técnica :

- O primeiro contador é carregado com a sequência $W_1 + W_2$ no seu latch ;
- O contador é disparado pelo CI-14;
- Logo em seguida é colocado no latch do contador a sequência $W_3 + W_4$;
- Após o término da contagem , o contador se alto dispara ;
- Um novo valor é colocado no latch ;

A forma de onda gerada por este primeiro contador está mostrada na Fig. 4.4 .

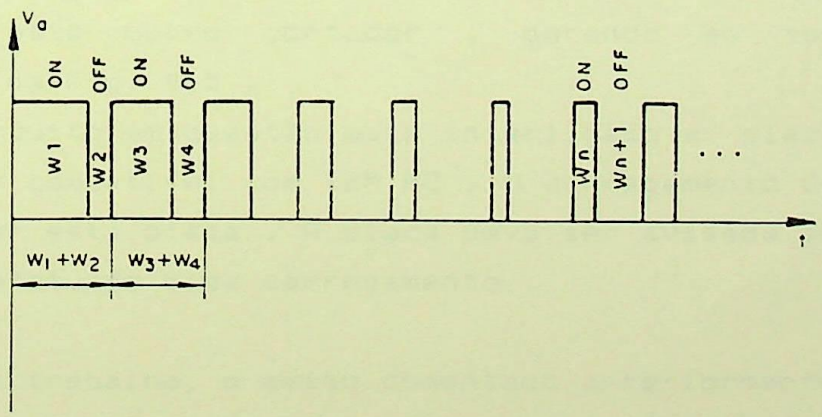


Fig. 4.3 Um signal PWM típico e sua sequência de pulsos ON e OFF

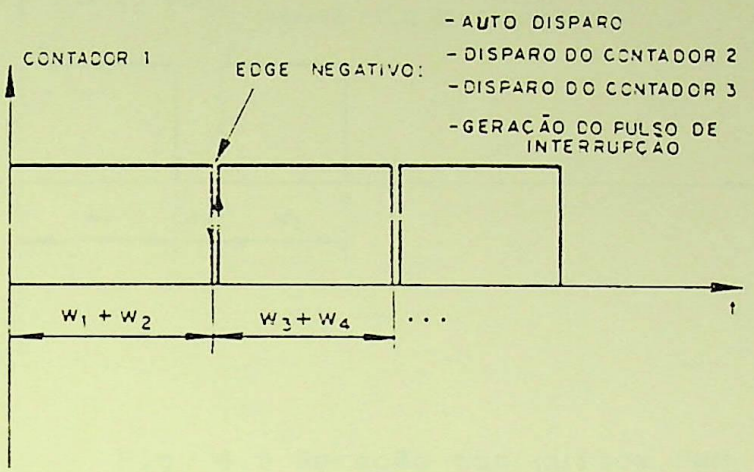


Fig. 4.4 Geração de pulsos $W_{n\ ON} + W_{n+1\ OFF}$

O segundo contador é carregado inicialmente com o valor $W_{n\ ON}$, de modo a permanecer ligado durante este período . Em seguida , o contador é disparado e procede à contagem . O latch deste contador é imediatamente carregado com o valor W_{n+1ON} . O contador visto anteriormente além de disparar a si mesmo , irá disparar este outro contador , gerando as formas de onda mostradas na Fig. 4.5 .

O circuito em questão está interligado a placa mãe de um computador compatível com IBM PC . O carregamento dos contadores é feito por esta placa . A placa deve ser avisada em que instante deve ser efetuado cada carregamento .

Neste trabalho, o aviso comentado anteriormente é realizado por intermédio das linhas de interrupção do controlador de interrupção 8259 do PC. Esta linha deve permanecer no nível lógico "1" por alguns micro-segundos para que o controlador

possa reconhecer a interrupção .Um terceiro contador será utilizado para gerar o sinal de interrupção .

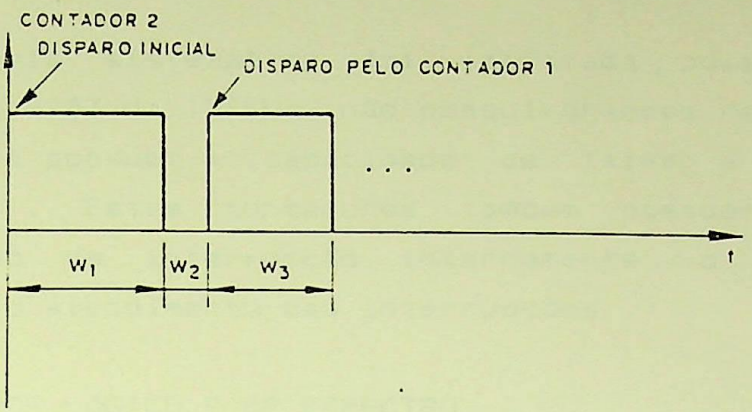


Fig. 4.5 Geração dos pulsos PWM

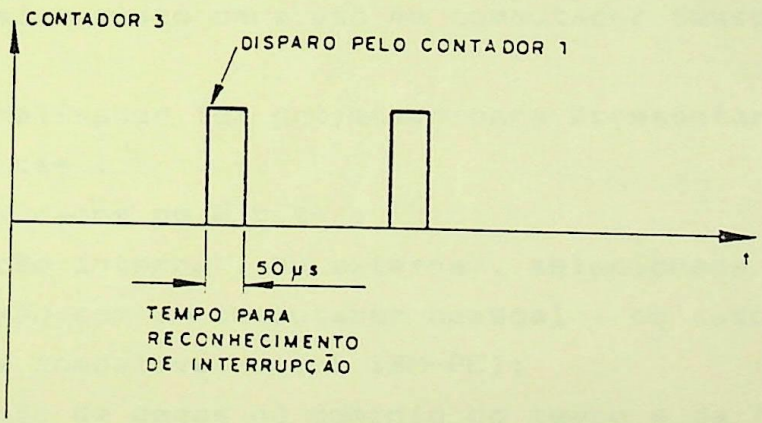


Fig. 4.6 Geração do sinal de interrupção

Este terceiro contador será carregado com 50 μs , que nas condições em questão é o tempo necessário para que o microprocessador reconheça a interrupção . Este contador será disparado pelo contador encarregado de gerar o tempo $W_{n\ ON} + W_{n+1}$

OFF , quando este chegar no final da contagem .

A Fig. 4.6 ilustra de modo simplificado o processo utilizado no terceiro contador.

Toda esta sistemática foi utilizada pois o contador empregado, o 8253 da INTEL , não possui grandes recursos . Alguns contadores já possuem a capacidade de fazer a modulação PWM diretamente . Estes contadores também possuem circuítos de prioritização de interrupção internamente o que facilita enormemente o atendimento das interrupções .

4.4 ANALISADOR LÓGICO E DE ESPECTRO

Como já foi mencionado , o conjunto aqui apresentado e formado por um circuíto de modulação e um analisador lógico e de espectro . O analisador aqui apresentado foi baseado em [9] , cujo objetivo principal foi o desenvolvimento de um analisador lógico de baixo custo para uso em computador pessoal .

Este analisador foi projetado para apresentar as seguintes características :

- Palavra de dados de 8 bits ;
- Temporização interna , ou externa , selecionada pelo usuário ;
- Interconexão com um computador pessoal (no caso foi utilizado um sistema compatível com o IBM-PC);
- Apresentação de dados no domínio do tempo e da frequência ;
- Conexão ao circuíto sob teste com ponta de prova compatível com lógica TTL ;
- Memória de aquisição com capacidade de 2048 palavras e de 8 bits ;

- Quatro modos de disparo , selecionados pelo usuário , ou seja :

a) Modo pós-disparo:após a palavra de disparo será adquirido até

2048 informações .

b) Modo pré-disparo : adquire dados até encontrar a palavra de disparo .

c) Modo Intermeriário : vem adquirindo amostras e , apos a palavra de disparo , continua a aquisição de mais uma quantidade de amostras definidas pelo usuário .

d) Modo pós e pré-disparo : adquire dados desde a palavra inicial até a palavra final .

Para a implementação dessas características , o circuito foi dividido nas seguintes funções :

a- Memória de aquisição ;
b- Seletor de endereçamento da memória;
c- Contador de aquisição ;
d- Contador programável de endereçamento ;
e- Buffer de dados ;
f- Comparador de sincronismo ;
g- Comparador de palavra final ;
h- Contador de números de eventos adquiridos ;
i- Comparador de endereço final ;
j- Contador programável de repetições da palavra de sincronismo ;
k- Gerador de pulso de aquisição ;
l- Seletor de pulso de aquisição ;
m- Habilitador de aquisição ;
n- Indicador de fim de aquisição ;
o- Programador de modo de operação ;
p- Circuito de interface para o microcomputador ;

O diagrama em blocos desse circuito está apresentado na Fig.4.7.

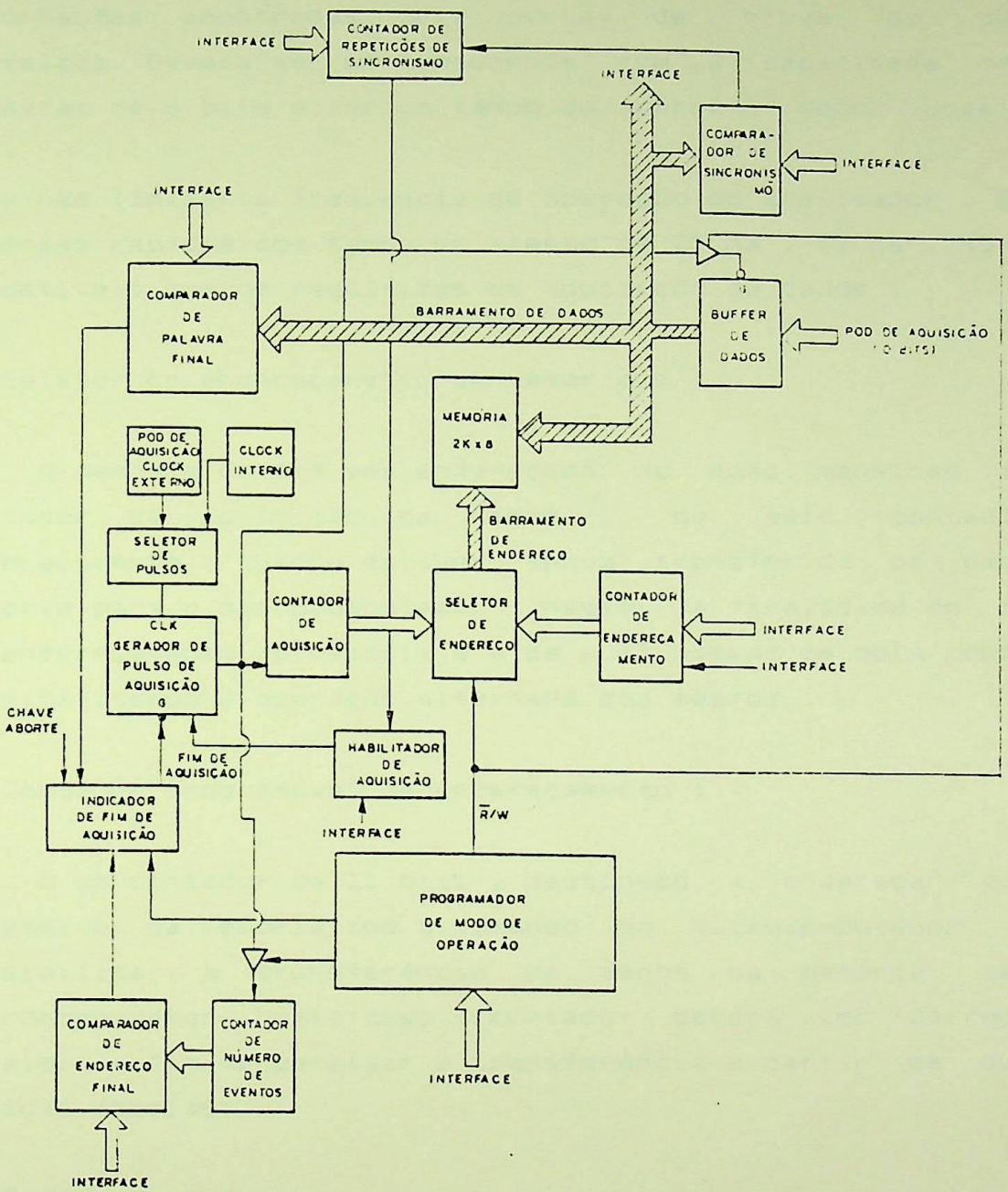


Fig. 4.7 Diagrama em bloco do circuíto desenvolvido.

4.4.1 Princípios De Operações Funcionais Do Circuito

A) Memória de aquisição :

A memória de aquisição tem por finalidade adquirir as informações amostradas pela ponta de prova do pod de aquisição. Deverá ser um componente com a capacidade de 2048 palavras de 8 bits e ter um tempo de acesso o menor possivel ,

para não limitar a frequência de operação do analisador . Existem memórias rápidas com tempo de acesso de 25 ns , 35 ns , 45 ns , compatíveis com os requisitos de aquisição de dados .

B) Seletor de endereçamento de memória :

A memória deverá ser endereçada de duas maneiras : pelo contador de aquisição de dados , ou pelo contador de endereçamento , quando estiver sendo transferido os dados da memória para o microcomputador . Assim , a finalidade do seletor de endereçamento da memória é a de multiplexar os dois contadores possibilitando a operação alternada dos mesmos .

C) Contador programável de endereçamento :

É um contador de 11 bits , destinado a endereçar posições sucessivas da memória sob o comando do microcomputador , para possibilitar a transferência de dados da memória para o microcomputador .Nesse caso , contador deverá ter carregamento paralelo a fim de permitir a transferência a partir de qualquer posição desejada .

D) Buffer de dados :

Será um circuíto habilitador de entrada das informções do pod de dados , quando estiver sendo realizados as aquisições dos

mesmos . Estará sendo habilitado pelo programador de modo de operação e pelo gerador de pulso de aquisição .

E) Comparador de sincronismo :

É um circuíto que receberá as informções do pod de dados e irá compará-las com o valor da palavra final de disparo , fornecida pelo circuíto de interface , possibilitando , assim , a operação do analisador nos modos pós-disparo (modo 1) e pós e pré-disparo (modo 4) .A igualdade desses dois valores informará o habilitador de interrupções .

F) Comparador de palavra final :

É um circuíto que receberá as informações do pod de dados e irá compará-las com o valor da palavra final de aquisição , fornecida pelo circuíto de interface , possibilitando a operação do analisador nos modos pós e pré-disparo (modo 2),intermediário (modo 3) e modo pós e pré-disparo (modo 4) .A igualdade desses dois valores informará o indicador de fim de aquisição , encerrando a mesma .

G) Contador de endereçamento :

É um contador de 11 bits , destinado a endereçamento sucessivo de memória , durante a aquisição dos dados .

H) Contador de número de eventos adquiridos :

É um contador de 11 bits , que informa para o comparador de endereço final o número de aquisições realizadas pelo analisador. Deverá operar no modo 1 e no modo 3 .

I) Contador programável de repitições da palavra de sincronismo :

Este circuito tem por finalidade contar quantas vezes ocorreu a palavra inicial de disparo . Isto possibilita fazer com que o analisador lógico possa ignorar a aquisição de dados em loops de programas. O número de repetições possíveis da palavra inicial estará compreendido entre 0 e 65535 vezes .

J) Comparador de endereço final :

É um circuito que receberá as informações do contador de número de eventos adquiridos e irá compará-las com o valor de número de aquisições , fornecidos pelo circuito de interface , realizando o número de aquisições desejadas pelo usuário . Encerrando , portanto , as aquisições quando ocorrer essa igualdade .

K) Gerador de pulsos de aquisição :

É um circuito destinado a habilitar a memória por um tempo suficiente para armazenagem de um dado verdadeiro no barramento de dados . O pulso de aquisição estará em sincronismo com o temporizador de dados válidos .

L) Seletor de pulsos :

É um circuito destinado a selecionar operações de temporização interna , ou temporização externa , bem como efetuar o sincronismo através da descida (edge negativo) , ou subida (edge positivo) , desse pulso de temporização .

M) Habilitador de aquisição :

É um circuito que recebe um comando do circuito de interface do microcomputador, ativando a operação do analisador , após o modo de operação do mesmo ter sido definido .

N) Indicador de fim de aquisição :

É um circuíto destinado a interromper a aquisição dos dados.Isso deverá ocorrer sob três condições : através de uma chave de abordagem do programa , que possibilita ao usuário interromper a aquisição ; ou quando o comparador de endereço final atuar ; ou quando o comparador de palavra final atuar .

O) Programador de modo de operação :

É um circuíto destinado a dizer quais partes do analizador deverão atuar nos diversos modos de operação .Receberá uma informação de controle através do circuíto de interface do microcomputador e definirá as partes que operam nos diversos modos .

P) Circuíto de interface para o microcomputador :

O analisador lógico opera independentemente de um microcomputador . O fato de se usar um microcomputador , ligado ao mesmo , se deve à utilização dos recursos de software para uma apresentação mais apreciavel das informações adquiridas pelo analisador . Assim , o circuíto de interface tem por objetivo conectar o analisador lógico a um microcomputador para possibilitar a programação do modo de operação do analisador e transferir os dados adquiridos para o microcomputador , a fim de que os mesmos possam ser apresentados lógica , ou graficamente , no video do microcomputador . Esse circuíto será melhor analisado no item seguinte .

4.4.2 Circuíto de interface com microcomputador :

O Diagrama em blocos desse circuíto está apresentado na Fig. 4.8. O circuíto de interface com o microcomputador tem por objetivo preparar a placa do analisador lógico para que a mesma

possa operar nos vários modos de aquisição .

Após preparado a operação em um modo de aquisição específico, um pulso de habilitação do analisador será emitido, liberando a operação do mesmo , até que ocorra um aviso de fim de aquisição .

Os circuítos de interface paralela são portas programadas como entrada , ou saída , destinados a manter valores necessários para as operações dos diversos modos de aquisição .

Esses valores tem os seguintes significados :

a) Palavra final : é um valor de 8 bits que será utilizado pelo comparador de palavra final para operação nos modos 2, 3, e 4 .

b) Palavra de amostragem : é um valor de 11 bits que será usado pelo comparador de endereço final nos modos 1 e 3 .

c) Número de sincronismo : é um valor de 16 bits que será carregado no contador de repetições de sincronismo , para definir quantas repetições devem ocorrer na palavra de disparo , nos modos 1 e 4 .

d) Palavra de sincronismo : é um valor de 8 bits que será usado pelo comparador de sincronismo para iniciar a aquisição de dados nos modos 1 e 4 .

e) Fim de aquisição : é um valor de 1 bit indicando o término da aquisição nos modos existentes .

f) Palavra de leitura : é um valor de 11 bits que será carregado no contador de endereçamento , para indicar o endereço inicial de leitura da memória ,quando estiver sendo feita a transferência dos dados da mesma para o computador .

g) Palavra de controle : é um valor de 8 bits destinado a selecionar quais partes do circuíto deverão operar em um modo específico , bem como selecionar temporização interna , ou externa , e transição positiva , ou negativa , do sinal de temporização .

h) Endereço final de memória : é um valor de entrada de 11 bits destinado a informar a última posição da memória que armazenou informação durante a aquisição dos dados .

i) Dado de saída do analisador : é um valor de saída de 8 bits correspondente ao conteúdo corrente da memória , que será usado para a transferência dos dados entre a memória e o microcomputador .

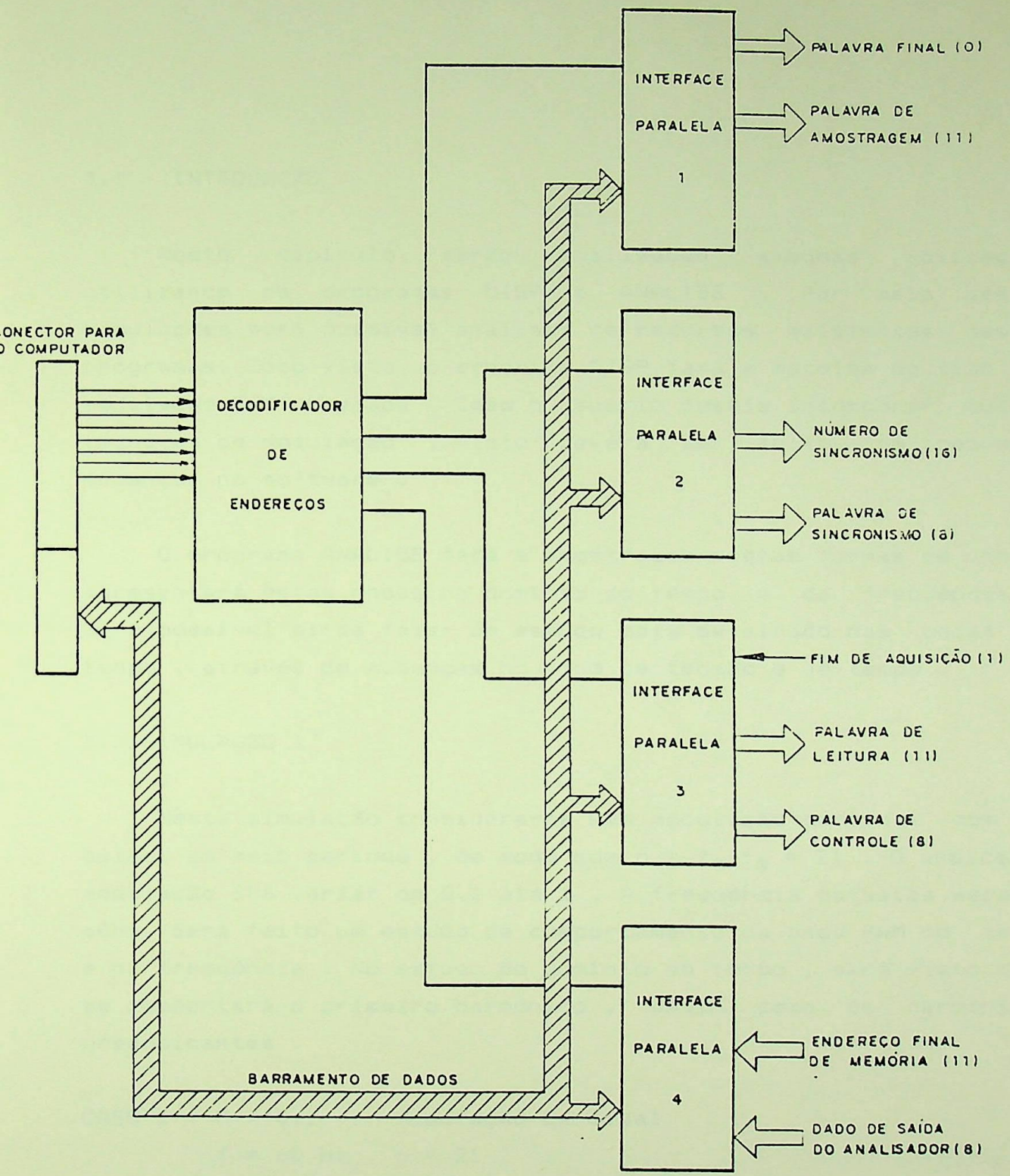


Fig. 4.8 Diagrama em bloco do circuito de interface

CAPÍTULO V
RESULTADOS OBTIDOS

5.1 - INTRODUÇÃO

Neste capítulo serão analisadas algumas aplicações utilizando os programas DISP e ANALISE . Por meio destas simulações será possível analisar os recursos existentes nestes programas. Como visto, o programa DISP fará a escolha do tipo de modulação PWM desejada . Caso o usuário deseje incorporar outras técnicas de modulação , isto deverá ser feito com pequenas mudanças no software .

O programa ANALISE fará a amostragem destas formas de onda e apresentará estas ondas no dominio do tempo e da frequência . Será possível ainda fazer um estudo mais detalhado das ondas no tempo , através de mudanças no eixo de tensão e de tempo .

5.2 SIMULAÇÃO 1

Nesta simulação considera-se uma modulação senoidal com 21 pulsos em meio período , de modo que $p = f_T/f_S = 21$. O índice de modulação irá variar de 0.1 até 1 . A frequência de saída será de 60Hz. Será feito um estudo de comportamento da onda PWM no tempo e na frequência . No estudo do domínio do tempo , será visto como se comportará o primeiro harmônico , assim como os harmônicos predominantes .

CASO 1 : M = 0.1 Modulação Senoidal
f = 60 Hz p = 21

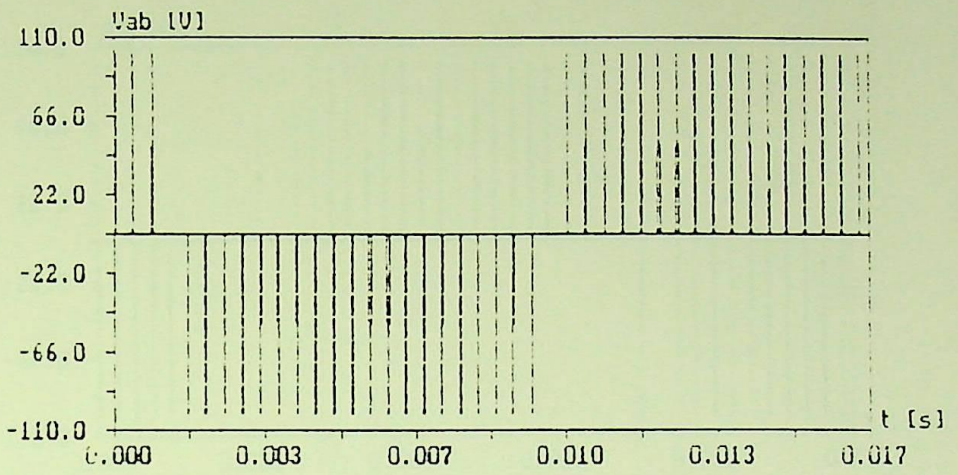


Fig 5.1 Tensão Vab para M = 0.1

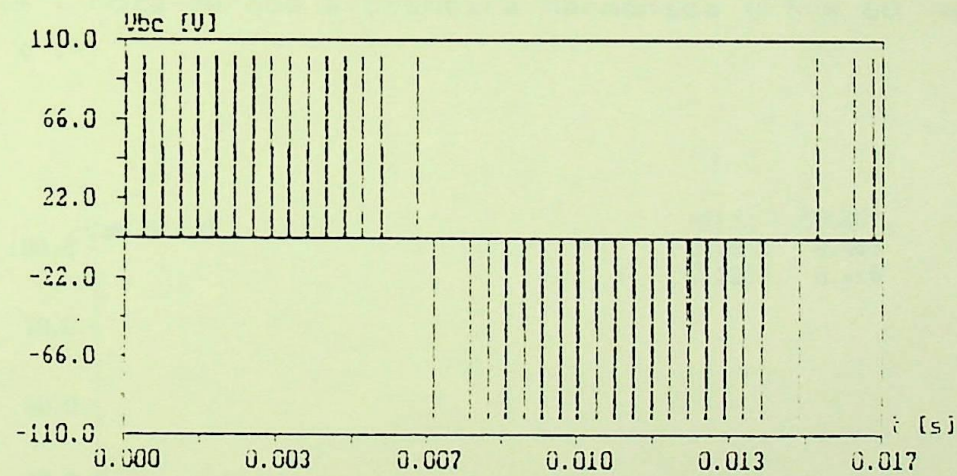


Fig. 5.2 Tensão Vbc para M = 0.1

Na Fig. 5.1 , temos a tensão Vab . Na Fig. 5.2 é apresentado a tensão Vbc e na Fig. 5.3 a tensão a tensão Vca .

Neste caso nota-se no domínio do tempo os pulsos de largura estreitas gerado pelo modulador nas três fases . Existe uma pequena diferença entre a tensão Vab e as demais devido a problemas de amostragem do sinal com pulsos estreitos de

chaveamento .

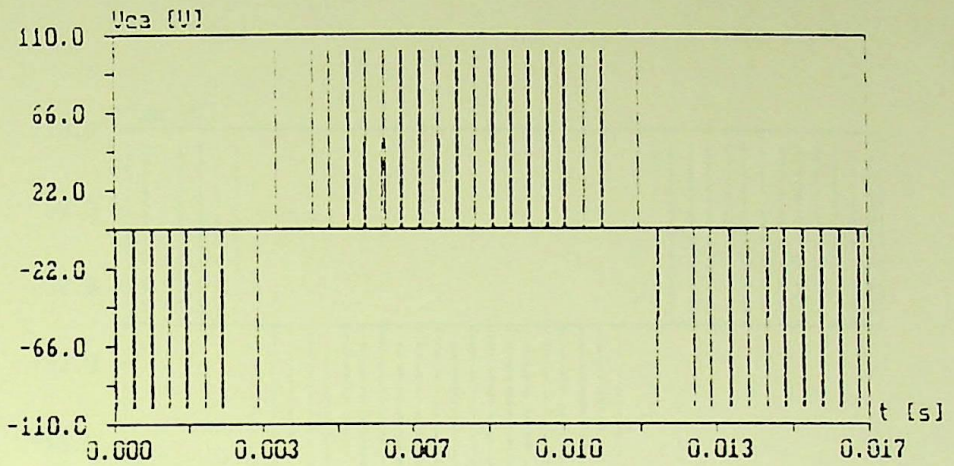


Fig. 5.3 Tensão Vca para M = 0.1

O espectro de frequência para este caso está ilustrado na Fig. 5.4 . Nota-se que a primeira harmônica (f = 60 Hz) vale 10.065 % .

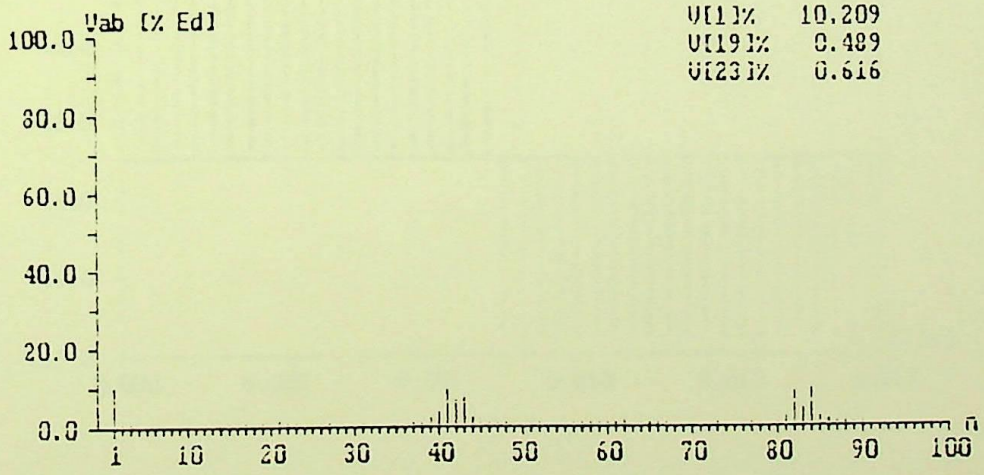


Fig 5.4 Espectro de frequência de Vab para M = 0.1

Harmônicos de ordem superior não terão importância prática, pois serão filtrados pelo sistema.

CASO 2 : M = 0.20 p = 21

f = 60 Hz Modulação Senoidal

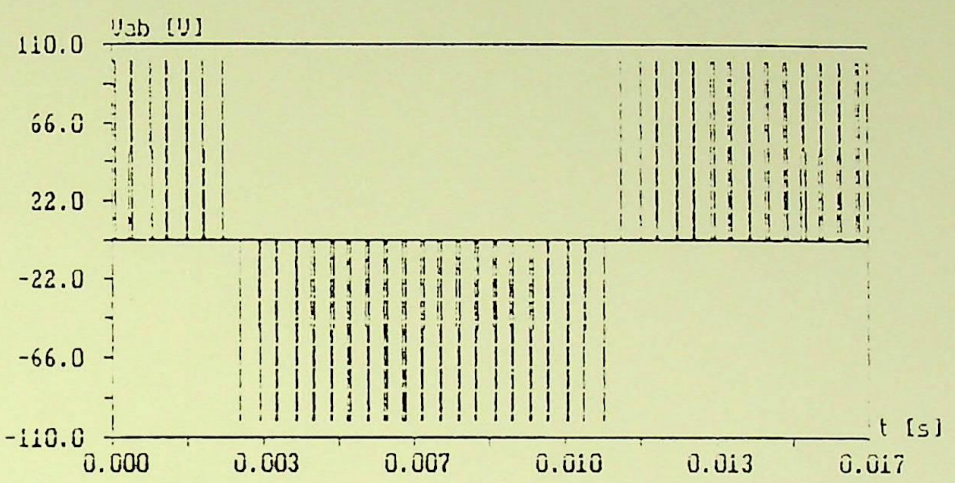


Fig. 5.5 Tensão Vab para M = 0.20

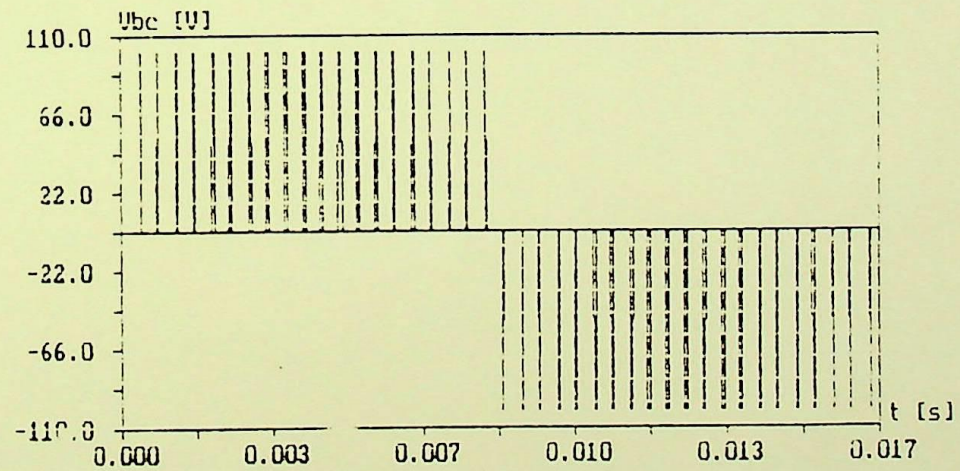


Fig. 5.6 Tensão Vbc para M = 0.20

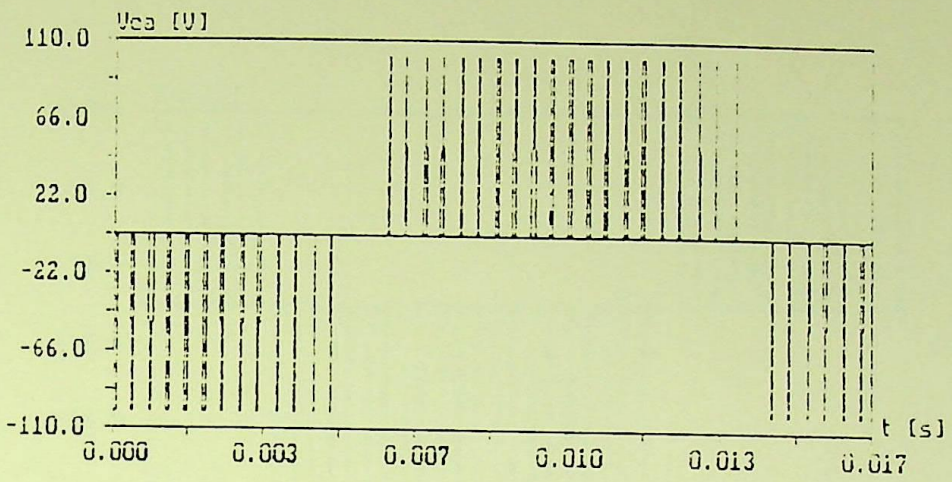


Fig. 5.7 Tensão Vca Para M = 0.20

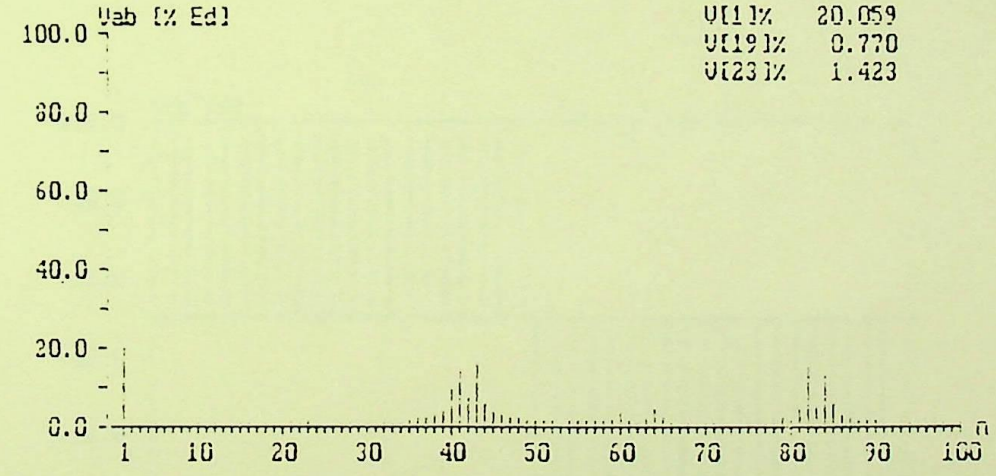


Fig. 5.8 Espectro de frequência da tensão Vab para M = 0.20

Para M = 0.20 , verifica-se que V[1]% = 19.776% .As harmônicas predominantes são as 19ª e 23ª com V[19]% = 0.760% e V[23]% = 1.403% .As raias em torno da 40ª e 80ª harmônicas não terão importância prática pois serão filtradas pelo sistema. Neste ponto é importante observar que está havendo uma linearidade entre V[1]% e o índice de modulação M .

CASO 3 : M = 0.30 p = 21

f = 60Hz Modulação Senoidal

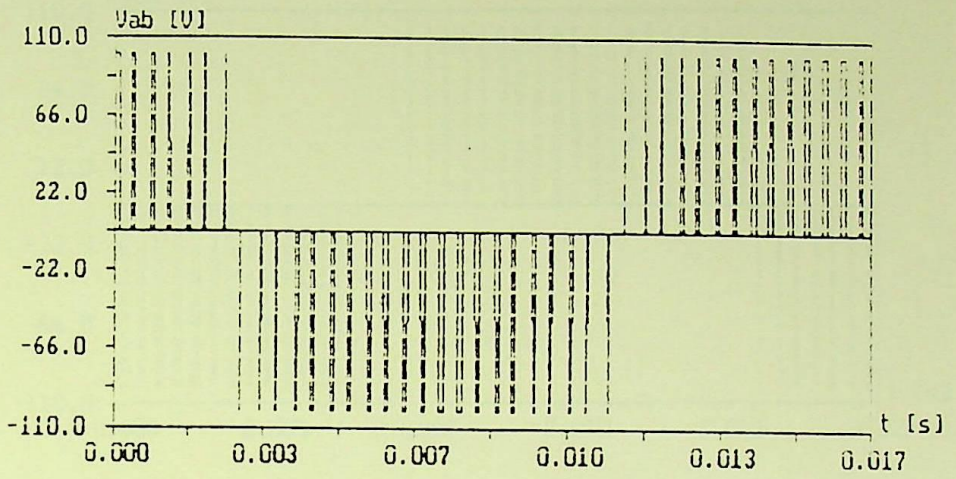


Fig. 5.9 Tensão Vab para M = 0.30

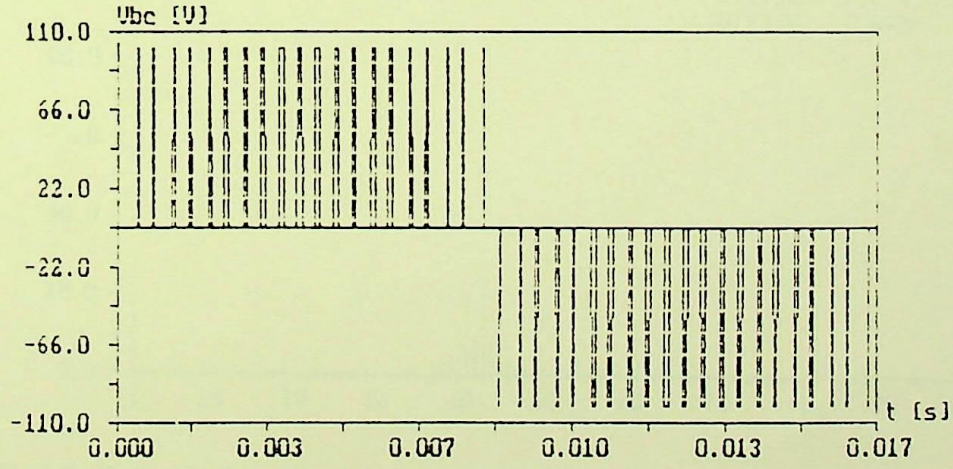


Fig. 5.10 Tensão Vbc para M = 0.30

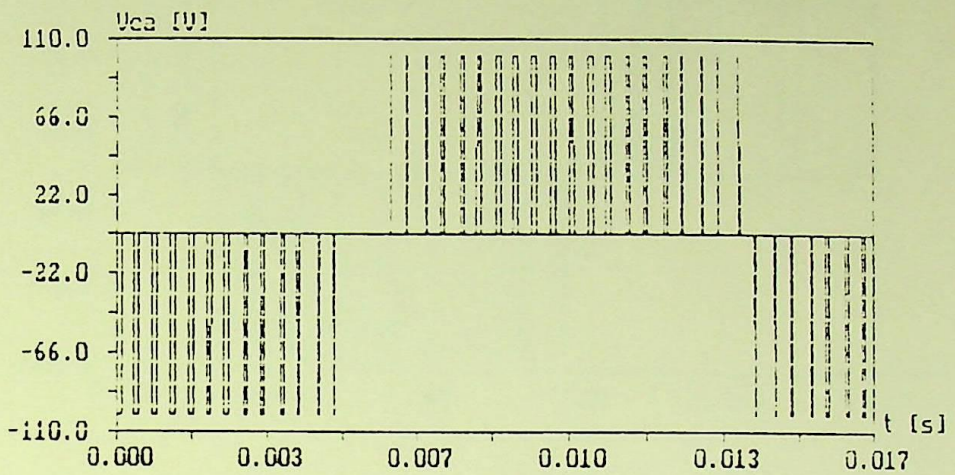


Fig.5.11 Tensão Vca para M = 0.30

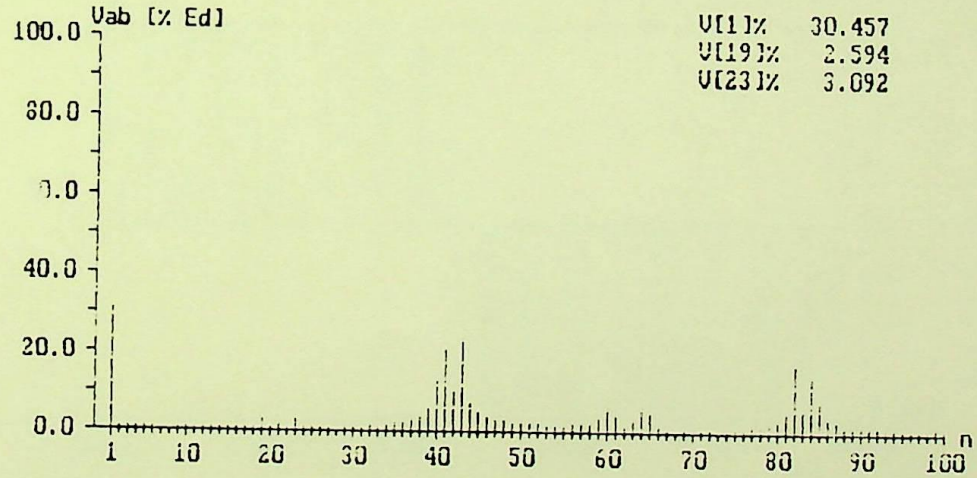


Fig. 5.12 Espectro de frequência da tensão Vab para M = 0.30

Examinando o espectro de frequência para M = 0.30 observa-se que V[1]% = 30.028% , V[19]% = 2.557 e que V[23]% = 3.048.

CASO 4 : M = 1.0 p = 21

f = 60 Hz Modulação Senoidal

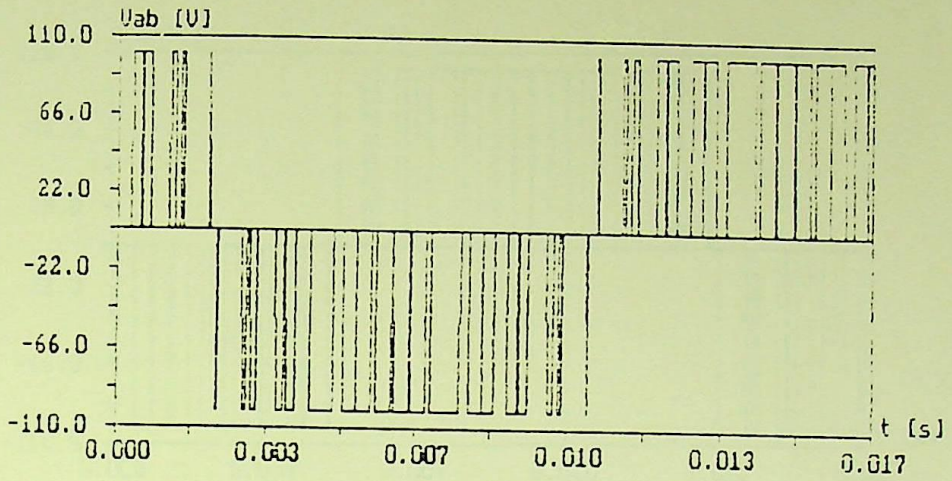


Fig. 5.13 Tensão Vab para M = 1.00

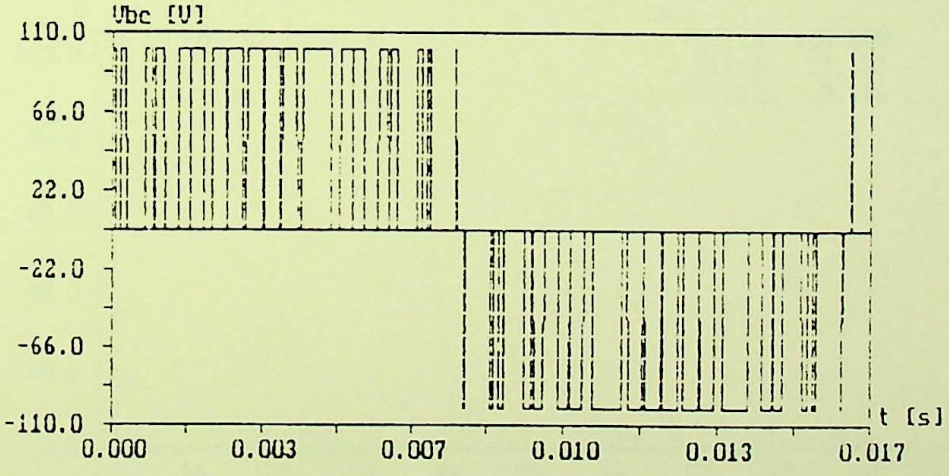


Fig. 5.14 Tensão Vbc para M = 1.00

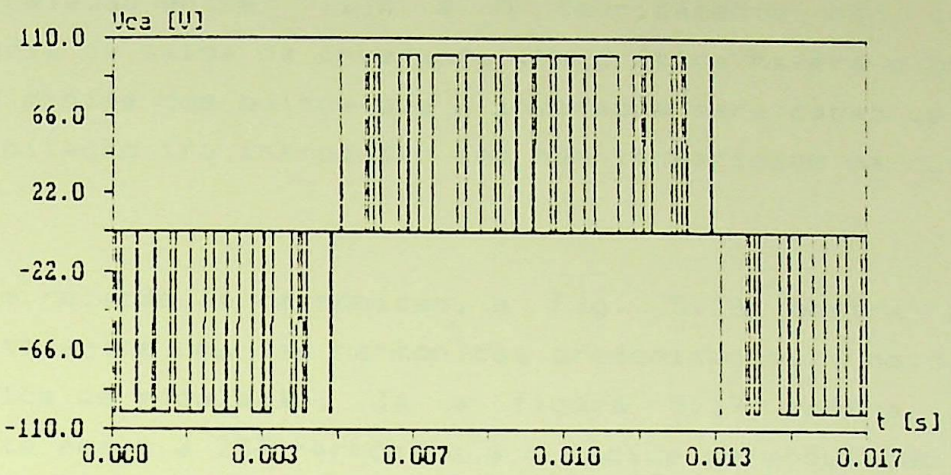


Fig. 5.15 Tensão Vca para M = 1.00

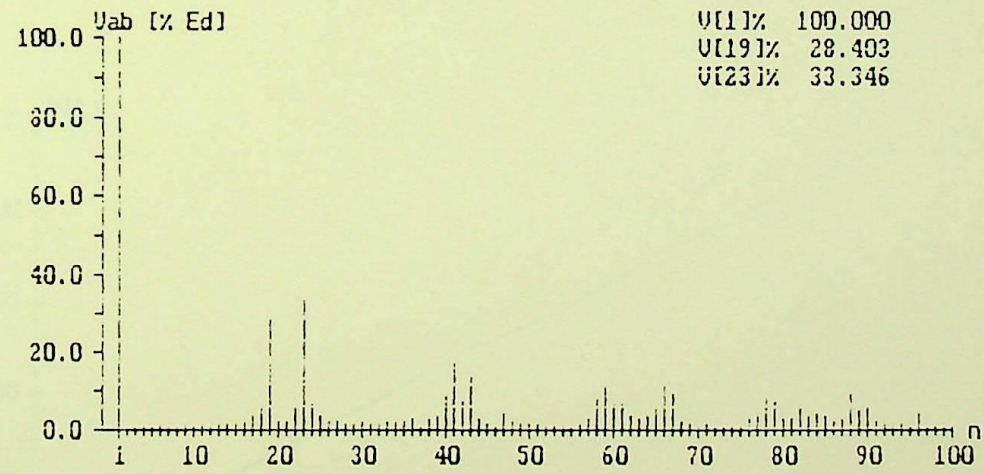


Fig. 5.16 Espectro de frequência da tensão Vab para M = 1.00

A partir da análise dos diferentes casos rodados , para M variando entre 0.1 e 1.00 será possível estabelecer a curva V[1]% em função de M. A Fig. 5.17 mostra esta relação Observa-se nesta figura uma relação linear entre V[1]% e M .Com esta relação será possível implementar um controle linear para a tensão no estator

através do índice M de modulação .

A relação entre V[1]% e M teoricamente não depende da frequência de saída da inversora .Na prática haverá o problema da largura mínima dos pulsos que a inversora será capaz de aceitar . Esta limitação irá introduzir uma não linearidade na curva V[1]% por M .

Com relação às harmônicas, a Fig. 5.18 mostra a relação existente entre uma das harmônicas predominantes , no caso 19ª , e o índice de modulação. Já a figura 5.19 mostra a relação existente entre a 23ª harmônica e o índice de modulação .

Nos dois casos verifica-se que para as harmônicas predominantes não se observa uma linearidade com o índice de modulação.

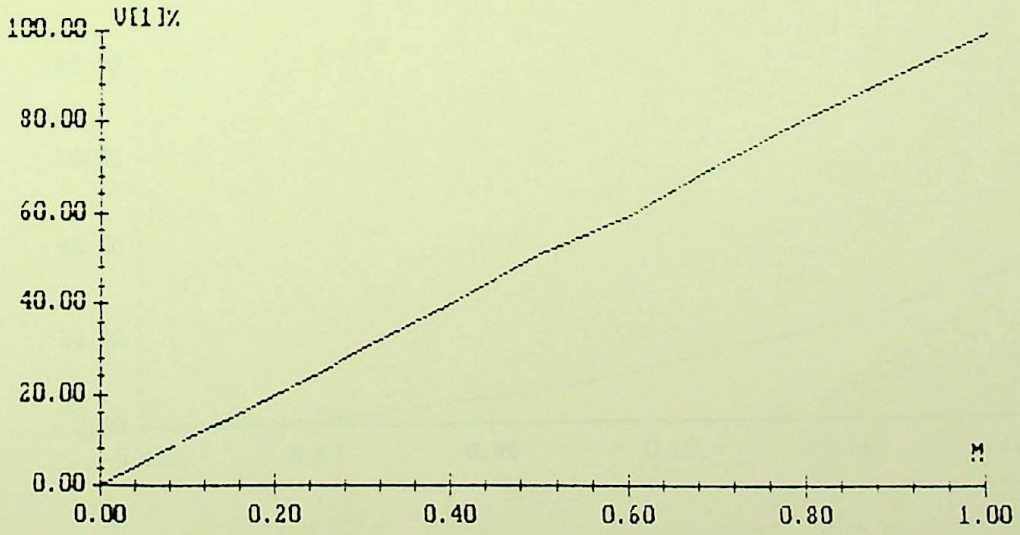


Fig. 5.17 Curva V[1]% em função de M para P = 21

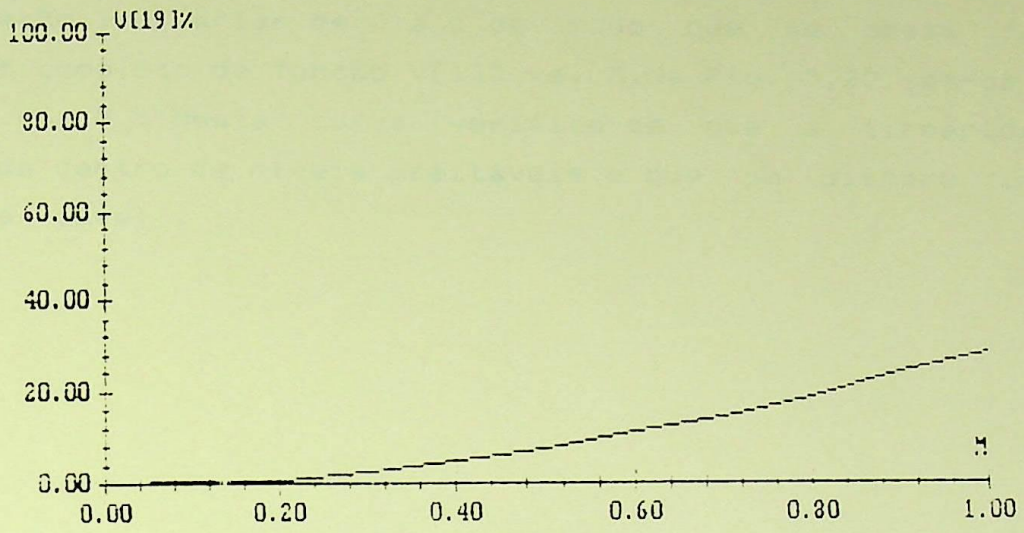


Fig. 5.18 Curva V[19]% em função de M para P = 21

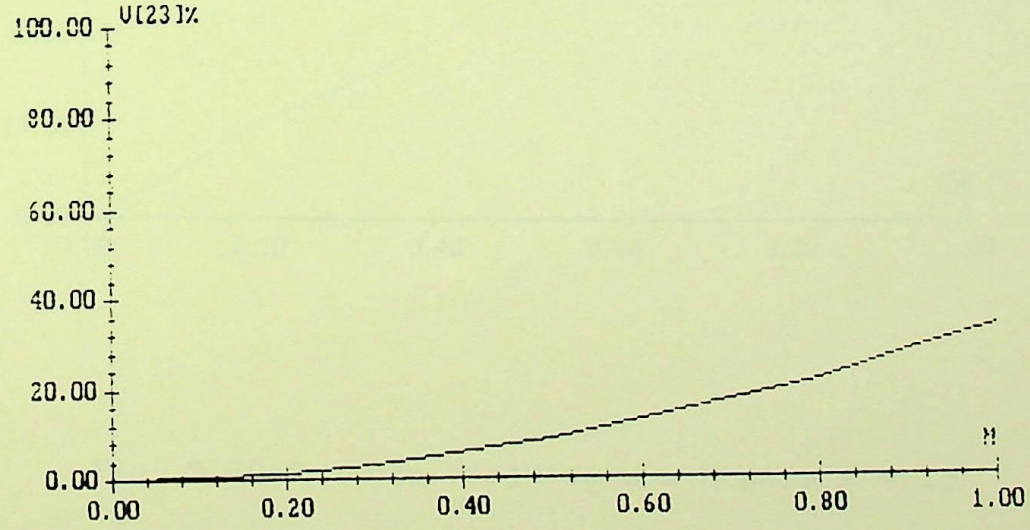


Fig. 5.19 Curva V[23]% em função de M para P = 21

5.3 SIMULAÇÃO 2 :

Nesta simulação iremos será estudado o comportamento da

primeira harmônicas quando existir 15 pulsos em meio período , e uma frequência de saída de 60 Hz .Para este caso o índice de modulação irá variar de 0 a 1 de modo que se possa fazer um estudo completo da função V[1]% vs. M.Na Fig. 5.20 tem-se a curva V[1]% vs. M . Nesta curva verifica-se que a linearidade foi mantida dentro de níveis aceitáveis e que um disparo com este modo é viável .

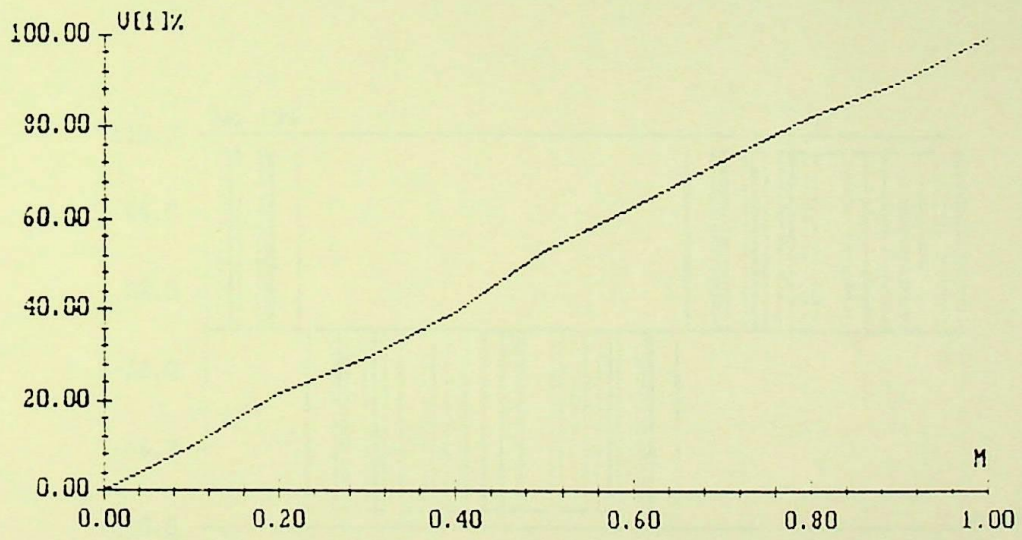


Fig. 5.20 Relação V[1]% versus M para P = 15

5.4 SIMULAÇÃO 3 :

A Fig. 5.21 apresenta a forma de onda da tensão Vab apresentada por um osciloscópio digital. Já na Fig. 5.22 temos a mesma forma de onda só que obtida por meio do Analisador Lógico e de Espectro.

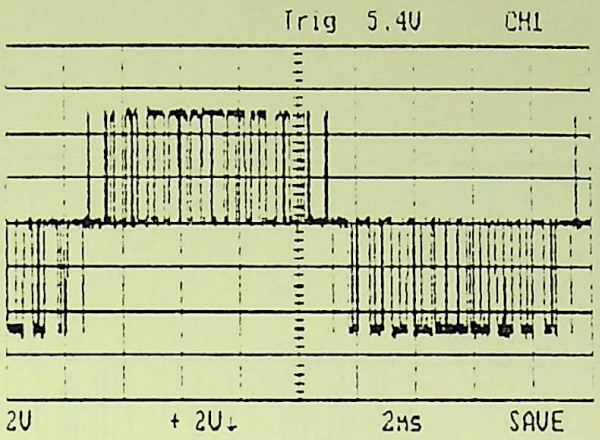


Fig. 5.21 Tensão Vab obtida por um osciloscópio digital

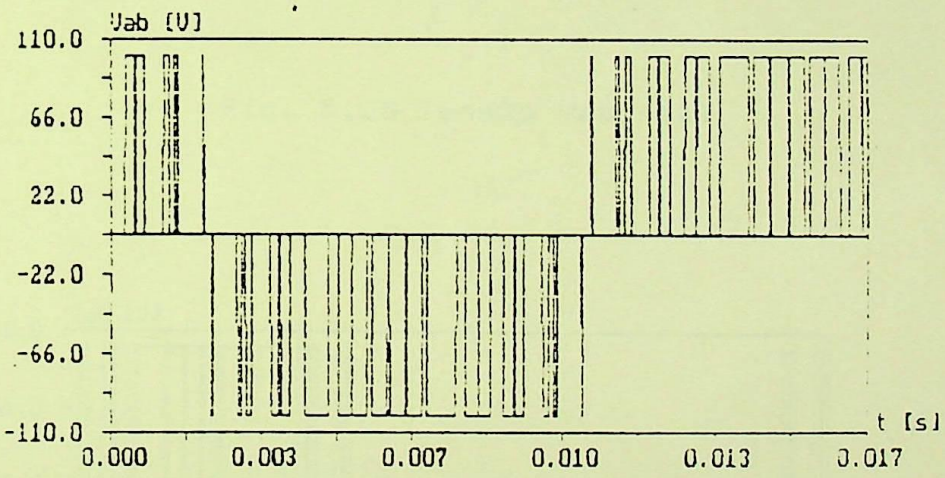


Fig. 5.22 Tensão Vab obtida no Analisador Lógico e de Espectro

5.5 VALIDAÇÃO DOS RESULTADOS :

Os resultados obtidos pelo circuito de modulação serão agora comparados com resultados teóricos obtidos por simulação .Serão analisados alguns casos já vistos anteriormentes e comparados deste modo com a teoria.

CASO 1 : P = 21 f = 60 Hz

M = 1.0 Modulação Senoidal

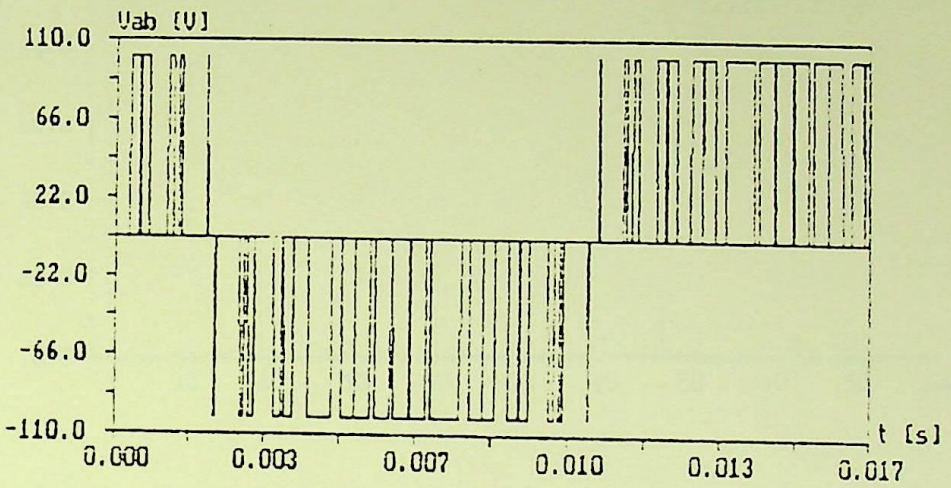


Fig. 5.23 Tensão Vab real

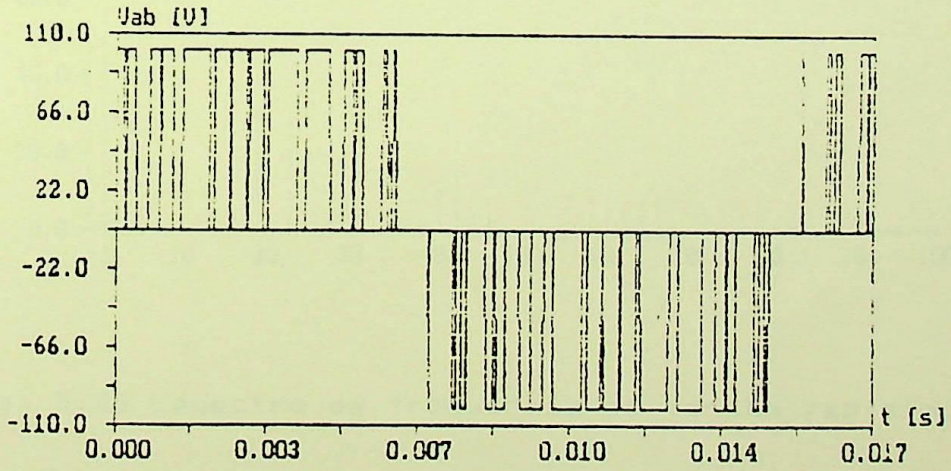


Fig. 5.24 Tensão Vab simulada

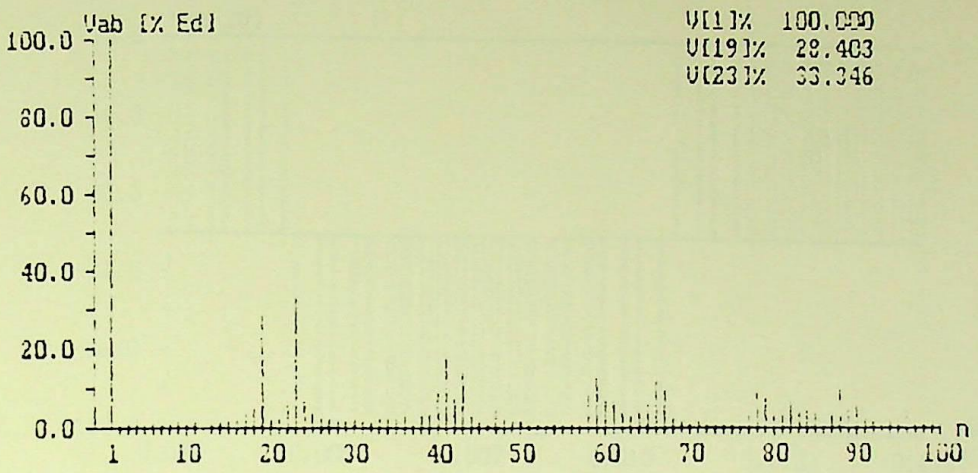


Fig. 5.25 Espectro de frequência da tensão Vab real

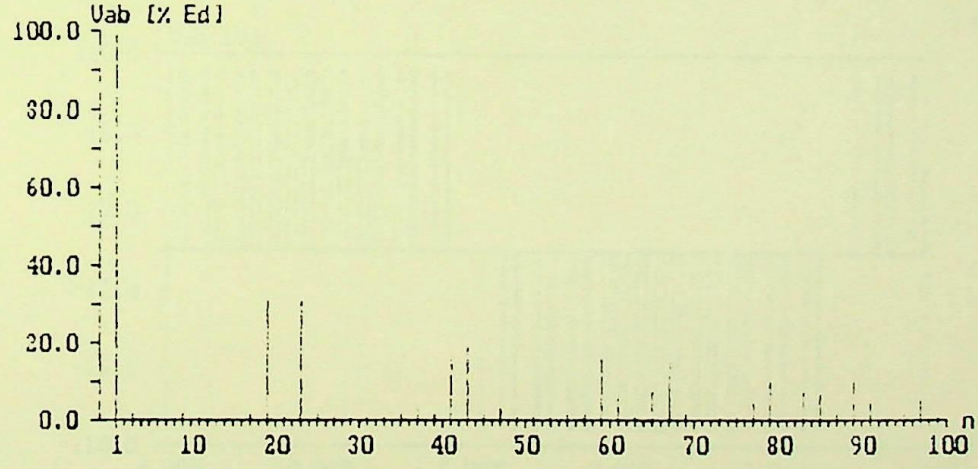


Fig. 5.26 Espectro de frequência da tensão Vab simulado

CASO 2 : P = 21 f = 60 Hz

M = 0.5 Modulação Senoidal

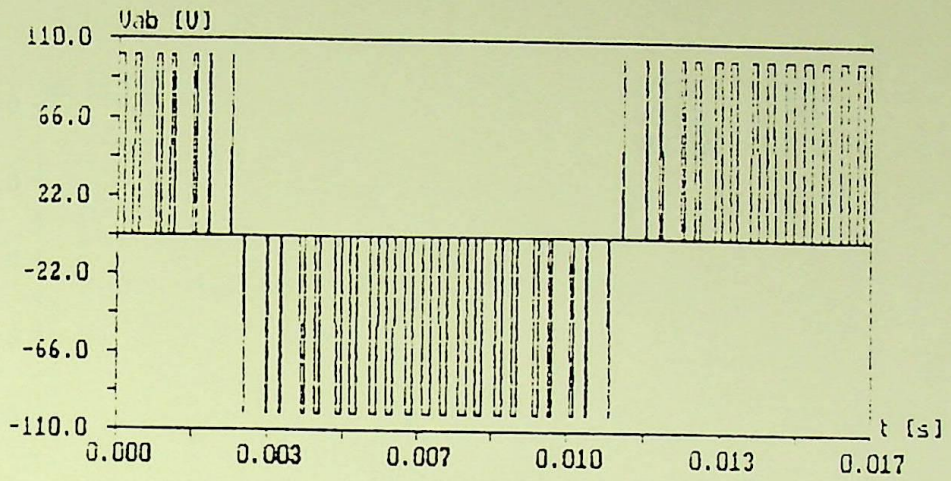


Fig. 5.27 Tensão Vab real

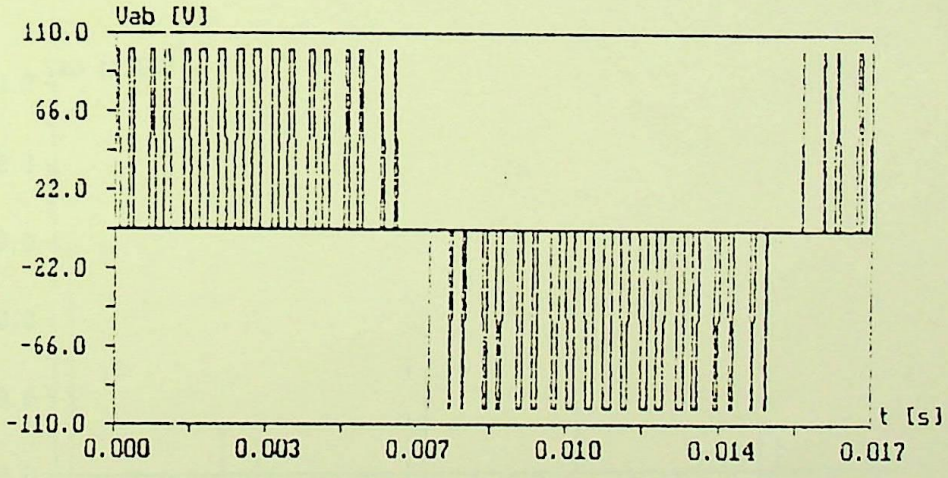


Fig. 5.28 Tensão Vab simulada

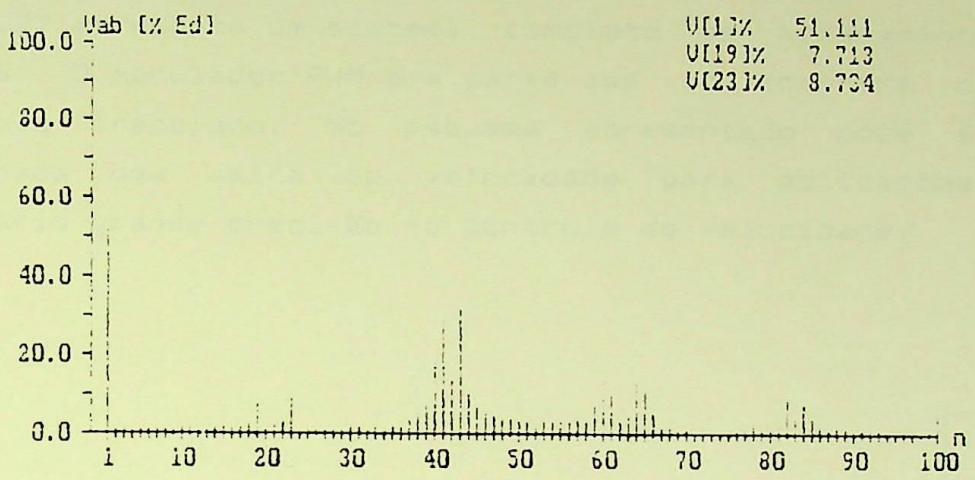


Fig. 5.29 Espectro de frequência da tensão Vab real

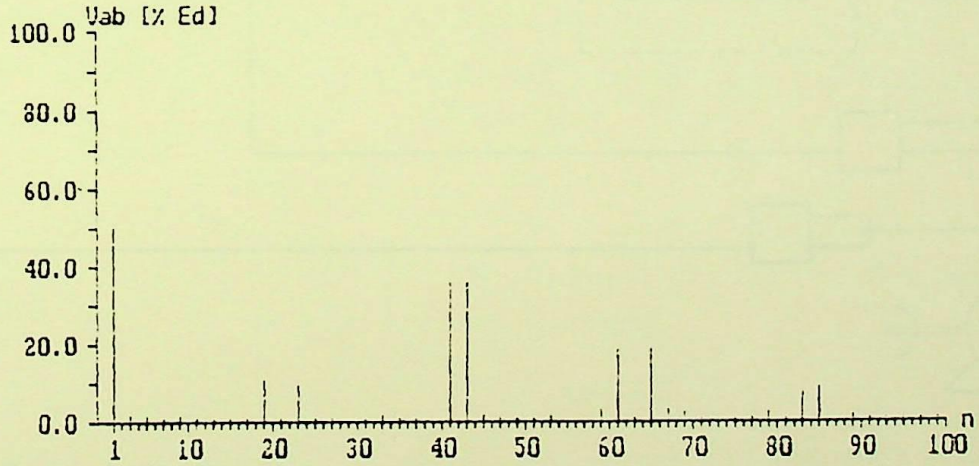


Fig. 5.30 Espectro de frequência da tensão Vab simulado

5.6 APLICAÇÃO PRÁTICA :

O modulador apresentado tem enorme aplicação em inversores PWM , sobretudo em acionamento de motor de indução trifásico. A Fig. 5.27 apresenta um sistema completo de acionamento destes motores . O modulador PWM é a parte que se encontra dentro do retângulo tracejado. No esquema apresentado pode ainda ser adicionada uma malha de velocidade para aplicações onde é necessário grande precisão no controle de velocidade.

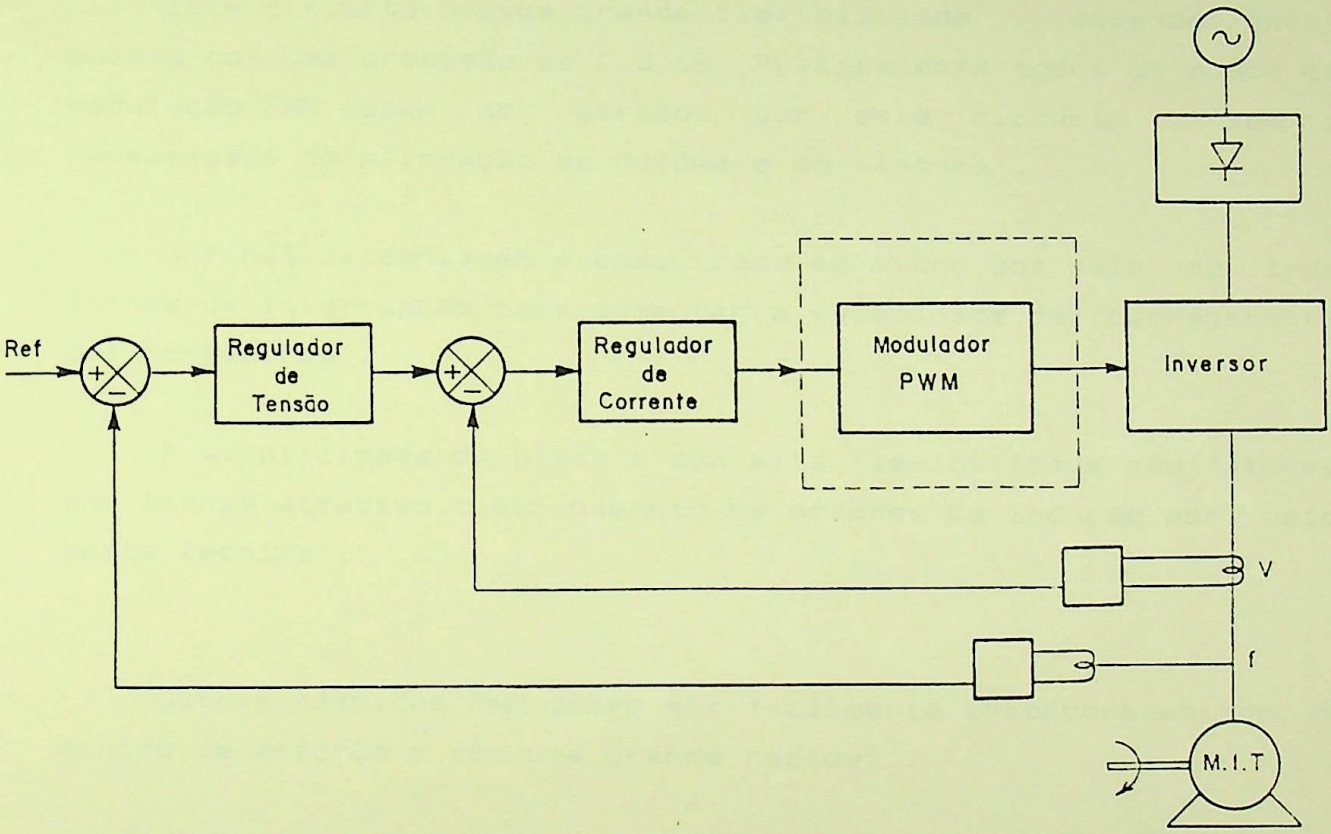


Fig. 5.31 Sistema completo de acionamento de M.I.T

CAPÍTULO VI
CONCLUSÕES

6.1 ANÁLISE DO CIRCUÍTO :

O circuíto gerador de sinais de gatilhamento foi desenvolvido para operar em conjunto com microcomputadores compativeis com IBM PC .

Este circuíto possue grande flexibilidade , podendo gerar pulsos com uma precesão de 0.5 μs .Praticamente todos os modos de modulação PWM podem ser gerados por este circuíto , sem a necessidade de alteração do hardware do sistema .

O final da contagem é comunicado ao micro por meio de très linhas de interrupção para aumentar a velocidade de carregamento dos contadores .

A simplicidade da placa e sua alta flexibilidade são fatores que tornam atrativo o acionamento de motores de indução por meio desta técnica .

Outras técnicas PWM podem ser facilmente incorporadas com um mínimo de esforço e com uma grande rapidez .

6.2 ANÁLISE DOS RESULTADOS :

Os resultados apresentados pelo circuíto de disparo mostram, através da análise harmônica , que o sistema PWM de acionamento possui a capacidade de fazer o controle da velocidade independente do controle de tensão .

Por outro lado , em várias técnicas de disparo , a variação linear entre V[1]% e M torna esta técnica adequada para acionamento de MIT .

A grande vantagem do sistema é a capacidade de fazer com rapidez e em tempo real o teste de novas técnicas PWM que futuramente serão propostas , sem com isto gastar tempo na implantação de hardware e software .

6.3 Sugestões Para Trabalhos Futuros :

Com base no circuíto desenvolvido poderão ser feitas váriaimplementações , tais como :

a- A confecção de um equipamento dedicado utilizando uma placa mãe de um IBM PC (US$ 100) e mais a interface dos temporizadores, com software em EPROM ou diskless;

b- Programação de outras técnicas de modulação tal como a Técnica De Eliminação Seletivas De Harmônicos ;

c- Utilização de técnica DMA para a transferência de dados da memória para os contadores ;

d -Implementação de funções de proteção e de alto diagnose ;

e- Implementação de controle em malha aberta ou fechada;

APÊNDICE I

A.1 Cálculo da Largura dos pulsos PWM para a modulação senoidal

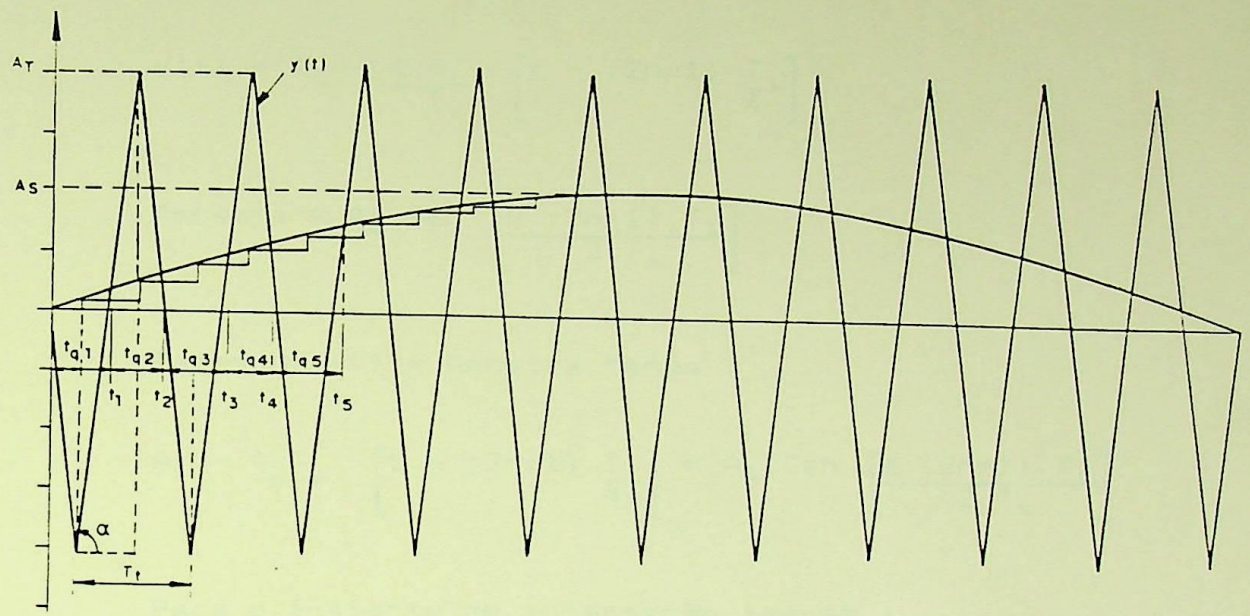


Fig. A.1 Modulação Senoidal

Da Fig. A.1 tem-se que :

$$y(t) = -A_t + \frac{4\,A_t}{T_t}\left(t - \frac{T_t}{4} \right)$$

$$Amostra_1 = A_s\ Sen\left(\frac{2\pi}{T_s}\frac{T_t}{4} \right)$$

$$Amostra_1 = A_s\ Sen\left(\frac{\pi}{2}\frac{T_t}{T_s} \right) = A_s\ Sen\left(\frac{\pi}{2p} \right)$$

No instante t_1 tem-se que : $y(t) = Amostra_1$. Assim :

$$-A_t + \frac{4\,A_t}{T_t}\ t_1 - T_t\ = A_s\ Sen\left(\frac{\pi}{2p} \right)$$

Deste modo :

$$t_1 = \frac{T_t}{2} + \frac{m\,T_t}{4}\ Sen\left(\frac{\pi}{2p} \right)$$

Para as interseções com índice pares ($t_2, t_4, \ldots$) temos :

$$(2n-1)\frac{T_t}{4} \leq t \leq (2n+1)\frac{T_t}{4}$$

$$t_g\alpha = \frac{-4\,A_t}{T_t}$$

$$y(t) = A_t - \frac{4\,A_t}{T_t}\left[t - (2n-1)\frac{T_t}{4}\right]$$

$$\text{Amostra} = A_a \,\text{Sen}\left[\frac{\pi\,(2n-1)\,T_t}{2\,T_a}\right]$$

Fazendo y(t) = Amostra temos :

$$A_t - \frac{4\,A_t}{T_t}\left[t - (2n-1)\frac{T_t}{4}\right] = A_a \,\text{Sen}\left[\frac{\pi\,(2n-1)\,T_t}{2\,T_a}\right]$$

Para o instante de interseção tem-se :

$$t = t_{pn}$$

Deste modo :

$$t_{pn} = \frac{T_t}{4}\left[2n - m\,\text{Sen}\left[\frac{\pi\,(2n-1)}{2p}\right]\right]$$

Para as interseções com índice impares ($t_3, t_5, \ldots$) temos :

$$(2n+1)\frac{T_t}{4} \leq t \leq (2n+3)\frac{T_t}{4}$$

$$t_g\alpha = \frac{4\,A_t}{T_t}$$

$$y(t) = -A_t + \frac{4\,A_t}{T_t}\left[t - (2n+1)\frac{T_t}{4}\right]$$

$$\text{Amostra} = A_a \,\text{Sen}\left[\frac{\pi\,(2n+1)\,T_t}{2\,T_a}\right]$$

Fazendo y(t) = Amostra temos :

$$-A_t + \frac{4\,A_t}{T_t}\left[t - (2n+1)\,\frac{T_t}{4}\right] = A_s\,\mathrm{Sen}\left[\frac{\pi\,(2n+1)}{2}\,\frac{T_t}{T_s}\right]$$

Para o instante de interseção tem-se :

$$t = t_{in}$$

Deste modo :

$$t_{in} = \frac{T_t}{4}\left[2n + 2 + m\,\mathrm{Sen}\left[\frac{\pi\,(2n+1)}{2p}\right]\right]$$

A largura dos pulsos será dada por :

$t_ql = t_in - t_pn$,onde n = l-1 com l par ;

$t_ql = t_{p(n+1)} - t_{i(n+1)}$, onde l = n com l ímpar ;

Assim tem-se para l par:

$$t_ql = \frac{T_t}{4}\left[2 + m\,\mathrm{Sen}\left[\frac{\pi(2n+1)}{2p}\right] + m\,\mathrm{Sen}\left[\frac{\pi(2n-1)}{2p}\right]\right]$$

Fazendo n = l-1 vem :

$$t_ql = \frac{T_t}{4}\left[2 + m\,\mathrm{Sen}\left[\frac{\pi(2l-1)}{2p}\right] + m\,\mathrm{Sen}\left[\frac{\pi(2l-3)}{2p}\right]\right]$$

Para l par vem :

$$t_ql = \frac{T_t}{4}\left[2 - m\,\mathrm{Sen}\left[\frac{\pi(2n-1)}{2p}\right] - m\,\mathrm{Sen}\left[\frac{\pi(2n-3)}{2p}\right]\right]$$

Fazendo n = l vem :

$$t_ql = \frac{T_t}{4}\left[2 - m\,\mathrm{Sen}\left[\frac{\pi(2l-1)}{2p}\right] - m\,\mathrm{Sen}\left[\frac{\pi(2l-3)}{2p}\right]\right]$$

De um modo geral temos :

$$t_q l = \frac{T_t}{2}\left[1+(-1)^{l+1}\frac{m}{2}\left\{\mathrm{Sen}\left[\frac{\pi(2l-1)}{2p}\right]+\mathrm{Sen}\left[\frac{\pi(2l-3)}{2p}\right]\right\}\right]$$

A.2 Cálculo da Largura dos pulsos PWM para a modulação com onda quadrada.

Da Fig. A.2 é possível calcular a largura dos pulsos para a modulação com onda quadrada tendo três pulsos em meio período.

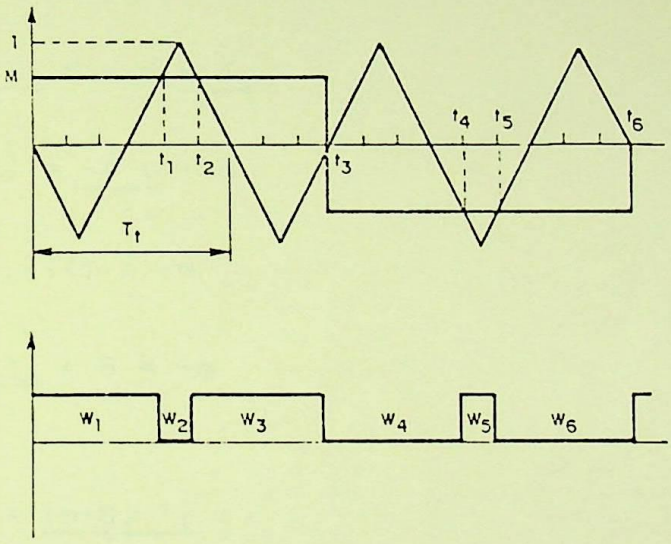


Fig. A2 Modulação com onda quadrada tendo 3 pulsos em meio período.

Para $\frac{T_t}{4} \leq t \leq \frac{3\,T_t}{4}$ tem-se :

$$y(t) = \frac{4\,t}{T_t} - 2$$

Em t_1 : $y(t) = m$, assim :

$$\frac{4\,t_1}{T_t} - 2 = m$$

$$t_1 = \frac{(m+2)\,T_t}{4}$$

Para $\frac{3\ T_t}{4} \leq t \leq T_t$ tem-se :

$$y(t) = \frac{-4\ t}{T_t} + 4$$

Em t_2 : $y(t) = m$

$$\frac{-4\ t_2}{T_t} + 4 = m$$

$$t_2 = \frac{(4-m)\ T_t}{4}$$

$$t_3 = 1.5\ T_t$$

Para $\frac{7\ T_t}{4} \leq t \leq \frac{9\ T_t}{4}$

$$y(t) = \frac{-4\ t}{T_t} + 8$$

Em t_4 : $y(t) = -m$

$$\frac{-4\ t_4}{T_t} + 8 = -m$$

$$t_4 = \frac{(m+8)\ T_t}{4}$$

Para $\frac{9\ T_t}{4} \leq t \leq \frac{11\ T_t}{4}$

$$y(t) = \frac{4\ t}{T_t} - 10$$

Em t_5 : $y(t) = -m$

$$-m = \frac{4\ t_5}{T_t} - 10$$

$$t_5 = \frac{(10-m)\ T_t}{4}$$

$$t_6 = 3\ T_t$$

$$W_1 = t_1 = (m+2)\ \frac{T_t}{4}$$

$$W_2 = t_2 - t_1 = \frac{T_t\ (1-m)}{4}$$

$$W_3 = t_3 - t_2 = \frac{T_t\ (m+2)}{4}$$

$$W_4 = t_5 - t_4 = T_t\ (m+2)$$

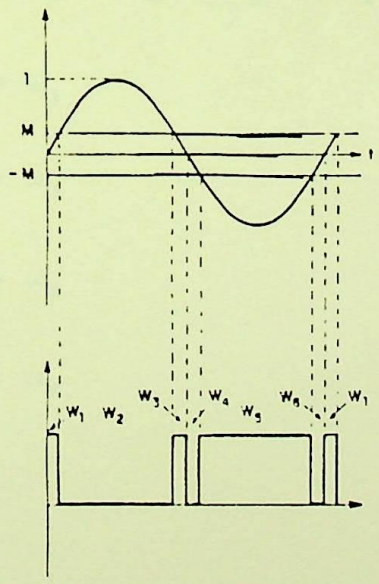
$$W_5 = t_5 - t_4 = \frac{T_t\ (1-m)}{2}$$

$$W_6 = t_6 - t_5 = \frac{(m+2)\ T_t}{4}$$

Para a modulação com 5 e 7 pulsos em meio período é possível obter a largura dos pulsos utilizando raciocínio semelhante.

A.3 Cálculo da Largura dos pulsos PWM para a modulação dos flancos.

O cálculo da largura dos pulsos para a modulação dos flancos será feito baseado na Fig. A3.



$T_1 = \text{arc Sen}(M)$

$T_2 = \pi - 2T_1$

$N_p = 6$
$W_1 = T_1$
$W_2 = T_2$
$W_3 = W_1$
$W_4 = W_1$
$W_5 = W_2$
$W_6 = W_1$

Fig. A3 - Modulação Dos Flancos

$$y(t) = \text{Sen}\left\{\frac{2\ \pi\ t}{T_a}\right\}$$

Em t_1, tem-se :

$$y(t) = m$$

$$\text{Sen}\left\{\frac{2\ \pi\ t_1}{T_s}\right\} = m$$

$$t_1 = \frac{T_s\ \text{arc Sen}\ (m)}{2\pi}$$

$$t_2 = \frac{T_s}{2} - t_1$$

$$t_4 = \frac{T_s}{2} + t_1$$

$$t_5 = T_s - t_1$$

Deste modo tem-se :

$$W_1 = t_1$$

$$W_2 = t_2 - t_1 = \frac{T_s}{2} - 2\ t_1$$

$$W_3 = W_1$$

$$W_4 = W_1$$

$$W_5 = W_2$$

$$W_6 = W_1$$

B.1 PROGRAMA EM LINGUAGEM DE MAQUINA PARA IMPLEMENTAR O MODULADOR PWM .

OBS: ESTE PROGRAMA SERA CHAMADO PELO PROGRAMA DISP, ESCRITO EM LINGUAGEM DE ALTO NIVEL.

```
CODE    SEGMENT BYTE PUBLIC

ASSUME CS:CODE,DS:DATA


;-------------------------------------------------------------;
;         ESTE PROGRAMA IMPLEMENTA O MODULADOR PWM             ;
;-------------------------------------------------------------;
            PUBLIC FIRE
FIRE        PROC NEAR
            CLI

            CALL INICIO                 ;CHAMAR SUB-ROTINA PARA
                                        ;INICIALIZAR PARAMETROS.

            CALL SET_COUNTER            ;CHAMA A SUB-ROTINA PARA
                                        ;PROGRAMAR CONTADORES SEM
                                        ;CARREGA-LOS.

            CALL SET_IRQ2               ;CHAMA SUB-ROTINA QUE
                                        ;PROGRAMA A INTERRUPCAO
                                        ;IRQ2, E BLOQUEIA DEMAIS
                                        ;INTERRUPCOES.

            CALL SET_IRQ3               ;CHAMA A SUB-ROTINA QUE
                                        ;PROGRAMA A INTERRUPCAO
                                        ;IRQ3, E BLOQUEIA DEMAIS
                                        ;INTERRUPCOES.

            CALL SET_IRQ5               ;CHAMA A SUB-ROTINA END
                                        ;PROGRAMA A INTERRUPCAO
```

```
                                    ;IRQ5, E BLOQUEIA DEMAIS
                                    ;INTERRUPCOES.

        MOV   POINTER1,08H
        MOV   BX,POINTER1
        MOV   AX,LARG[BX]
        ADD   BX,2H
        MOV   CX,LARG[BX]
        ADD   BX,2H
        MOV   POINTER1,BX
        ADD   CX,AX

        CALL LOAD_C_A_G_1           ;COLOCA PRIMEIRO VALOR NO
                                    ;CONTADOR
        MOV   AX,320H
        CALL LOAD_C_A_G_4



        MOV   POINTER2,04H
        MOV   BX,POINTER2
        MOV   AX,LARG[BX]
        ADD   BX,2H
        MOV   CX,LARG[BX]
        ADD   BX,2H
        MOV   POINTER2,BX
        ADD   CX,AX

        CALL LOAD_C_A_G_2           ;COLOCA PRIMEIRO VALOR NO
                                    ;CONTADOR.
        MOV   AX,320H
        CALL LOAD_C_A_G_5



        MOV   POINTER3,00H
```

```
        MOV   BX,POINTER3
        MOV   AX,LARG[BX]
        ADD   BX,2H
        MOV   CX,LARG[BX]
        ADD   BX,2H
        MOV   POINTER3,BX
        ADD   CX,AX

        CALL LOAD_C_A_G_3          ;COLOCA PRIMEIRO VALOR NO
                                   ;CONTADOR
        MOV   AX,320H
        CALL LOAD_C_A_G_6


MOV     DX,PORTA                   ;CARREGA DX COM O END.
                                   ;DAS PORTAS DE CONTROLE
                                   ;DOS GATES

MOV     AL,ESTADO                  ;CARREGAR AL COM O ESTADO
                                   ;ANTERIOR DAS PORTAS

AND     AL,0F8H                    ;PRODUZIR UM PULSO NOS
                                   ;GATES DOS CONTADORES
                                   ;DO TIMER A E TIMER B
OUT     DX,AL
OR      AL,07H
OUT     DX,AL

MOV     ESTADO,AL                  ;SALVAR ESTADO ATUAL DAS
                                   ;PORTAS.
        STI
        IN    AL,21H
        AND   AL,0D3H
        OUT   21H,AL
        mov   al,68h
        out   20h,al
```

```
        MOV   BX,POINTER1
        MOV   AX,LARG[BX]
        ADD   BX,2H
        MOV   CX,LARG[BX]
        ADD   BX,2H
        MOV   POINTER1,BX
        ADD   CX,AX



        CALL LOAD_C_A_G_1

        MOV   BX,POINTER3
        MOV   AX,LARG[BX]
        ADD   BX,2H
        MOV   CX,LARG[BX]
        ADD   BX,2H
        MOV   POINTER3,BX
        ADD   CX,AX

        CALL LOAD_C_A_G_3


        MOV   BX,POINTER2
        MOV   AX,LARG[BX]
        ADD   BX,2H
        MOV   CX,LARG[BX]
        ADD   BX,2H
        MOV   POINTER2,BX
        ADD   CX,AX

        CALL LOAD_C_A_G_2

LB1:    WAIT                          ;ESPERA POR INTERRUPCAO.
        STI
        JMP LB1                       ;APOS ATENDER UMA
```

```
                                        ;INTERRUPCAO,ESPERA-SE POR
                                        ;OUTRA.

FIRE        ENDP

;-------------------------------------------------------------;
; ESTA ROTINA ESTABELECE UMA ROTINA ESPECIAL DE INTERRPCAO PARA;
;IRQ2  E  HABILITA  INTERRUPCAO  PARA  IRQ2  REPROGRAMANDO  O;
;CONTROLADOR DE INTERRUPCAO.                                  ;
;-------------------------------------------------------------;
SET_IRQ2  PROC NEAR


     PUSH      BP                   ;SALVA BP
     MOV       BP,SP
     SUB       SP,2                 ;ALOCA ESPACO PARA
                                    ;ARMAZENAMENTO DE UMA
                                    ;PALAVRA TEMPORARIA.

     IN        AL,21H               ;SALVA MASCARA DE
                                    ;INTERRUPCAO
     MOV       CS:OLD_MASK,AL

     MOV       [BP-2],DS            ;SALVA DS NA MEMORIA

     XOR       AX,AX                ;ENDERECA SEGMENTO O COM
                                    ;DS
     MOV       DS,AX

     MOV       WORD PTR DS:(0AH*4),OFFSET CODE:MY_IRQ2
     MOV       DS:(0AH*4+2),CS      ;CS CONTEM O SEGMENTO PARA
                                    ;ENDERECAR MY_IRQ2.

     MOV       DS,[BP-2]            ;RESTAURA DS
     MOV       SP,BP

     POP       BP                   ;RESTAURA BP
```

```
     RET                                ;RETORNA

 SET_IRQ2  ENDP

;-------------------------------------------------------------;
; ESTA ROTINA ESTABELECE UMA ROTINA ESPECIAL DE INTERRPCAO PARA;
;IRQ3  E  HABILITA  INTERRUPCAO  PARA  IRQ3  REPROGRAMANDO  O;
;CONTROLADOR DE INTERRUPCAO.                                  ;
;-------------------------------------------------------------;
 SET_IRQ3  PROC NEAR


     PUSH      BP                     ;SALVA BP
     MOV       BP,SP
     SUB       SP,2                   ;ALOCA ESPACO PARA ARMAZENAMEN

                                      ;ARMAZENAMENTO DE UMA
                                      ;PALAVRA TEMPORARIA

     IN        AL,21H                 ;SALVA MASCARA DE
                                      ;INTERRUPCAO
     MOV       CS:OLD_MASK,AL

     MOV       [BP-2],DS              ;SALVA DS NA MEMORIA

     XOR       AX,AX                  ;ENDERECA SEGMENTO O COM
                                      ;DS
     MOV       DS,AX

     MOV       WORD PTR DS:(OBH*4),OFFSET CODE:MY_IRQ3
     MOV       DS:(OBH*4+2),CS        ;CS CONTEM O SEGMENTO PARA
                                      ;ENDERECAR MY_IRQ3.

     MOV       DS,[BP-2]              ;RESTAURA DS
     MOV       SP,BP

     POP       BP                     ;RESTAURA BP
```

```
        RET                             ;RETORNA


 SET_IRQ3  ENDP

;-----------------------------------------------------------------;
; ESTA ROTINA ESTABELECE UMA ROTINA ESPECIAL DE INTERRPCAO PARA;
;IRQ2 E HABILITA INTERRUPCAO PARA IRQ2 REPROGRAMANDO 0;
;CONTROLADOR DE INTERRUPCAO.                                     ;
;-----------------------------------------------------------------;

 SET_IRQ5  PROC NEAR


        PUSH    BP                      ;SALVA BP
        MOV     BP,SP
        SUB     SP,2                    ;ALOCA ESPACO PARA
                                        ;ARMAZENAMENTO DE UMA
                                        ;PALAVRA TEMPORARIA.

        IN      AL,21H                  ;SALVA MASCARA DE
                                        ;INTERRUPCAO
        MOV     CS:OLD_MASK,AL

        MOV     [BP-2],DS               ;SALVA DS NA MEMORIA

        XOR     AX,AX                   ;ENDERECA SEGMENTO 0 COM
        MOV     DS,AX                   ;DS

        MOV     WORD PTR DS:(ODH*4),OFFSET CODE:MY_IRQ5
        MOV     DS:(ODH*4+2),CS         ;CS CONTEM O SEGMENTO PARA
                                        ;ENDERECAR MY_IRQ5.

        MOV     DS,[BP-2]               ;RESTAURA DS
        MOV     SP,BP

        POP     BP                      ;RESTAURA BP
```

```
        RET                                     ;RETORNA


SET_IRQ5  ENDP



;-------------------------------------------------------------;
; MY_IRQ2                                                     ;
;                                                             ;
; ESTA PROCEDURE MANIPULA A INTERRUPCAO IRQ2.NELA SAO SALVADOS ;
;APENAS OS REGISTROS USADOS.E FEITO UM BLOQUEIO DE OUTRAS IN-  ;
;TERRUPCOES DURANTE A EXECUCAO.                               ;
;-------------------------------------------------------------;


MY_IRQ2  PROC NEAR

            CLI                                 ;BLOQUEAR INTERRUPCAO
            PUSH AX
OK2:        PUSH BX
            PUSH CX
            PUSH DX
            MOV  BX,POINTER1
            MOV  AX,LARG[BX]
            ADD  BX,2H
            MOV  CX,LARG[BX]
            ADD  BX,2H
            CMP  BX,NP2
            JB   MY21
            MOV  BX,0008H
MY21:       MOV  POINTER1,BX
            ADD  CX,AX

            CALL LOAD_C_A_G_1
```

```
          MOV  DX,CONTROL_T_C
          MOV  AL,MODO_1_C_0
         OUT  DX,AL

         MOV  AX,320H
         CALL LOAD_C_A_G_4

        MOV      AL,20H
        OUT      20H,AL                    ;ATENDIDA.
        POP  DX
        POP  CX
        POP  BX
END2:   POP  AX


        STI                                ;HABILITAR NOVAS
                                           ;INTERRUPCOES
        IRET                               ;RETORNAR DE MY_IRQ2.

MY_IRQ2   ENDP

;-----------------------------------------------------------------;
; MY_IRQ3                                                         ;
;                                                                 ;
; ESTA PROCEDURE MANIPULA A INTERRUPCAO IRQ3.NELA SAO SALVADOS    ;
;APENAS OS REGISTROS USADOS.E FEITO UM BLOQUEIO DE OUTRAS IN-     ;
;TERRUPCOES DURANTE A EXECUCAO.                                   ;
;-----------------------------------------------------------------;

MY_IRQ3  PROC NEAR

        CLI                                ;BLOQUEAR INTERRUPCAO
        PUSH AX
        PUSH BX
        PUSH CX
        PUSH DX
           MOV  BX,POINTER2
```

```
            MOV   AX,LARG[BX]
            ADD   BX,2H
            MOV   CX,LARG[BX]
            ADD   BX,2H
            CMP   BX,NP2
            JB    MY31
            MOV   BX,0008H
MY31:       MOV   POINTER2,BX
            ADD   CX,AX

         CALL LOAD_C_A_G_2

         MOV   DX,CONTROL_T_C
         MOV   AL,MODO_1_C_1
         OUT   DX,AL

         MOV   AX,320h
         CALL LOAD_C_A_G_5


         MOV   AL,20H
         OUT   20H,AL                    ;ATENDIDA.
         POP   DX
         POP   CX
         POP   BX
END3:    POP   AX
         STI                                ;HABILITAR NOVAS
                                            ;INTERRUPCOES
         IRET                               ;RETORNAR DE MY_IRQ3.

 MY_IRQ3   ENDP

 ;-----------------------------------------------------------;
 ; MY_IRQ5                                                   ;
 ;                                                           ;
 ; ESTA PROCEDURE MANIPULA A INTERRUPCAO IRQ5.NELA SAO SALVADOS ;
 ;APENAS OS REGISTROS USADOS.E FEITO UM BLOQUEIO DE OUTRAS IN-  ;
 ;TERRUPCOES DURANTE A EXECUCAO.                             ;
```

```
;-------------------------------------------------------------;

MY_IRQ5  PROC NEAR

        CLI                                  ;BLOQUEAR INTERRUPCAO
        PUSH AX
        PUSH BX
        PUSH CX
        PUSH DX
           MOV  BX,POINTER3
           MOV  AX,LARG[BX]
           ADD  BX,2H
           MOV  CX,LARG[BX]
           ADD  BX,2H
           CMP  BX,NP2
           JB   MY51
           MOV  BX,0008H
MY51:      MOV  POINTER3,BX
           ADD  CX,AX

        CALL LOAD_C_A_G_3

        MOV  DX,CONTROL_T_C
        MOV  AL,MODO_1_C_2
        OUT  DX,AL


        MOV     AX,320H
        CALL LOAD_C_A_G_6

        MOV      AL,20H
        OUT      20H,AL                      ;ATENDIDA.
        MOV  BUSY,0
        MOV  AL,0D3H
        OUT  21H,AL
        POP  DX
        POP  CX
        POP  BX
```

```
END5:   POP  AX
        STI                                  ;HABILITAR NOVAS
                                             ;INTERRUPCOES
        IRET                                 ;RETORNAR DE MY_IRQ2.

 MY_IRQ5   ENDP



 ;-------------------------------------------------------------;
 ; LOAD_C_A_G_1                                                ;
 ; ESTA PROCEDURE CARREGA OS CONTADORES 0 E 1 DO TIMER A.ISTO E ;
 ; FEITO DA SEGUINTE MANEIRA:                                  ;
 ;          * REGISTRO AX:VALOR PARA CONTADOR 0  DO TIMER A    ;
 ;          * REGISTRO CX:VALOR PARA CONTADOR 1  DO TIMER A    ;
 ; TODOS OS REGISTROS AFETADOS SERAO RESTAURADOS AOS VALORES    ;
 ;ORIGINAIS.                                                   ;
 ; OS CONTADORES SAO DISPARADOS APOS TEREM SIDOS CARREGADOS     ;
 ;-------------------------------------------------------------;

 LOAD_C_A_G_1 PROC NEAR


     PUSH      AX                         ;SALVAR RESGISTROS
     PUSH      DX

     MOV       DX,TIMER_A_0               ;CARREGAR DX COM O
                                          ;ENDERECO DO CONTADOR 0
                                          ;0 DO TIMER A.
     OUT       DX,AL                      ;CARREGAR CONTADOR 0 DO
                                          ;TIMER A.
     MOV       AL,AH
     OUT       DX,AL

     MOV       DX,TIMER_A_1               ;CARREGAR DX COM O
                                          ;ENDERECO DO CONTADOR 1 DO
                                          ;DO TIMER A.
```

```
    MOV     AX,CX


    OUT     DX,AL                   ;CARREGAR CONTADOR 1 DO
    MOV     AL,AH                   ;TIMER A
    OUT     DX,AL




    POP     DX                      ;RECUPERAR REGISTROS.
    POP     AX

    RET                             ;RETORNAR DE LOAD_COUNTER.


    LOAD_C_A_G_1 ENDP
;-------------------------------------------------------------;
; LOAD_C_A_G_2                                                ;
; ESTA PROCEDURE CARREGA OS CONTADORES 2 E 0 DO TIMER A E B    ;
; RESPECTIVAMENTE.ISTO E FEITO DA SEGUINTE MANEIRA:           ;
;          * REGISTRO AX:VALOR PARA CONTADOR 2  DO TIMER A    ;
;          * REGISTRO CX:VALOR PARA CONTADOR 0  DO TIMER B    ;
; TODOS OS REGISTROS AFETADOS SERAO RESTAURADOS AOS VALORES    ;
;ORIGINAIS.                                                   ;
; OS CONTADORES SAO DISPARADOS APOS TEREM SIDOS CARREGADOS     ;
;-------------------------------------------------------------;

LOAD_C_A_G_2 PROC NEAR


    PUSH    AX                      ;SALVAR RESGISTROS
    PUSH    DX

    MOV     DX,TIMER_A_2            ;CARREGAR DX COM O
                                    ;ENDERECO DO CONTADOR 2
                                    ;DO TIMER A
```

```
    OUT     DX,AL                   ;CARREGAR CONTADOR 2 DO
    MOV     AL,AH                   ;TIMER A.
    OUT     DX,AL

    MOV     DX,TIMER_B_0            ;CARREGAR DX COM O
                                    ;ENDERECO DO CONTADOR
    MOV     AX,CX                   ; O DO TIMER B.


    OUT     DX,AL                   ;CARREGAR CONTADOR 2 DO
    MOV     AL,AH                   ;TIMER B.
    OUT     DX,AL



    POP     DX                      ;RECUPERAR REGISTROS.
    POP     AX

    RET                             ;RETORNAR DE LOAD_COUNTER.


    LOAD_C_A_G_2 ENDP
;-------------------------------------------------------------;
; LOAD_C_A_G_3                                                ;
; ESTA PROCEDURE CARREGA OS CONTADORES 1 E 2 DO TIMER B       ;
; ISTO E FEITO DA SEGUINTE MANEIRA:                           ;
;          * REGISTRO AX:VALOR PARA CONTADOR 1  DO TIMER B    ;
;          * REGISTRO CX:VALOR PARA CONTADOR 2  DO TIMER B    ;
; TODOS OS REGISTROS AFETADOS SERAO RESTAURADOS AOS VALORES    ;
;ORIGINAIS.                                                   ;
; OS CONTADORES SAO DISPARADOS APOS TEREM SIDOS CARREGADOS    ;
;-------------------------------------------------------------;

LOAD_C_A_G_3 PROC NEAR


    PUSH    AX                      ;SALVAR RESGISTROS
```

```
    PUSH    DX

    MOV     DX,TIMER_B_1            ;CARREGAR DX COM O
                                    ;ENDERECO DO CONTADOR 1
                                    ;1 DO TIMER B.
    OUT     DX,AL                   ;CARREGAR CONTADOR 1
    MOV     AL,AH                   ;DO TIMER B
    OUT     DX,AL

    MOV     DX,TIMER_B_2            ;CARREGAR DX COM O
                                    ;ENDERECO DO CONTADOR 2
    MOV     AX,CX                   ;2 DO TIMER B


    OUT     DX,AL                   ;CARREGAR CONTADOR 2 DO
    MOV     AL,AH                   ;TIMER B
    OUT     DX,AL



    POP     DX                      ;RECUPERAR REGISTROS.
    POP     AX


    RET                             ;RETORNAR DE LOAD_COUNTER.


    LOAD_C_A_G_3 ENDP


;-------------------------------------------------------------;
; LOAD_C_A_G_4                                                ;
; ESTA PROCEDURE CARREGA O CONTADOR O TIMER C.ISTO E FEITO DA  ;
; SEGUINTE MANEIRA:                                           ;
;          * REGISTRO AX:VALOR PARA CONTADOR O  DO TIMER C     ;
; TODOS OS REGISTROS AFETADOS SERAO RESTAURADOS AOS VALORES    ;
;ORIGINAIS                                                    ;
```

```
;-------------------------------------------------------------;

LOAD_C_A_G_4 PROC NEAR


    PUSH    AX                      ;SALVAR RESGISTROS
    PUSH    DX

    MOV     DX,TIMER_C_0            ;CARREGAR DX COM O
                                    ;ENDERECO DO CONTADOR
                                    ;0 DO TIMER C
    OUT     DX,AL                   ;CARREGAR CONTADOR 0 DO
    MOV     AL,AH                   ;TIMER C
    OUT     DX,AL



    POP     DX                      ;RECUPERAR REGISTROS.
    POP     AX


    RET                             ;RETORNAR DE LOAD_COUNTER.


    LOAD_C_A_G_4 ENDP

;-------------------------------------------------------------;
; LOAD_C_A_G_5                                                ;
; ESTA PROCEDURE CARREGA O CONTADOR 1 TIMER C.ISTO E FEITO DA ;
; SEGUINTE MANEIRA:                                           ;
;          * REGISTRO AX:VALOR PARA CONTADOR 1  DO TIMER C    ;
; TODOS OS REGISTROS AFETADOS SERAO RESTAURADOS AOS VALORES    ;
;ORIGINAIS.                                                   ;
;-------------------------------------------------------------;

LOAD_C_A_G_5 PROC NEAR
```

```
    PUSH    AX                      ;SALVAR RESGISTROS
    PUSH    DX

    MOV     DX,TIMER_C_1            ;CARREGAR DX COM O
                                    ;ENDERECO DO CONTADOR
                                    ;1 DO TIMER C.
    OUT     DX,AL                   ;CARREGAR CONTADOR O DO
    MOV     AL,AH                   ;TIMER C.
    OUT     DX,AL



    POP     DX                      ;RECUPERAR REGISTROS.
    POP     AX

    RET                             ;RETORNAR DE LOAD_COUNTER.

    LOAD_C_A_G_5 ENDP

;-------------------------------------------------------------;
; LOAD_C_A_G_6                                                ;
; ESTA PROCEDURE CARREGA O CONTADOR 2 TIMER C.ISTO E FEITO DA ;
; SEGUINTE MANEIRA:                                           ;
;  * REGISTRO AX:VALOR PARA CONTADOR 2  DO TIMER C            ;
; TODOS OS REGISTROS AFETADOS SERAO RESTAURADOS AOS VALORES   ;
; ORIGINAIS.                                                  ;
;-------------------------------------------------------------;

LOAD_C_A_G_6 PROC NEAR

    PUSH    AX                      ;SALVAR RESGISTROS
    PUSH    DX

    MOV     DX,TIMER_C_2            ;CARREGAR DX COM O
```

```
                                        ;ENDERECO DO CONTADOR
                                        ;2 DO TIMER C.
        OUT     DX,AL                   ;CARREGAR CONTADOR 0 DO
        MOV     AL,AH                   ;TIMER C .
        OUT     DX,AL



        POP     DX                      ;RECUPERAR REGISTROS.
        POP     AX


        RET                             ;RETORNAR DE LOAD_COUNTER.


        LOAD_C_A_G_6 ENDP
```

```
;-------------------------------------------------------------;
; A PRODEDURE SET_COUNTER PROGRAMA OS TIMERS DA PLACA DE INTERFA;
; CES.ESTES TIMERS POSSUI OS ENDERECOS:                       ;
;             * CONTADOR 0 DO TIMER A:100H                    ;
;             * CONTADOR 1 DO TIMER A:101H                    ;
;             * CONTADOR 2 DO TIMER A:102H                    ;
;             * PALAVRA DE CONTROLE DO TIMER A:103H           ;
;             * CONTADOR 0 DO TIMER B:104H                    ;
;             * CONTADOR 1 DO TIMER B:105H                    ;
;             * CONTADOR 2 DO TIMER B:106H                    ;
;             * PALAVRA DE CONTROLE DO TIMER B:107H           ;
;             * CONTADOR 0 DO TIMER C:108H                    ;
;             * CONTADOR 1 DO TIMER C:109H                    ;
;             * CONTADOR 2 DO TIMER C:10AH                    ;
;             * PALAVRA DE CONTROLE DO TIMER C:10BH           ;
; ESTES ENDERECOS PODERAO SER ALTERADOS NA CHAVE DIP DA PLACA. ;
; NA UTILIZACAO DESTA PROCEDURE NENHUM PARAMETRO E NECESSARIO  ;
```

```
; POIS OS MODOS DE PROGRAMACAO JA ESTAO DEFINIDOS COMO SEGUE:    ;
;                                                               ;
;            * CONTADOR 0 DO TIMER A:MODO 1                     ;
;            * CONTADOR 1 DO TIMER A:MODO 1                     ;
;            * CONTADOR 2 DO TIMER A:MODO 1                     ;
;            * CONTADOR 0 DO TIMER B:MODO 1                     ;
;            * CONTADOR 1 DO TIMER B:MODO 1                     ;
;            * CONTADOR 2 DO TIMER B:MODO 1                     ;
;            * CONTADOR 0 DO TIMER C:MODO 1                     ;
;            * CONTADOR 1 DO TIMER C:MODO 1                     ;
;            * CONTADOR 2 DO TIMER C:MODO 1                     ;
;                                                               ;
;---------------------------------------------------------------;

SET_COUNTER PROC NEAR


    PUSH DX                         ;SALVAR REGISTROS USADOS
    PUSH AX                         ;DENTRO DA PROCEDURE.

    MOV     DX,CONTROL_T_A          ;CARREGAR DX COM O ENDERE-
                                    ;CO DA PALAVRA
                                    ;DE CONTROLE DO TIMER A.

    MOV     AL,MODO_1_A_0           ;CARREGA AL COM A PALAVRA
                                    ;DE CONTROLE PARA MODO 1
                                    ; 1 PARA O CONTADOR 0 DO
                                    ;DO TIMER A.

    OUT     DX,AL                   ;PROGRAMA O CONTADOR.

    MOV     AL,MODO_1_A_1           ;MODO 1 PARA CONTADOR 1 DO
                                    ;TIMER A.
    OUT     DX,AL                   ;PROGRAMA O CONTADOR.

    MOV     AL,MODO_1_A_2           ;MODO 1 PARA CONTADOR 2 DO
                                    ;TIMER A
    OUT     DX,AL                   ;PROGRAMA O CONTADOR
```

```
MOV     DX,CONTROL_T_B          ;CARREGAR DX COM O
                                ;ENDERECO DA PALAVRA
                                ;DE CONTROLE DO TIMER B.

MOV     AL,MODO_1_B_0           ;CARREGA AL COM A PALAVRA
                                ;DE CONTROLE PARA  MODO  1
                                ;PARA O CONTADOR 0 DO
                                ;DO TIMER B.

OUT     DX,AL                   ;PROGRAMA O CONTADOR.

MOV     AL,MODO_1_B_1           ;MODO 1 PARA CONTADOR 1 DO
                                ;TIMER B.
OUT     DX,AL                   ;PROGRAMA O CONTADOR.

MOV     AL,MODO_1_B_2           ;MODO 1 PARA CONTADOR 2
                                ;TIMER B.
OUT     DX,AL                   ;PROGRAMA O CONTADOR

MOV     DX,CONTROL_T_C

MOV     AL,MODO_1_C_0           ;MODO 1 PARA OS CONTADORES
                                ;1, 2, 3 .
OUT     DX,AL                   ;DO TIMER C.
MOV     AL,MODO_1_C_1

OUT     DX,AL
MOV     AL,MODO_1_C_2

OUT     DX,AL



POP     AX                      ;RECUPERA OS REGISTROS
POP     DX
```

```
    RET                                 ;RETORNA DA PROCEDURE

SET_COUNTER   ENDP


;-------------------------------------------------------------;
;ESTA SUB-ROTINA TEM COMO FUNCAO INICIALIZAR AS VARIAVEIS      ;
;UTILIZADAS.                                                  ;
;-------------------------------------------------------------;


INICIO PROC NEAR


     MOV   TIMER_A_0,200H
     MOV   TIMER_A_1,201H
     MOV   TIMER_A_2,202H
     MOV   CONTROL_T_A,203H
     MOV   MODO_1_A_0,32H
     MOV   MODO_1_A_1,72H
     MOV   MODO_1_A_2,0B2H
     MOV   TIMER_B_0,204H
     MOV   TIMER_B_1,205H
     MOV   TIMER_B_2,206H
     MOV   CONTROL_T_B,207H
     MOV   MODO_1_B_0,32H
     MOV   MODO_1_B_1,72H
     MOV   MODO_1_B_2,0B2H
     MOV   TIMER_C_0,208H
     MOV   TIMER_C_1,209H
     MOV   TIMER_C_2,20AH
     MOV   CONTROL_T_C,20BH
     MOV   MODO_1_C_0,32H
     MOV   MODO_1_C_1,72H
     MOV   MODO_1_C_2,0B2H
     MOV   BUSY,00H
     MOV   PORTA,20CH
     MOV   ESTADO,00H
```

```
     RET

INICIO  ENDP



CODE ENDS


DATA SEGMENT WORD PUBLIC
EXTRN   LARG:WORD,NP2:WORD
TIMER_A_0    DW  ?
TIMER_A_1    DW  ?
TIMER_A_2    DW  ?
CONTROL_T_A  DW  ?
MODO_1_A_0   DB  ?
MODO_1_A_1   DB  ?
MODO_1_A_2   DB  ?
TIMER_B_0    DW  ?
TIMER_B_1    DW  ?
TIMER_B_2    DW  ?
CONTROL_T_B  DW  ?
MODO_1_B_0   DB  ?
MODO_1_B_1   DB  ?
MODO_1_B_2   DB  ?
TIMER_C_0    DW  ?
TIMER_C_1    DW  ?
TIMER_C_2    DW  ?
CONTROL_T_C  DW  ?
MODO_1_C_0   DB  ?
MODO_1_C_1   DB  ?
MODO_1_C_2   DB  ?
OLD_MASK     DB  ?
BUSY         DB  ?
PORTA        DW  ?
ESTADO       DB  ?
```

```
POINTER1    DW  ?
POINTER2    DW  ?
POINTER3    DW  ?

DATA ENDS



END
```

APÊNDICE III

C.1 PROGRAMA DISP PARA CHAMADA DA ROTINA DE MODULACAO

```
{$L DISPAROA}

PROGRAM DISPARO;

USES PRINTER,GRAPH,CRT;


VAR    F,M : REAL;
       TEST : BOOLEAN;
       KEY2,KEY3:CHAR;
       LARGURA:ARRAY[0..43] OF REAL;
       NO,NP2,NP,P,I:INTEGER;
       LARG:ARRAY[0..47] OF INTEGER;

PROCEDURE FIRE;EXTERNAL;


PROCEDURE INTEIRO;

VAR I:INTEGER;


BEGIN
   NP2:=2*(NP+4);
   LARG[0]:=ROUND(1/F/3*1000000);
   LARG[1]:=LARG[0];
   LARG[2]:=LARG[0];
   LARG[3]:=LARG[0];
   FOR I:=4 TO 47 DO   LARG[I]:=ROUND(LARGURA[I-4]/PI/F*1000000);
END;


PROCEDURE  COMPLETO;


BEGIN
    NP:=2;
    LARGURA[0]:=PI;
    LARGURA[1]:=PI;
END;


PROCEDURE FLANCOS;

VAR  ALFA:REAL;
```

```
BEGIN
   NP:=6;
   IF (M=1) THEN ALFA:=PI/2 ELSE ALFA:=ARCTAN(M/SQRT(1-M*M));
   LARGURA[0]:=PI-2*ALFA;
   LARGURA[1]:=ALFA;
   LARGURA[2]:=ALFA;
   LARGURA[3]:=LARGURA[0];
   LARGURA[4]:=ALFA;
   LARGURA[5]:=ALFA;
END;

PROCEDURE MODOQUADRADO;


BEGIN
     CASE P OF

               3:BEGIN
                 NP:=6;
   LARGURA[0]:=(PI/6*(2+M));
   LARGURA[1]:=(PI/3*(1-M));
   LARGURA[2]:=LARGURA[0];
   LARGURA[3]:=LARGURA[0];
   LARGURA[4]:=LARGURA[1];
   LARGURA[5]:=LARGURA[0];
                 END;
        5:BEGIN
   NP:=10;
   LARGURA[0]:= (PI/3+PI/12*M);
   LARGURA[1]:= (PI/6*(1-M));
   LARGURA[2]:= (PI/6*M);
   LARGURA[3]:= LARGURA[1];
   LARGURA[4]:= LARGURA[0];
   LARGURA[5]:= LARGURA[0];
   LARGURA[6]:= LARGURA[1];
   LARGURA[7]:= LARGURA[2];
   LARGURA[8]:= LARGURA[1];
   LARGURA[9]:= LARGURA[0];
    END;
        7:BEGIN
   NP:=14;
           LARGURA[0]:=(PI/18*(6+M));
   LARGURA[1]:=(PI/9*(1-M));
   LARGURA[2]:=(PI/9*M);
   LARGURA[3]:=LARGURA[1];
   LARGURA[4]:=LARGURA[2];
   LARGURA[5]:=LARGURA[1];
   LARGURA[6]:=LARGURA[0];
   LARGURA[7]:=LARGURA[0];
   LARGURA[8]:=LARGURA[1];
   LARGURA[9]:=LARGURA[2];
   LARGURA[10]:=LARGURA[1];
   LARGURA[11]:=LARGURA[2];
   LARGURA[12]:=LARGURA[1];
   LARGURA[13]:=LARGURA[0];
```

```
   END;
 END;

END;

PROCEDURE IMPRIMEMODOQUA;


VAR     I:INTEGER;

BEGIN
WRITELN(LST,'MODO QUADRADO'); WRITELN(LST);
WRITELN(LST,ROUND(NP/2), '  VEZES RETANGULAR');WRITELN(LST);
WRITELN(LST,'FREQUENCIA:  ',F);WRITELN(LST);
WRITELN(LST,'INDICE DE MODULACAO:  ',M);WRITELN(LST);
WRITELN(LST,'NUMERO DE PULSOS:  ',NP);WRITELN(LST);
FOR I:=4 TO NP+3 DO
BEGIN
WRITELN(LST,I-3,'    ',ROUND(LARG[I]/2));
END;
READLN;
END;

PROCEDURE SENO;



VAR    CT,T,CL,K:REAL;
       NUM,I:INTEGER;


BEGIN
      NP:=P*2;
      NUM:=1;
      CT:=0;
      T:=2*PI/4/P;
      FOR I:= 1 TO ROUND(((P-1)/2))  DO
      BEGIN
            CL:=CT;
            CT:=0.5*T*(SIN((2*I-1)*T)+SIN((2*I+1)*T));
            K:=CL+CT;
            LARGURA[I-1]:=2*T+NUM*M*K;
            NUM:=NUM*(-1);
      END;
      LARGURA[ROUND((P-1)/2)]:=T*(1+NUM*M*SIN((2*I-1)*T))*2;
      NUM:=ROUND((P-1)/2-1);
      FOR I:= ROUND(((P-1)/2+1)) TO P-1 DO
      BEGIN
            LARGURA[I]:=LARGURA[NUM];
            NUM:=NUM-1;
      END;
      NUM:=P-1;
      FOR I:=P TO (2*P-1) DO
      BEGIN
```

```
            LARGURA[I]:=LARGURA[NUM];
            NUM:=NUM-1;
      END;
END;

PROCEDURE IMPRIMESENO;


VAR I:INTEGER;

BEGIN
      WRITELN(LST,'MODO SENOIDAL');WRITELN(LST);
      WRITELN(LST,'INDICE DE MODULACAO:  ',M);WRITELN(LST);
      WRITELN(LST,'FREQUENCIA DE SAIDA:  ',F);WRITELN(LST);
      WRITELN(LST,'NUMERO DE PULSOS:  ',NP);WRITELN(LST);

      FOR I:=4  TO NP+3 DO
      BEGIN
            WRITELN(LST,I-3,'    ',ROUND(LARG[I]/2));
      END;
READLN;
END;


BEGIN

CLRSCR;
FOR I:=0 TO 43 DO LARGURA[I]:=0.0;
TEST:= TRUE;
WHILE TEST DO
BEGIN
WRITELN;
WRITELN;
WRITELN;
WRITELN('          ESCOLHA A TECNICA PWM DESEJADA');
WRITELN;WRITELN;WRITELN;
WRITELN('          [s] Onda Senoidal');
WRITELN;
WRITELN('          [q] Onda Quadrada');
WRITELN;
WRITELN('          [f] Modulacao dos Flancos');
WRITELN;
WRITELN('          [b] Bloco Completo       ');
WRITELN;WRITELN;WRITELN;
WRITELN('Entre com a letra escolhida:   ');
REPEAT
KEY2 :=READKEY;
UNTIL (KEY2 = 's') OR (KEY2 = 'S')OR(KEY2 = 'Q') OR (KEY2 = 'q' )
  OR (KEY2 = 'F') OR (KEY2 = 'f')OR(KEY2 = 'B') OR (KEY2 = 'b' );
IF (KEY2 = 'S') OR (KEY2 = 's') THEN NO:=1;
IF (KEY2 = 'Q') OR (KEY2 = 'q') THEN NO:=2;
IF (KEY2 = 'F') OR (KEY2 = 'f') THEN NO:=3;
IF (KEY2 = 'B') OR (KEY2 = 'b') THEN NO:=4;
CLRSCR;
WRITELN;WRITELN;
```

```
CASE NO OF
  1:BEGIN
    WRITELN('                    TECNICA DE MODULACAO SENOIDAL ');
         WRITELN;WRITELN;
         WRITELN('ENTRE COM O INDICE DE MODULACAO');
         READLN( M );
         WRITELN;WRITELN;
         WRITELN('ENTRE COM O NUMERO DE PULSOS EM MEIO PERIODO');
         WRITELN('AS OPCOES SAO:');
         WRITELN('- 9 PULSOS');
         WRITELN('- 15 PULSOS');
         WRITELN('- 21 PULSOS');
         WRITELN;WRITELN;
         REPEAT
         READLN( P );
         UNTIL (P=9) OR (P=15) OR (P=21);
         WRITELN;WRITELN;
         WRITELN('ENTRE COM A FREQUENCIA DE SAIDA');
         READLN( F ) ;
         SENO;
     INTEIRO;
         FIRE;
   END;

  2:BEGIN
 WRITELN('               TECNICA DE MODULACAO COM ONDA QUADRADA');
     WRITELN;WRITELN;
         WRITELN('ENTRE COM O INDICE DE MODULACAO');
         READLN( M ) ;
         REPEAT
             WRITELN;WRITELN;
             WRITELN('ENTRE COM O NUMERO DE PULSOS POR PERIODO');
             WRITELN('AS OPCOES SAO:');
             WRITELN('- 3 PULSOS');
             WRITELN('- 5 PULSOS');
             WRITELN('- 7 PULSOS');
         READLN(P);
         UNTIL (P=3) OR (P=5) OR (P=7);
         WRITELN;WRITELN;
         WRITELN('ENTRE COM A FREQUENCIA DE SAIDA');
         READLN ( F );
     MODOQUADRADO;
     INTEIRO;
     FIRE;
    END;

  3:BEGIN
      WRITELN('      MODULACAO DOS FLANCOS');WRITELN;
      WRITELN('      ENTRE COM O INDICE DE MODULACAO');
      READLN(M);WRITELN;
      WRITELN('    ENTRE COM A FREQUENCIA DE SAIDA');
      READLN(F);
      FLANCOS;
      INTEIRO;
```

```
      FIRE
   END;
  4:BEGIN
      WRITELN('     BLOCO COMPLETO'); WRITELN;
      WRITELN('     ENTRE COM A FREQUENCIA DE SAIDA');
      READLN(F);
      COMPLETO;
      INTEIRO;
      FIRE;
   END;
WRITELN('     DESEJA EXECUTAR OUTRO CASO? S/N');
KEY2 := READKEY;
IF (KEY2 = 'S' OR KEY2 = 's') THEN TEST := FALSE;
END;

END.
```

APÊNDICE IV

TÉCNICAS DE MODULAÇÃO PWM

A4.1 - Introdução

Existem diversos métodos que possibilitam o controle da velocidade de um MIT. Dentre estes métodos, um dos mais importantes, atualmente, é a técnica de modulação PWM. Dentro desta técnica, existem diversos tipos de formação da onda PWM visando melhorar uma ou outra característica da onda resultante.

Este capítulo mostra como é gerada uma onda PWM e os tipos de onda utilizadas no acionamento do MIT com a respectiva análise espectral de cada uma. Para tanto, desenvolveu-se um programa com as leis de formação PWM em conjunto com a transformada rápida de Fourier .

A4.2 - Formação da onda PWM

A formação da onda PWM é obtida através de um circuito eletrônico (controlador PWM) que tem como finalidade gerar os sinais de gatilho (comando) do inversor. Os pricinpais sinais gerados no controlador PWM, mostrados na figura A4.1, são: uma onda triangular, que é a portadora de Modulação PWM; um sistema trifásico de tensões alternadas, que serve de referência; e os sinais de gatilhamento (G_1, G_2, G_3, G_4, G_5 e G_6) das chaves do inversor.

Para gerar os sinais de gatilhamento G_1 e G_4, compara-se a onda triangular com a tensão (referência) da fase A, seguindo o seguinte critério:

- Quando a portadora triangular é menor que a referência senoidal, fase A (de ωt_1 a ωt_2), é liberado um pulso em G_1 (nível alto) e G_4 permanece em nível baixo.

- Quando a portadora triangular é maior que a referência senoidal, a fase A (de ωt_2 a ωt_3), é liberado um pulso

em G_4 (nível alto) e G_1 é levado para nível baixo.

Observando-se a figura A4.1, nota-se que, nos instantes de intersecção da portadora triangular e referência senoidal (fase A), existe uma comutação entre as chaves 1 e 4

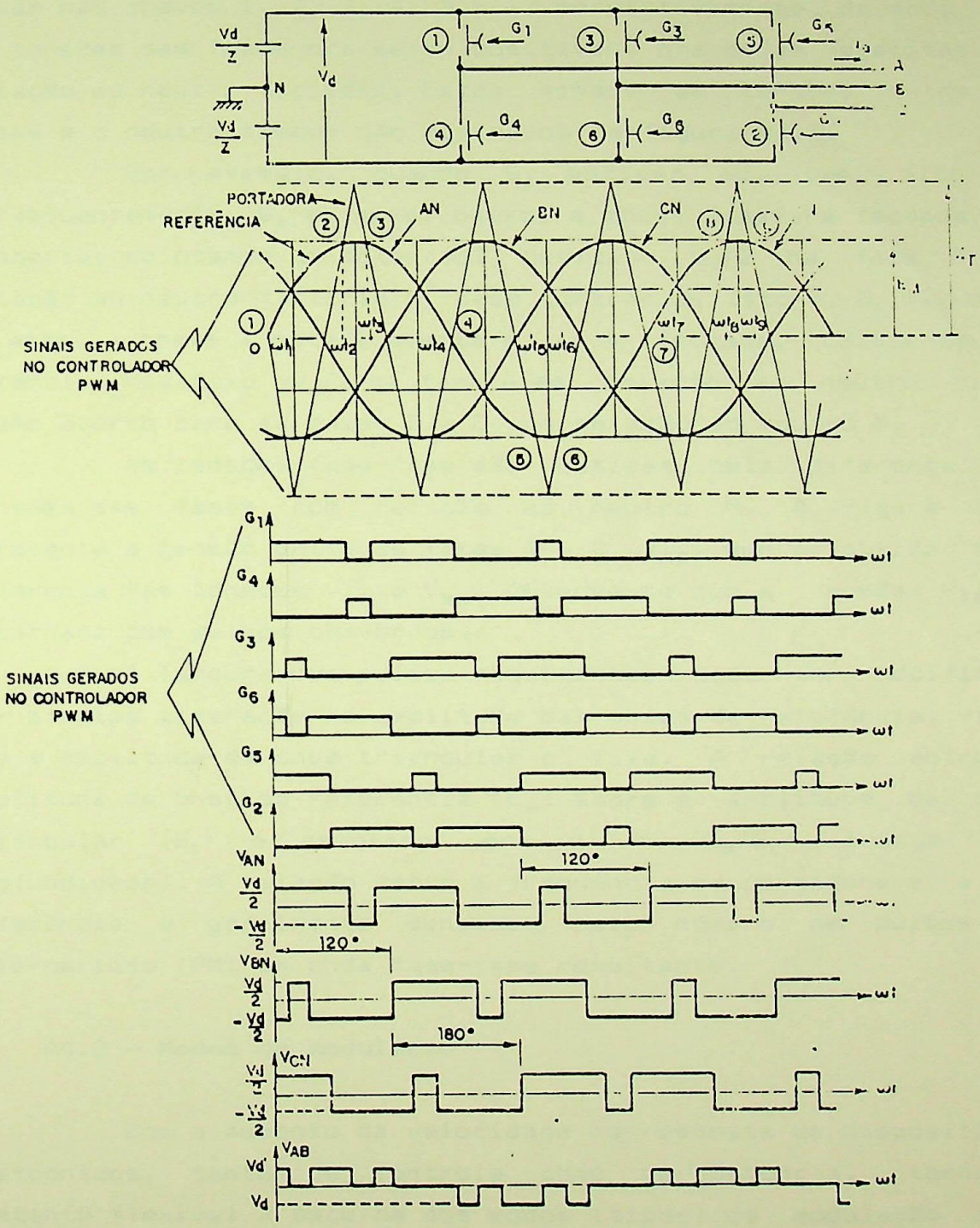


Fig. A4.1 - Formação das ondas PWM

Analogamente, pode-se analisar os sinais G_3 e G_6 pela comparacão da portadora triangular com a referência senoidal da fase B, e os sinais G_5 e G_2 pela comparação com a referência senoidal da fase C.

Os sinais de gatilhamento G_1, G_2, G_3, G_4, G_5 e G_6 irão atuar nas chaves 1, 2, 3, 4, 5 e 6, respectivamente, de modo que as tensões das fases ora sejam positivas, ora sejam negativas em relação ao neutro fictício. Estes sinais de tensões entre as fases e o neutro também são mostrados na figura A4.1.

Por exemplo, quando G_1 estiver em nível alto e, consequentemente, G_4 em nível baixo, a chave 1 estará fechada e a 4 aberta, colocando um potencial positivo $V_d/2$ na fase A em relação ao neutro fictício N. Caso contrário, isto é, G_1 baixo e G_4 alto, a chave 1 estará aberta e a 4 fechada, colocando um potêncial negativo $-V_d/2$ na fase A em relação ao neutro N. O mesmo ocorre para as fases B e C com relação ao neutro N.

As tensões fase-fase são obtidas pela diferença das tensões das fases com relação ao neutro N. A figura A4.1 apresenta a tensão entre as fases A e B, V_{AB}, que é obtida pela diferença das tensões V_{AN} e V_{BN}. Observa-se que a tensão V_{AB} é alternada com pulsos chaveados.

A largura dos pulsos resultantes pode ser modificada por simples alteração na amplitude das ondas de referência, visto que a amplitude da onda triangular é fixa. A relação entre a amplitude da onda de referência (E_m) sobre a amplitude da onda triangular (E_t) é denotada por $m = E_m/E_t$ (índice de profundidade). A relação entre a frequência da portadora e a da referência é geralmente denotada pelo número de pulsos em meio-período (PM) da onda fase-fase resultante.

A4.3 - Modos de modulaçao

Com o aumento da velocidade de resposta de dispositivos eletrônicos, tanto de controle como de potência, torna-se bastante flexível a escolha dos modos (tipos) de modulação para atender as características do conjunto motor/carga.

Sabe-se que o conjugado produzido por um motor é

resultado da corrente, que por sua vez é resultado entre a tensão aplicada e a força eletromotriz (f.e.m.) induzida. Portanto, quando um motor é alimentado por tensões do tipo PWM, as correntes resultantes possuem um grande conteúdo harmônico e estes harmônicos produzirão como efeito uma pulsação no conjugado do motor. O mais adequado então seria alimentar a máquina com uma forma de onda com PM grande (fazendo com que os harmônicos gerados sejam de alta ordem), pois quanto maior PM menor será o conteúdo harmônico das correntes. Isto não é possível em toda faixa de operação da máquina devido ao tempo crítico de comutação entre as chaves, porque à medida em que se aumenta a frequência, também se aumenta a largura dos pulsos para responder a uma determinada característica V/f. Para resolver este problema, fazem-se mudanças no modo de modulação, de maneira que a máquina tenha boa resposta em toda faixa de operação.

A tabela A4.1 mostra os modos de modulação com as suas respectivas faixas de trabalho. Deve-se observar que estes modos são compatíveis com os novos equipamentos microprocessados existentes no mercado .

Tabela A4.1

modos	frequência [Hz]	tipo de modulação
1	3 - 18	senoidal PM = 45
2	18 - 30	senoidal PM = 21
3	30 - 38	senoidal PM = 15
4	38 - 44	senoidal PM = 9
5	44 - 48	trapezoidal PM = 7
6	48 - 52	trapezoidal PM = 5
7	52 - 56	trapezoidal PM = 3
8	56 - 60	modulação dos flancos
9	> 60	blocos completos

Os modos descritos na tabela A4.1 com os respectivos

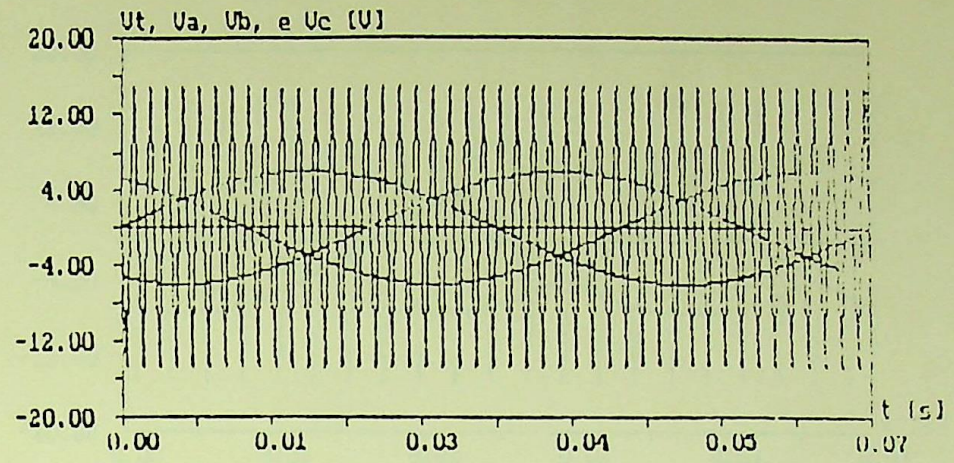
conteúdos harmônicos são mostrados nas figuras A4.2 a A4.10. As amplitudes das componentes harmônicas mostradas foram calculadas em porcentagem da amplitude da componente fundamental da onda de blocos completos (U_d).A portadora está sícronizada com as três fases de referência apenas quando PM é múltiplo de 3. Por este motivo observa-se nas figuras A4.6 e A4.7, que as portadoras foram modificadas, justamente para que estejam sincronizadas com as três fases de referência.

A figura A4.9 mostra a modulação dos flancos, que é formada através da comparação das retas de modulação $+E_f$ e $-E_f$ e a origem. Este modo é necessário para evitar o degrau entre o modo 7 e o modo 9 (blocos completos), na curva V/f.

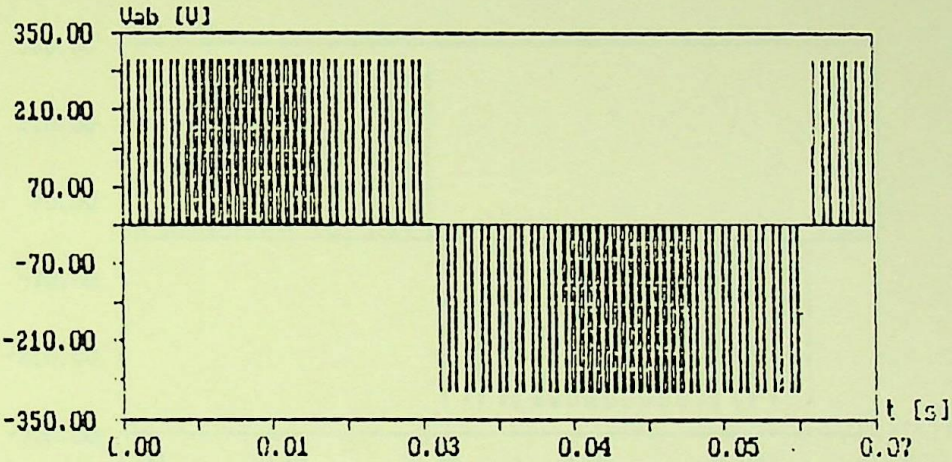
Observando-se as figuras A4.2 a A4.10, nota-se que: quanto maior PM, maiores serão as ordens das harmônicas com maior amplitude; e quanto maior m, maior será a relação entre a amplitude da fundamental e a amplitude da harmônicas. Com a modulação PWM consegue-se fazer a componente fundamental variar proporcionalmente a m, inclusive mantendo relações do tipo $(V/f)^k$, onde k é uma constante que depende do tipo da carga (constante, proporcional a velocidade, proporcional a velocidade ao quadrado).

Existem, além desses tipos de modulação, outros tipos de formação da onda PWM que podem ser implementadas em microcomputadores com algumas vantagens ou desvantagens sobre os modos descritos acima que, para serem implementados requerem uma elevada capacidade de processamento, o que iria encarecer sobremodo o custo de um equipamento deste tipo.

a)



b)



c)

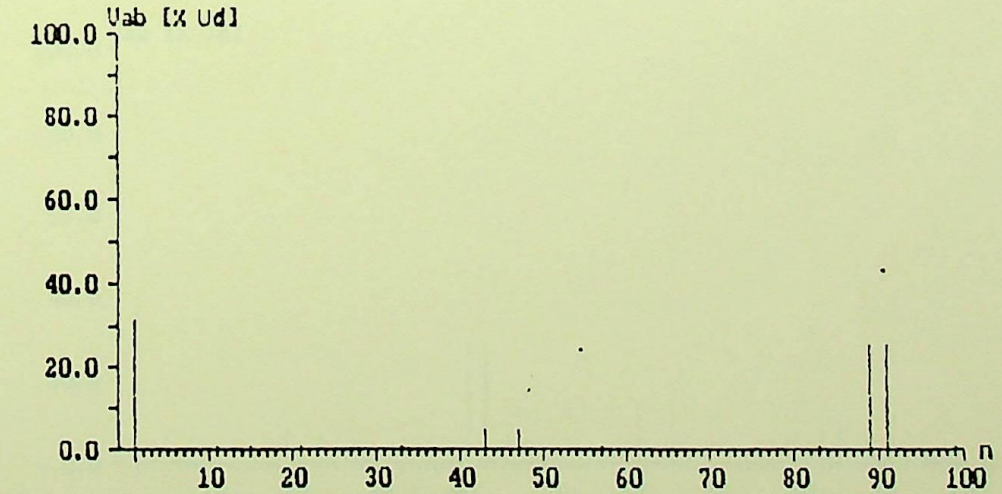
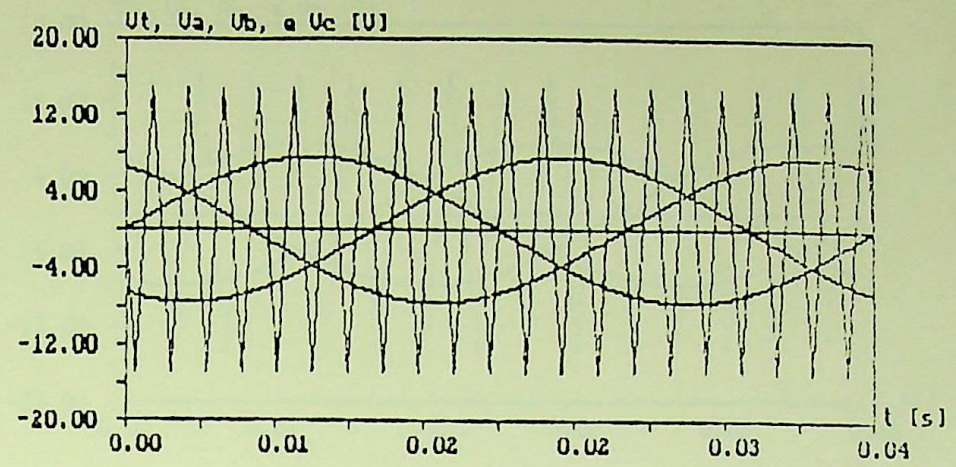
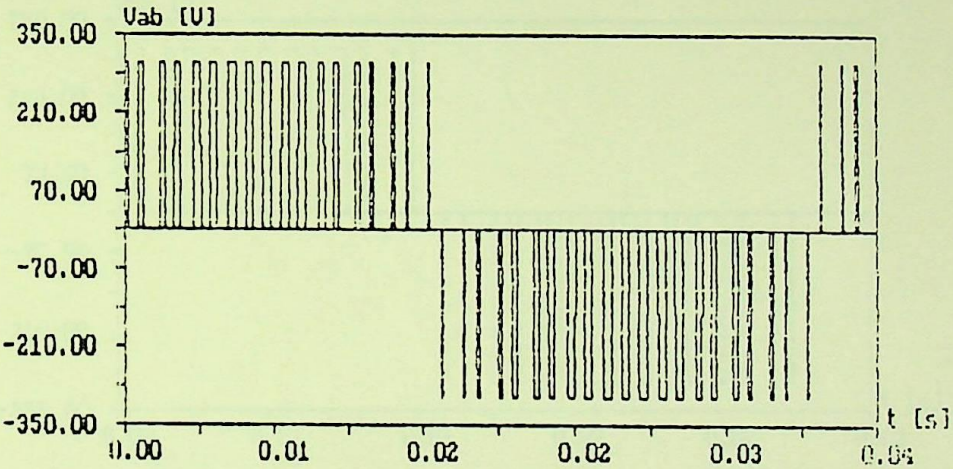


Fig A4.2 - Modo 1 (m = 0.4)

a)



b)



c)

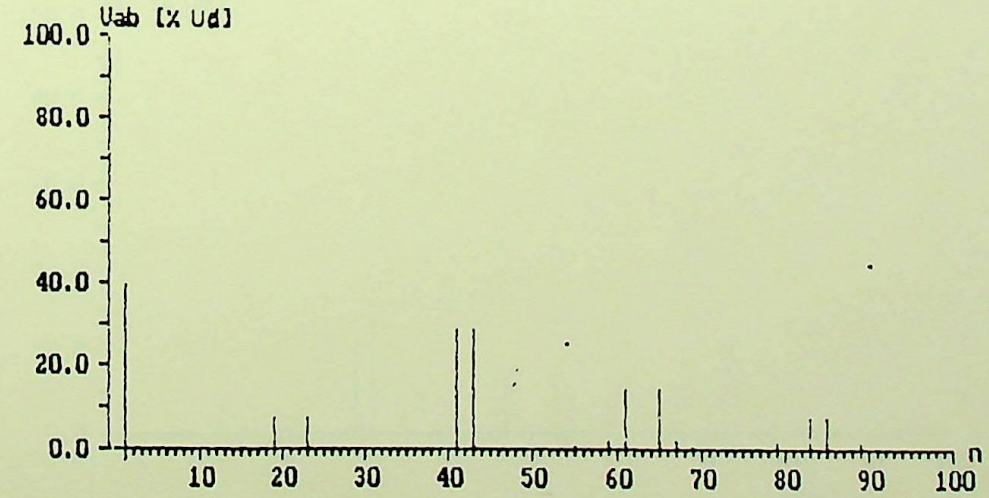
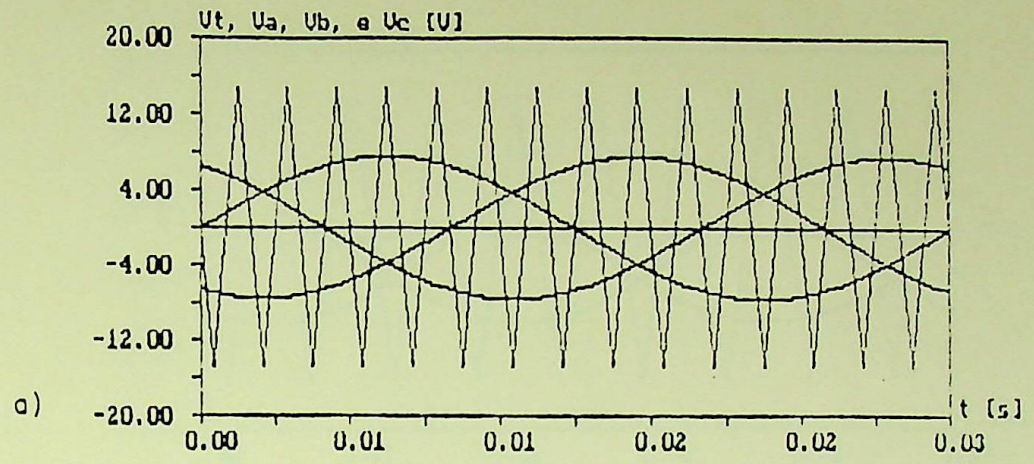
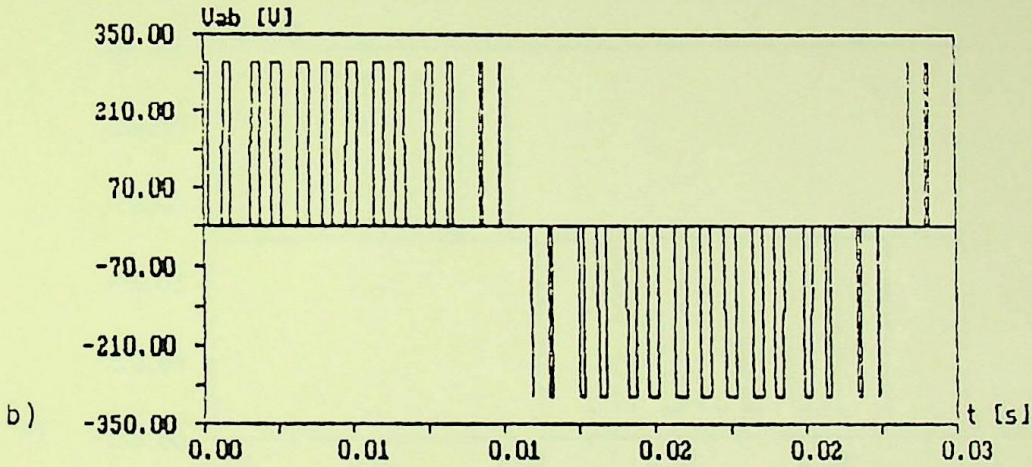


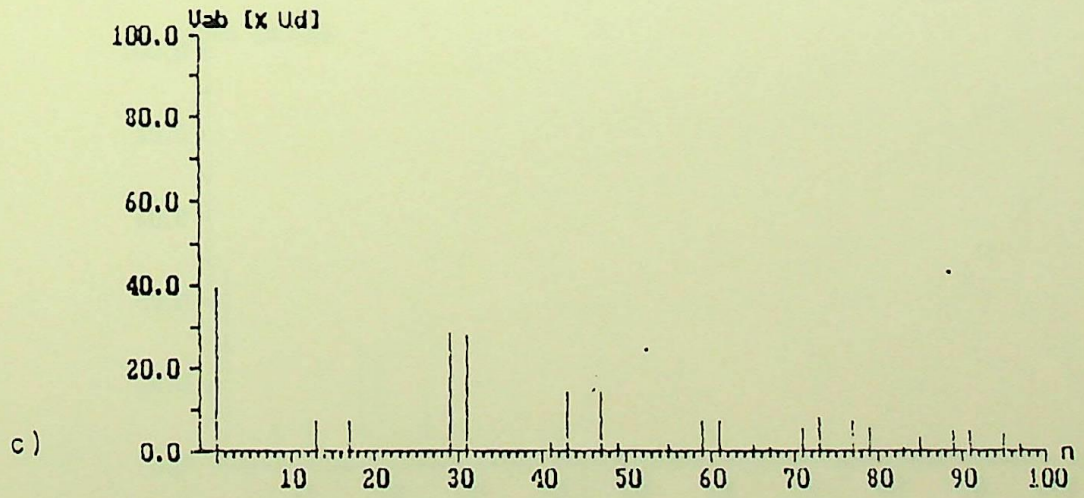
Fig. A4.3 - Modo 2 (m = 0.5)



a)



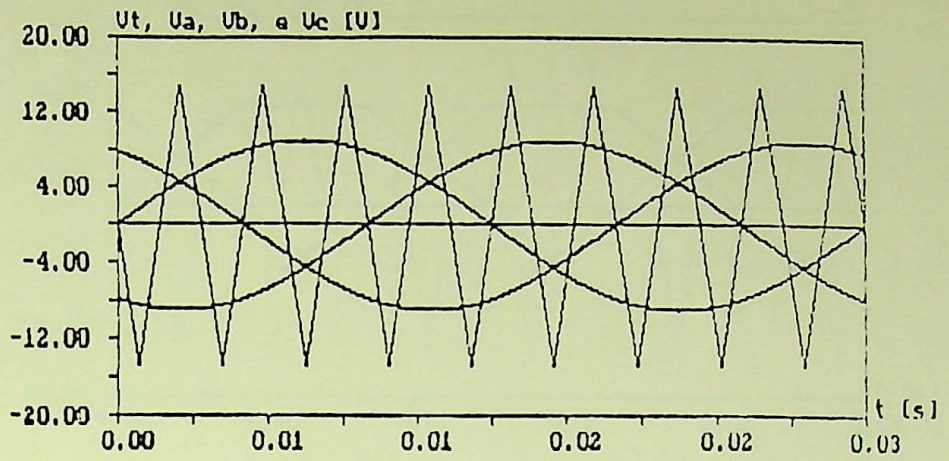
b)



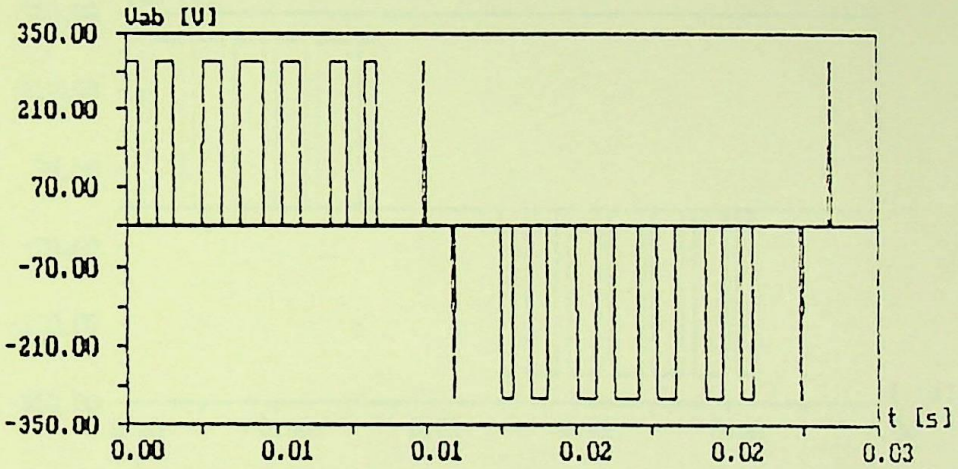
c)

Fig. A4.4 - Modo 3 (m = 0.5)

a)



b)



c)

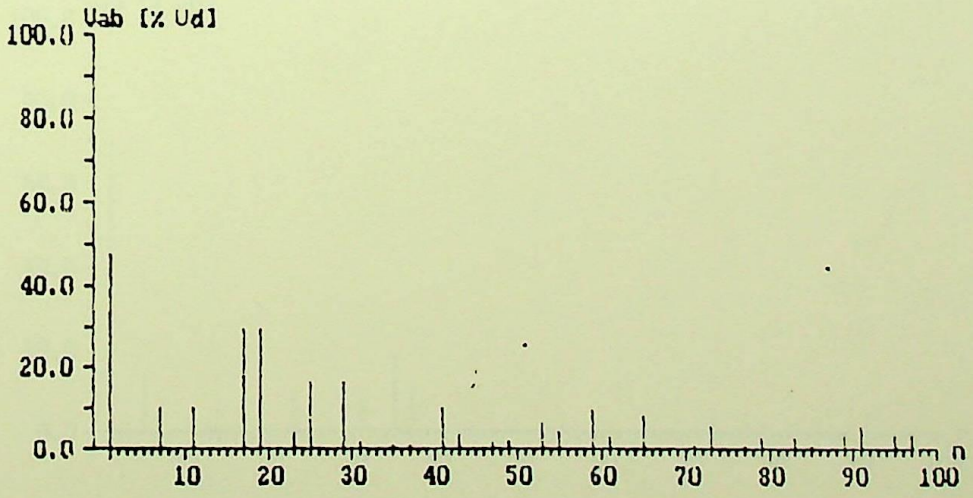
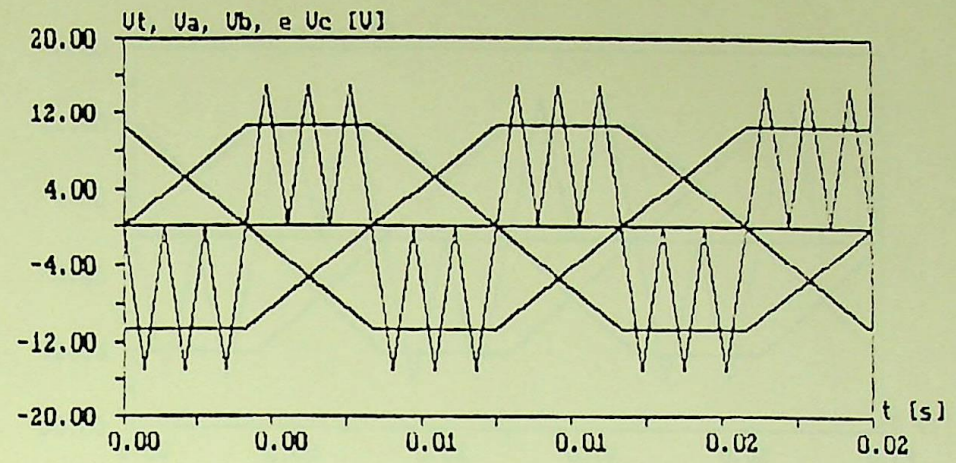
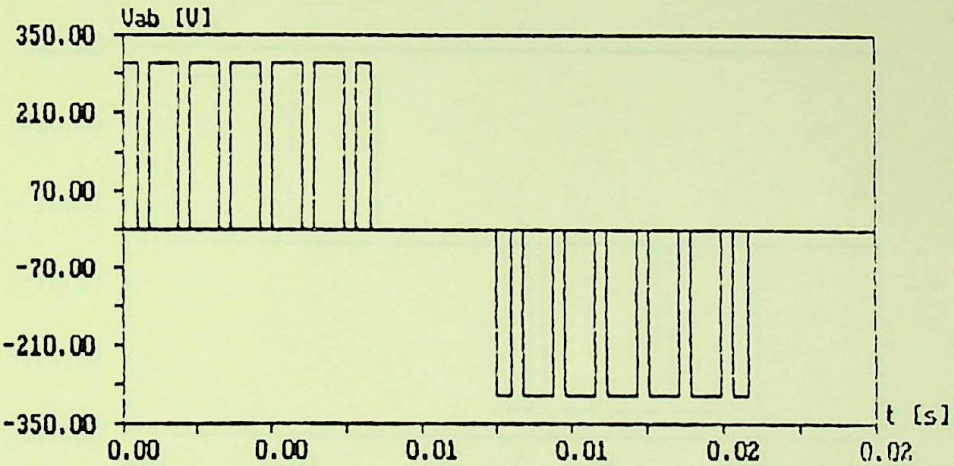


Fig. A4.5 - Modo 4 (m = 0.6)

a)



b)



c)

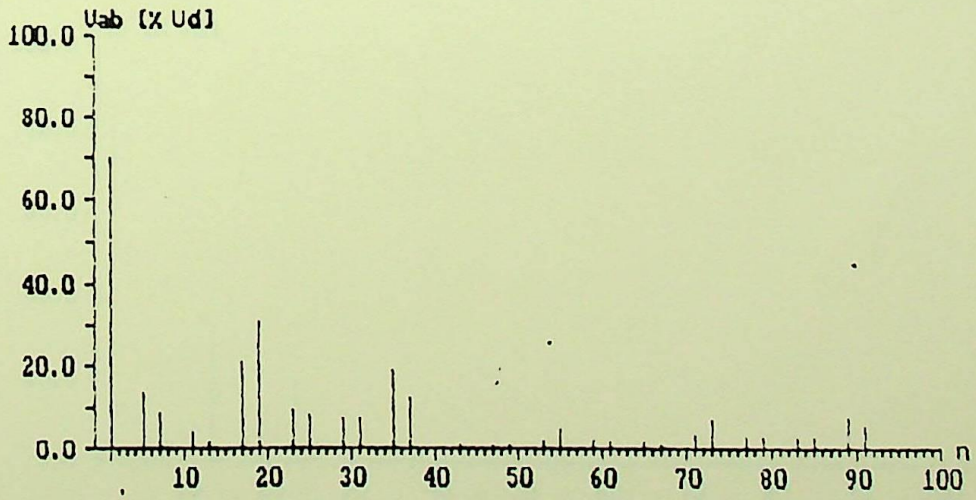
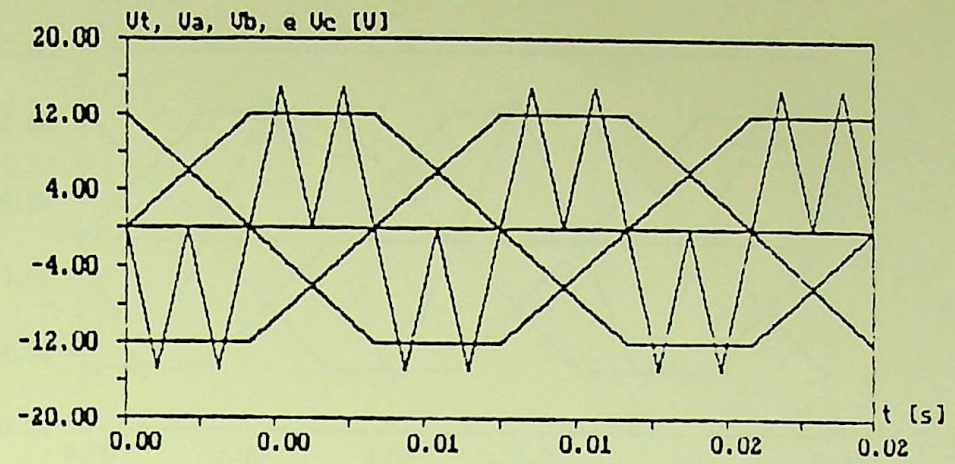
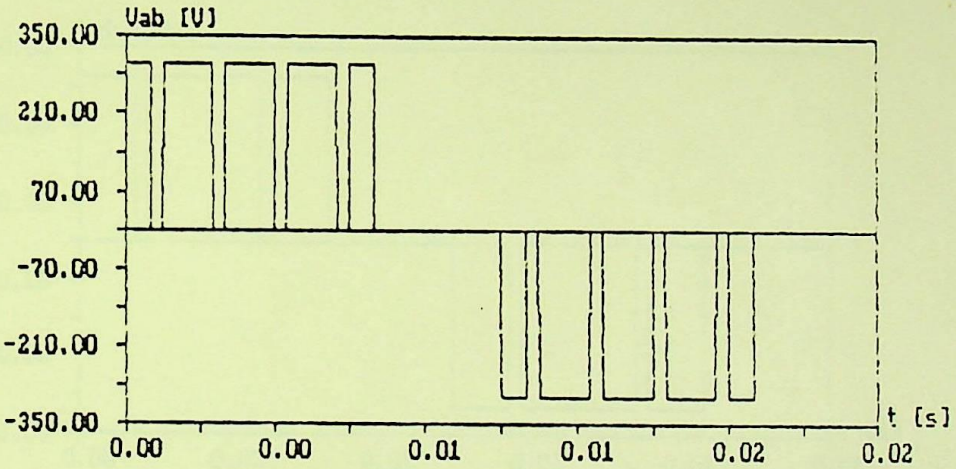


Fig. A4.6 - Modo 5 (m = 0.7)

a)



b)



c)

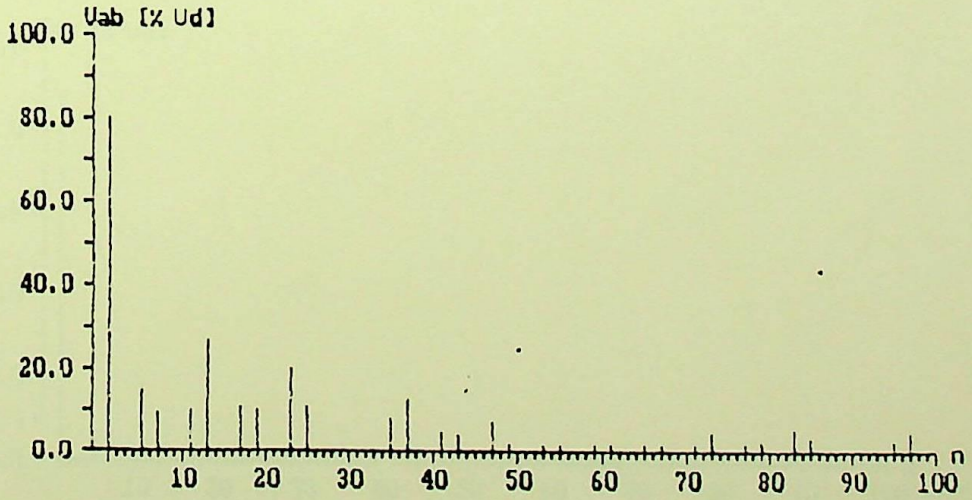
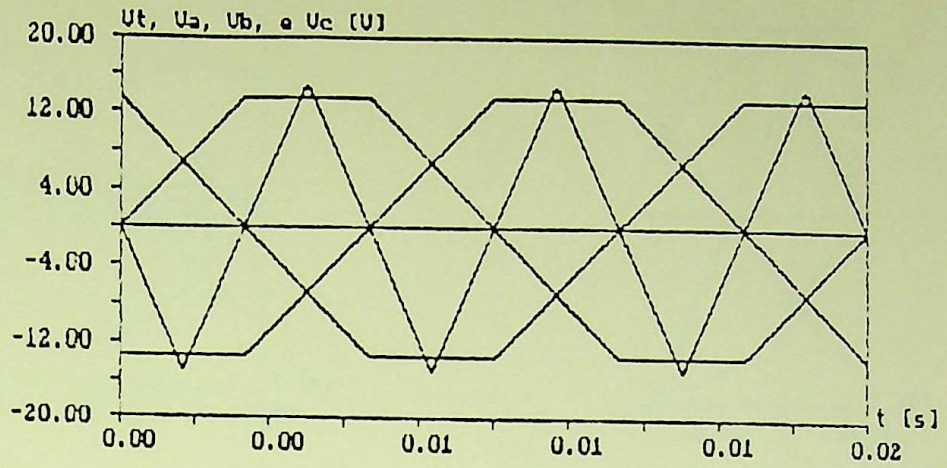
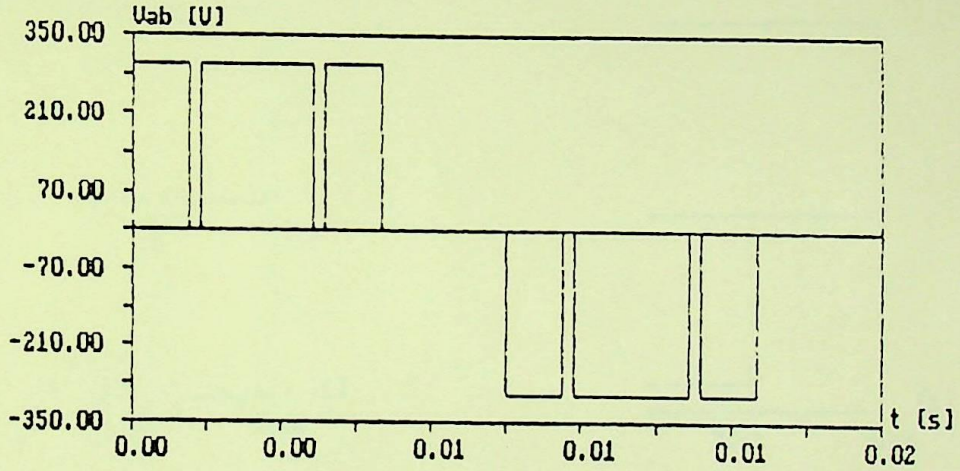


Fig. A4.7 - Modo 6 (m = 0.8)

a)



b)



c)

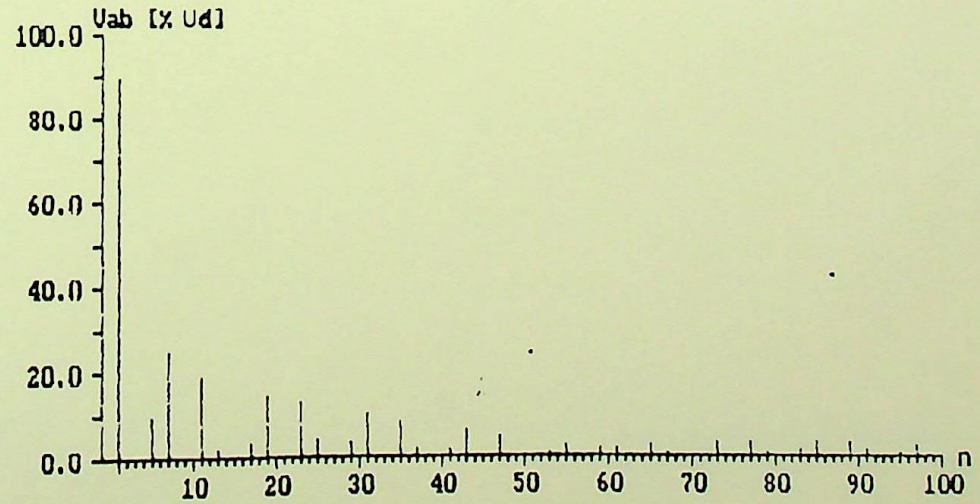
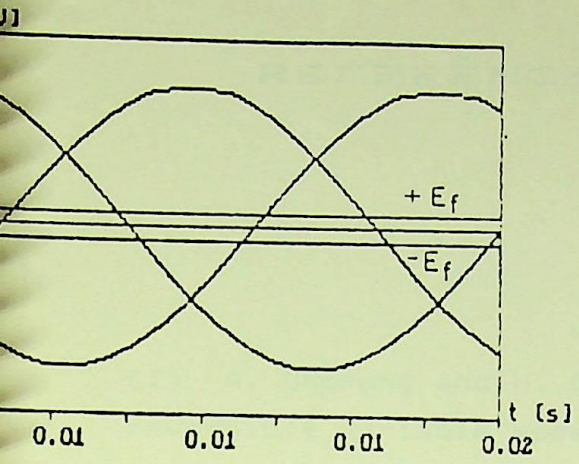
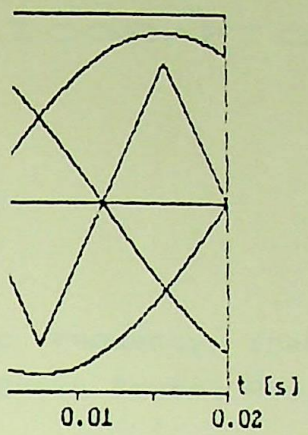
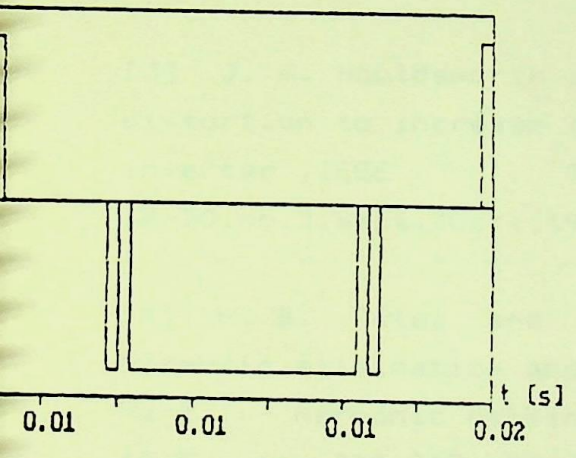
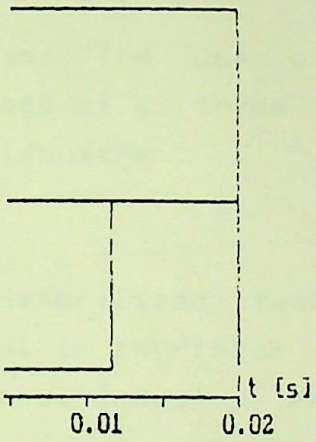


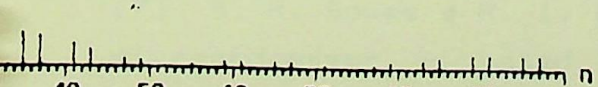
Fig. A4.8 - Modo 7 (m = 0.9)


+ Ef
-Ef
t [s]
0.01
0.01
0.01
0.02

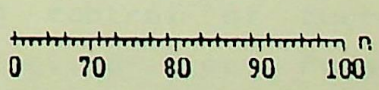

t [s]
0.01
0.02


t [s]
0.01
0.01
0.01
0.02


t [s]
0.01
0.02


n
40
50
60
70
80
90
100

do 8 (m = 0.9)


n
0
70
80
90
100

1.2)

REFERÊNCIAS BIBLIOGRÁFICAS

[1] A. Shonung and H. Stemmler,"Static frequency changers with reversible variable speed ac drives", Brown Boveri Rev.,Aug./Set. 1964.

[2] T. Onishi and H. Okitsu,"A novel PWM techniques for three phase inverter/converter",in Conf. Rec. 1983 IPEC (Tokyo Japão).

[3] J. A. Houldsworth and D. A. Grant,"The use of harmonic distortion to increase the output voltage of a three phase PWM inverter",IEEE Trans. Industry Appl.,vol. IA-20,no.5,Sept./Oct.,1982

[4] H. S. Patel and R. G. Hoft,"Generalized techniques of harmonic elimination and voltage control in thyristor inverters: Part I - Harmonic elimination" IEEE Trans. Industry Appl., vol. IA-9 , pp. 310-317, May/june 1973 .

[5] P. N. Enjeti, P. D. Ziogas, and J. F. Lindsay, "Programmed PWM techniques to eliminate harmonics a critical evaluation" , IEE IAS Ann. Mtg. Conf. Reg., 1988 , pp. 418-430 .

[6] H. W. Van Der Broeck, H. Skudelny, e G. V. Stanke, "Analysis and realization of a pulse width modulation based on voltage space vectors",IEEE Trans. Industry Appl., vol. 24 ,no. 1 , pp. 142-150 , Jan/Feb. 1988 .

[7] S. R. Bowes e R. I. Buloough, "Fast modelling techniques for microprocessor - based optimal PWM control of current fed inverter drivers", Proc. Inst. Elec. Eng., B- Elect. Power Appl., vol. 131, no.4 , pp. 59-75, 1988.

dhyay e N. Meher,"Microprocessor
eedback control strategy for a current
on motor drive.",IEEE Trans. Industry
9-288,Apr. 1989 .

olvimento de Um Analisador Lógico A
nputador Pessoal",dissertação de
ico de Aeronautica",São José dos

icação dos Parâmetros Dinâmicos do
Filtro de Kalman",dissertação de
ngenharia de Itajubá,Itajubá, M. G. ,

nced pulse-width-modulated inverter
,pp. 145-154.

ox,"Several modulation techniques for
2,IA-8,pp.636-643.

C. J. Nordby ,"Pulse width inverter
modulation ",IEEE conference record
.

ti e C. J. Nordby ,"Pulse Width
ves with improved modulation",IEEE
.

tion of pulse-width-mudulation theory
rive inverters",IEEE conference of

unket,"Development of a modulation
drive ",IEEE Trans.,1979,IA-15,pp.

wer factor for a square
ecord of IAS ,1978,pp.

"Introdution to PWM speed
',Electron. Components &

.aszlo,"PWM inverter for
power engineering

trol strategies for PWM
p.211-215 .

dulator circuit for PWM
3, 1978 , pp.493-501.

F. Donati , D. Sbragia e
for uniterruptible power

t digital control of a
conference record of IAS

ed control for single and
9.

ee-phase digital PWM
304.

sor-controlled three-phase
)- 120-122.

:ontroller for integrated
:e record of IAS , 1977 ,

pp. 471-476 .

[28] J. M. D. Murphy , L. S. Horwards e R. G. Hoft ,"Microprocessor control of a PWM inverter induction motor drive",IEEE,PESC. 1979 , pp.344-348 .

[29] W. Richter ,"Microprocessor-controlled inverter-fed synchronous motor drive",IEE Conf. Publ.,1979,pp. 161-164 .

[30] E. Dwyer e B. T. Ooi ,"A look up table based microprocessor controller for a three phase PWM inverter ",IEEE,IECI Proceedings Industrial and control application of microprocessors",1979, pp.19-22.

[31] H. Le-Huy ,"A microprocessor-controlled pulse-width-modulated inverter",Ibid,Industrial applications of microprocessors",1978,pp. 223-226.

[32] L. C. P. Humblet , F. G. C. De Buck ,B. Verbeke e P. De Valck ,"A realisation example of a microprocessor-driven PWM transistor inverter",IEE Conf. Publ.,1979,pp. 151-156 .

[33] P. C. Sen ,J.Trezise e M. Sack ,"Microprocessor control of an induction motor with flux regulation",IEEE Conference record of IAS, 1978,pp. 664-668.

[34] F. De Buck , L. Humblet , L. De Wael e G. Hoebeeck ,"A microprocessor-based frequency inverter,suitable for experimental purposes",International conference on electrical machines,proceedings part 2 ,1978, pp. E6/2-1 até E6/2-9.

[35] J. M. D. Murphy , R. G. Horft e L. S. Howard ,"Controlled slip operations of an induction motor with optimum PWM waveforms",IEE Conf.Publ. 179 , 1979 ,pp. 157-160.

[36] E. Anzaldi , E. Bassi , F. Benzi , G. Petrecca ,"Microprocessor techniques in electrical variable speed drives:a design methodology", in 'Journées d'electroniques:"Microprocessors:A tool for the future ", 1979 ,

n of microprocessors to A.C. and D.C
s",IEEE conference record of IAS ,

dpierre e J. C. Hapiot ,"Apllication
ome electrical variable speed drives",
. Industrial apllications of
174-180.

Haneyoshi , "A microprocessor-based
system of a inverter-fed induction
87-193.

armonic elimination of microprocessor
microprocessors",1979,pp. 278-283.

Fiorini ,"A microcomputer-based
troller for PWM inverters",ibid, 1980,

igning microprocessor-controlled PWM
978 , pp. D3-1 a D3-6.

Hoft ,"Optimum PWM waveform of a
nverter", IEEE, PESC , 1978 , pp.

ected harmonic reduction in static
, 1964 ,CE-83,pp. 374-378.

ultiple pulse modulation in static
output harmonic and provides smooth
ibid,IGA-6,pp. 357-360.

Hoft ,"Generalised techniques of
973,IA-9,pp. 310-317 .

eneralised techniques of
l in thyristor inverters",

alysis and comparison of
motor drives",Proceedings

pulse-width-modulation for
IA-13 ,pp 38-44.

Sciavicco,"On the ordinary
roceedings IFAC, 1977,pp.

e A. F. Mergen ,"Loss
notors with PWM inverter
979 , 2, (5),pp. 167-173 .

,"Harmonic reduction in
approach",IEEE conference

il ,"A control strategy for
Trans.,1980, IECI-27,

an improved output voltage
nference,1979,pp. 376-381.
dal pulse-width-modulated
. 1279-1285 .

approach to the analisy
in power convertors",ibid.,

processor control of PWM

inverters", IEE Proc., vol. 128, pt. B, nº 6, Nov. 1981.

[58] F. C. Zach, R. Martinez, S. Keplinger, e A. Seiser, "Dynamically optimal switching patterns for PWM inverters drives", IEEE Trans. Ind. Appl. , vol. 21, nº 4, Julho/Agosto.1985.

[59] M. Varnovitsky, "A Microcomputer-based control signal for a three-phase switching PWM inverter", IEEE Trans. Ind. Appl., vol. IA-19, nº 2, Março/Abril. 1983.

[60] E. O. Brigham, The Fast Fourier Transform. Englewood Cliffs, NJ: Prentice Hall, 1988.

[61] K. E. Addoweesh, W. Shepherd e L. N. Hulley, "Induction Motor Speed Control Using a Microprocessor-Based PWM Inverter", IEEE Trans. Ind. Electronic, vol. 36, nº 4, Nov. 1989, pg. 516-522.

ED. PPG.
V.
DOAÇÃO

BIM - EFEI

DATA	Rubri a

EFEI - BIBLIOTECA MAUÁ
8200691
NÃO DANIFIQUE ESTA ETIQUETA