UNIVERSIDADE DE SÃO PAULO

INSTITUTO DE FÍSICA E QUÍMICA DE SÃO CARLOS

CONTRIBUIÇÕES AO ESTUDO E DESENVOLVIMENTO DE SISTEMAS CONFIÁVEIS COM MICROPROCESSADORES.

Valentin Obac Roda

Tese apresentada ao Instituto de Física e Química de São Carlos, Universidade de São Paulo, para o Concurso de Livre Docência.

Departamento de Física e Ciência dos Materiais.

São Carlos

1991

USP / IFQSC / SBI

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC
FÍSICA

AGRADECIMENTOS

A todos que direta ou indiretamente contribuiram para a consecução deste trabalho, e em particular:

À Odete, minha esposa, pelo companheirismo e pela colaboração na revisão dos textos.

À Lia M. Canhe Chináglia, secretária do Grupo de Instrumentação e Informática (GII) do DFCM/USP pela organização da documentação dos diversos projetos de pesquisa relacionados com este trabalho.

À Maria B. de Souza (Benê), desenhista do GII, pela paciência e eficiência na editoração eletrônica dos desenhos.

Aos técnicos do Laboratório de Instrumentação Eletrônica, Lírio, Ivanilda, Marcos e Ailton pela colaboração na implementação dos circuitos.

Ao Departamento de Física e Ciencia dos Materiais/USP pelas condições propícias ao desenvolvimento do trabalho.

À Fundação de Amparo a Pesquisa do Estado de São Paulo e ao Conselho Nacional de Desenvolvimento Tecnoógico pelos diversos auxilios que permitiram que as pesquisas aqui relatadas fossem realizadas.

RESUMO

No presente trabalho estão apresentadas algumas contribuições para a área de sistemas confiáveis com microprocessadores.

Dois aspectos relevantes foram abordados: o estudo da interferência externa, particularmente dos distúrbios da rede elétrica, na operação de microcomputadores e a utilização de técnicas de redundância como meio de aumentar a confiabilidade desses sistemas.

Para a análise do efeito dos distúrbios elétricos sobre microcomputadores, foi desenvolvido um sistema de testes dedicado. A observação do efeito dos distúrbios foi feita comparando-se o comportamento de um equipamento sujeito aos distúrbios com uma réplica idêntica considerada como padrão. Utilizando o sistema de testes desenvolvido submeteu-se um microcomputador dedicado aos distúrbios elétricos, tendo-se obtido resultados bastante interessantes que estão aqui relatados.

Como contribuição ao uso de redundância para aumentar a confiabilidade de sistemas com microcomputadores são apresentados: o desenvolvimento de memórias com correção/detecção de erros utilizando códigos de Hamming, o desenvolvimento de um microcomputador tolerante a falhas com uma arquitetura denominada arquitetura em anel e a proposta de uma arquitetura tolerante a falhas DUPLEX/HÍBRIDA com repetição parcial de operações para recuperação de falhas transientes.

SUMMARY

In this work, some contributions in the field of microprocessor based systems reliability are presented.

Two major aspects are covered, the effect of external interference, particularly electrical disturbances, on microcomputer behavior and the use of redundancy to increase the reliability of microcomputer systems.

A dedicated test set-up has been developed to analyze the effects of electrical disturbances on microcomputers. The effects of such disturbances have been observed by applying controlled electrical pulses on the power rails of a microcomputer and comparing its outputs with an identical "gold" microcomputer. By using the test set-up, interesting results were obtained which are related.

Some contributions have been done for the development of reliable microprocessor based systems using redundancy techniques. These are: memory systems using Hamming detecting/correcting codes, a ring type fault tolerant microcomputer based on N modular redundancy and a DUPLEX/HYBRID architecture using a "rollback" technique for transient faults recovery.

ÍNDICE

CAPÍTULO 1: CONFIABILIDADE EM SISTEMAS DE COMPUTAÇÃO 1

1. INTRODUÇÃO 1

1.2. CONCEITOS BASICOS DE CONFIABILIDADE . 2

1.3. TAXONOMIA DAS FALHAS. 6

1.4. FALHAS PERMANENTES E FALHAS TRANSIENTES 9

1.5. CAMINHOS PARA O AUMENTO DA CONFIABILIDADE 12

1.6. PROPOSTA DO TRABALHO 15

CAPÍTULO 2: ESTUDO SOBRE O EFEITO DE DISTÚRBIOS DA REDE ELÉTRICA NA OPERAÇÃO CONFIÁVEL DE MICROCOMPUTADORES 17

2.1. INTRODUÇÃO 17

2.2. SOBRE A QUALIDADE DO SUPRIMENTO DE REDE ELÉTRICA 18

2.3. DIAGNÓSTICO E TESTE DE DISTÚRBIOS DA REDE EM MICROCOMPUTADORES 25

2.4. CARACTERIZAÇÃO DO RUÍDO DA REDE 27

2.5. Monitor microprocessado de transientes da rede. 32

2.6. ENSAIOS SOBRE O EFEITO DE DISTÚRBIOS DA REDE ELÉTRICA EM MICROCOMPUTADORES 34

2.7. SISTEMA DE TESTES PARA VERIFICAR O COMPORTAMENTO DE MICROCOMPUTADORES SUJEITOS A DISTÚRBIOS DA REDE ELÉTRICA 36

2.8. MECANISMO DE DETECÇÃO E

IDENTIFICAÇÃO DAS FALHAS. 39
2.9. DESCRIÇÃO DOS CIRCUITOS IMPLEMENTADOS 41
2.9.1. Microcomputador supervisor. 42
2.9.2. Os módulos sob teste 44
2.9.3. Comparador 45
2.9.4. Gerador de distúrbios 46
2.10. DESCRIÇÃO DOS PROGRAMAS UTILIZADOS. 49
2.10.1. Rotinas do Supervisor 49
2.10.2. Rotinas dos módulos sob teste 49
2.11. ENSAIOS SOBRE A SUSCEPTIBILIDADE AOS DISTÚRBIOS DA REDE. 50
2.11.1. Considerações gerais sobre os distúrbios aplicados. 50
2.11.2. Resultados dos ensaios 52
2.12. PROPOSTAS PARA MINIMIZAR O EFEITO DOS DISTÚRBIOS EM EQUIPAMENTOS COM MICROPROCESSADORES. 57
2.13. CONCLUSÕES 58

CAPÍTULO 3: DESENVOLVIMENTO DE ALGUNS SISTEMAS CONFIÁVEIS 61
3.0. INTRODUÇÃO. 61
3.1. DETECÇÃO E CORREÇÃO DE ERROS EM MEMÓRIAS PARA MICROCOMPUTADORES . 61
3.1.1. Considerações sobre a confiabilidade de memórias a semicondutor. 62
3.1.2. Codificação para detecção e correção de erros 63
3.1.3. Implementação do esquema de única

detecção/única correção para uma memória de 8 bits. 74
3.1.4. Implementação do esquema de dupla detecção/única correção para uma memória de 8 bits. 78
3.2 - MICROCOMPUTADOR TOLERANTE A FALHAS COM ARQUITETURA EM ANEL 81
3.2.1. Aspectos gerais da arquitetura em anel . . 81
3.2.2. Características da arquitetura implementada 84
3.2.3. Detalhes da implementação do microcomputador tolerante a falhas em anel 86
3.2.4. Programas de gerenciamento para o microcomputador tolerante a falhas, em anel 91
3.2.5. Sincronização no microcomputador tolerante a falhas, em anel. 94
3.2.6. Considerações sobre a confiabilidade e o desempenho do microcomputador tolerante a falhas, em anel. 96
3.3. ARQUITETURA TOLERANTE A FALHAS DUPLEX-HÍBRIDA 99
3.3.1. Sistema DUPLEX 99
3.3.2. Arquiteturas DUPLEX-HÍBRIDAS 103

3.3. ARQUITETURA TOLERANTE A FALHAS DUPLEX-HÍBRIDA 104
3.3.1. Sistema DUPLEX 99
3.3.2. Arquiteturas DUPLEX-HÍBRIDAS 102

3.3.3. Considerações sobre o chaveamento dos

módulos de reposição em arquiteturas HÍBRIDAS 106
3.3.4. Sistema Duplex Híbrido com repetição parcial de operações 111
3.4. CONCLUSÕES. 119

CAPÍTULO 4: CONCLUSÕES FINAIS E PERSPECTIVAS FUTURAS 122

REFERÊNCIAS BIBLIOGRÁFICAS 129

APÊNDICE A: DIAGRAMAS ESQUEMÁTICOS DAS MEMÓRIAS COM DETECÇÃO/CORREÇÃO DE ERROS 138

APÊNDICE B: CONSIDERAÇÕES SOBRE A CONFIABILIDADE DO SISTEMA DUPLEX-HÍBRIDO. 141

APÊNDICE C: EQUAÇÕES DA CONFIABILIDADE DOS SISTEMAS DUPLEX/HÍBRIDOS 146

CAPÍTULO 1

CONFIABILIDADE EM SISTEMAS DE COMPUTAÇÃO

1. INTRODUÇÃO

Sistemas utilizando computadores tem tido crescente aplicação em quase todas as áreas da atividade humana: acadêmicas, comerciais, controle de processos, espaciais, médicas, militares e de transporte coletivo, entre outras.

Existem paises onde cidades são quase que inteiramente informatizadas, utilizando o computador como meio de vida [OLIVEIRA JR., 1988]. O aumento da dependência humana na informática torna necessário que sistemas de computação sejam construidos de maneira que se possa confiar nos serviços prestados pelos mesmos.

Quando um sistema com microprocessador deixa de operar corretamente, dependendo da aplicação, os resultados podem ir do desagradável ao catastrófico. Uma falha momentânea ou transiente pode fazer com que o usuário perca horas de edição no caso de parada do computador, bases de dados podem ser alteradas ou perdidas. Em algumas aplicações mais específicas a falha de operação do equipamento poderá resultar em riscos à vida humana ou grandes perdas de capital, como é o caso de equipamentos militares, aeroespaciais e daqueles utilizados em transações comerciais. Equipamentos com maior confiabilidade podem também ser importantes como estratégia de "marketing" se

SERVIÇO DE DIFUSÃO

puderem ser comercializados a preços competitivos com equipamentos tradicionais.

As causas de falhas dos equipamentos são as mais diversas possíveis e na maioria dos casos não previsíveis: componentes podem ter seus parâmetros alterados por envelhecimento (oxidação de conectores e soquetes, contaminação de transistores e circuitos integrados, deterioração da isolação de condutores); distúrbios externos ao sistema, principalmente distúrbios de rede e eletromagnéticos, podem introduzir alterações na operação dos equipamentos; programas podem apresentar operação incorreta em algumas circunstâncias.

Em aplicações nas quais é imprescindivel manter a operação correta dos equipamentos devem ser utilizadas técnicas adequadas e procedimentos que permitam que o equipamento continue a operar corretamente e de maneira confiável, independente de falhas de circuitos, programas ou falhas induzidas.

1.2. CONCEITOS BÁSICOS DE CONFIABILIDADE

Um sistema que execute as aplicações de um usuário de maneira suficientemente precisa e em um tempo satisfatório pode ser considerado confiável, embora uma análise mais minuciosa do sistema possa vir a indicar possíveis limitações. Na prática, a avaliação da confiabilidade de um sistema pode ser conseguida por critérios pragmáticos como por exemplo: o intervalo médio de tempo em que o equipamento opera corretamente sem precisar ser consertado. Entretanto, a obtenção de um determinado grau de confiabilidade para uma dada aplicação envolve um grande número de elementos que muitas vezes é impossível controlar. Segundo **AVIZIENIS [1975]** computação confiável significa a execução correta de um determinado conjunto de algoritmos e

envolve os seguintes elementos:

Especificações completas e precisas de todos os programas utilizados.

Verificação e teste de validade de todos os programas.

Eliminação de erros de projeto dos circuitos ("hardware").

Execução correta dos programas e proteção dos dados no caso de falha dos circuitos.

Proteção do sistema computacional contra paradas induzidas por erros ou invasão premeditada de programas ou dados.

Para implementação eficiente (custo efetiva) de sistemas confiáveis, deve-se ter meios de avaliar a confiabilidade para uma dada aplicação e de prever qual o ganho de confiabilidade para um dado investimento. Pesquisas teóricas sobre confiabilidade tem definido algumas medidas básicas para avaliação da confiabilidade e desenvolvido técnicas de modelamento matemático para prever a confiabilidade de sistemas específicos.

Para saber se um sistema atende a certos requisitos de confiabilidade ou para compará-lo com outros sistemas são necessários dados quantitativos. Alguns dos conceitos associados com a operação correta de sistemas e consequentemente com a sua confiabilidade são:

Confiabilidade ("Reliability"): é a propriedade que indica a probabilidade condicional do sistema ter sobrevivido o intervalo de tempo 0 a t, considerando-se que estivesse operando corretamente no instante de tempo t=0 **[SIEWIOREK e SWARZ, 1982]**. O parâmetro confiabilidade é utilizado para equipamentos nos quais a reparação não é possível ou é extremamente onerosa (por exemplo equipamentos a bordo de satélites) ou para equipamentos que estão atuando em funções críticas e cuja operação não pode ser interrompida ainda que seja durante a duração de

um conserto (por exemplo computadores de bordo de aeronaves).

Disponibilidade ("Availability"): é a propriedade que indica a probabilidade de que o sistema esteja em operação em um determinado instante de tempo t **[SIEWIOREK E SWARZ, 1982]**. A disponibilidade é um parâmetro utilizado em equipamentos nos quais é possivel interromper, sem maiores consequências, a operação do equipamento para que ele seja consertado. A disponibilidade é também indicada pela relação do tempo de operação sobre o tempo total de vida do equipamento.

$$Disponibilidade = \frac{tempo\ de\ operacao}{tempo\ de\ operacao + tempo\ inoperante}$$

Um conceito relativamente recente é o de Segurança de funcionamento ("Dependability") **[LAPRIE, 1985; PAULA JR. et al., 1989]** que indica a confiança do usuário (humano ou físico) no servico fornecido por um sistema. A segurança de funcionamento de um sistema de computação pode ser definida como a qualidade de serviço fornecida pelo sistema de maneira que o usuário possa justificadamente depositar confiança neste sistema. O serviço fornecido por um sistema é o comportamento desse sistema, conforme percebido por outro sistema (ou usuários), interagindo com o sistema em questão **[FRAGA e LEMOS, 1987]**. Segundo este conceito, as duas principais formas utilizadas para quantificar a qualidade do serviço fornecido em relação ao serviço especificado são a confiabilidade e a disponibilidade (definidas agora em função do serviço fornecido). *Confiabilidade* é uma medida do serviço continuamente realizado a partir de uma referência de tempo inicial. *Disponibilidade* é a medida do serviço realizado em relação à alternância de estados de operação correta

e interrupções.

Para comparação de diferentes sistemas, são também utilizados o *tempo médio para a ocorrência de falhas* ("Mean Time To Failure - MTTF") para sistemas não reparáveis, definido como:

$$MTTF = \int_0^{\infty} R(t)\,dt \qquad R(t) \text{ - Confiabilidade}$$

e o *tempo medio entre falhas* ("Mean Time Between Failures - MTBF") para sistemas reparáveis. Tambem é utilizado o *tempo médio entre reparos* ("Mean Time to Repair - MTTR"). Assim, o tempo médio entre falhas pode ser definido como:

$$MTBF = MTTF + MTTR$$

Tempo médio entre falhas = tempo médio para a ocorrência de falhas + tempo médio entre reparos

Em algumas aplicações específicas é importante que o sistema funcione corretamente durante um certo tempo pré-determinado, esse tempo é chamado tempo de missão ("mission time"). Por exemplo o sistema eletrônico de um míssil deve funcionar sem falhas no intervalo de tempo entre o lançamento do míssil e o instante em que o míssil atinge o alvo; um computador de bordo de uma aeronave deve operar sem falhas durante o tempo em que a aeronave está voando.

Para escolher o melhor de dois sistemas A e B, dado um certo tempo de missão T, a comparação dos sistemas requer o conhecimento prévio das confiabilidades R_A e R_B. Dessa maneira são definidos os fatores de melhora da confiabilidade e de melhora do tempo de missão **[AVIZIENIS, 1978]**.

O *fator de melhora da confiabilidade* ("reliability improvement factor - RIF") é definido como:

$$RIF = \frac{(1-R_A)}{(1-R_B)}$$

para um dado tempo de missão T. Este fator indica qual a melhora em confiabilidade se for utilizado o sistema B em lugar do A.

Se o tempo de missão não for especificado, o *fator de melhora do tempo de missão* ("mission time improvement factor") é uma medida comparativa conveniente. Este fator é definido como:

$$MTIF = \frac{T_A}{T_B} \qquad para\ um\ determinado\ R_{espec}$$

Onde, R_{espec} é uma dada confiabilidade e T_A e T_B são os instantes de tempo em que as confiabilidades do sistema são iguais a R_{espec}.

1.3. TAXONOMIA DAS FALHAS.

Segundo **AVIZIENIS [1978]**, um sistema computacional pode ser visto sob os pontos de vista físico e lógico (ou operacional).

Do ponto de vista puramente físico um sistema computacional é uma máquina constituida por um número grande de componentes eletrônicos, mecânicos e eletromecânicos. Os componentes obedecem às leis da física e o comportamento do sistema pode ser descrito em termos das funções no tempo das variáveis físicas: corrente, tensão, posição, etc.

Do ponto de vista lógico, os computadores são caixas pretas que executam funções de processamento de informação para atender a determinadas necessidades humanas.

O comportamento da máquina lógica é satisfatório enquanto os parâmetros dos componentes físicos e a velocidade de operação

estiverem dentro de limites inicialmente especificados.

Na prática, a especificação do sistema deve atender os requisitos do projetista, do construtor, do vendedor e do usuário, sendo por essa razão dificil de caracterizar completamente, sendo muitas vezes imprecisa e incompleta. Para se poder avaliar a confiabilidade de um sistema é necessário definir uma especificação absoluta ("authoritative specification") que nem sempre é a mais completa porém é importante que ela possa ser aplicada a qualquer situação para determinar se o comportamento do sistema é ou não aceitável [ANDERSON e LEE, 1982]. Mudanças físicas indesejáveis (defeitos) ocorrem por alteração dos parâmetros físicos dos componentes em todos os tipos de circuitos eletrônicos. A ocorrência dos defeitos é aleatória e não controlável.

Os defeitos de circuitos podem ser classificados como: defeitos permanentes, defeitos intermitentes ou temporários e defeitos causados por interferência externa à operação do sistema. Os defeitos causam desvios nos valores inicialmente especificados em projeto de uma ou mais variáveis lógicas do computador. Essa mudança nas variáveis é normalmente chamada falha lógica ou simplesmente falha. O sintoma imediato de qualquer defeito no computador é a falha. A falha geralmente causa um erro no programa que está sendo executado, que geralmente faz com que a instrução seja executada incorretamente causando um erro no resultado computado.

A ocorrência de uma falha pode causar um erro, que é um desvio da máquina lógica do seu fluxo normal de programa. Quando o erro ocorre, a máquina lógica deixa de executar corretamente pelo menos uma das tarefas inicialmente especificadas e falha, total ou parcialmente, em executar a função de processamento de dados.

O conceito de falhas pode ser estendido para falhas em

programação que incluem desvio indesejado do fluxo normal de programas e corrupção de dados e de programas.

Uma escolha sistemática de técnicas adequadas para minimizar as falhas em sistemas de computação deve começar pela classificação das falhas e identificação das classes de falhas que podem ocorrer no sistema que está sendo projetado. As falhas podem ser classificadas de acordo com os seguintes parâmetros [AVIZIENIS 1982]:

Extensão (local ou distribuida)

Valor (determinado ou indeterminado)

Duração (permanente ou intermitente)

Causa (física ou humana)

A extensão de uma falha especifica quantas variáveis lógicas são afetadas simultaneamente nos circuitos do computador. Falhas locais (estatísticamente independentes) são afetadas apenas por simples variáveis ao passo que falhas distribuidas (relacionadas) podem ser afetadas por uma ou mais variáveis fisicas, um módulo de reposição ou um sistema inteiro. A proximidade física dos dispositivos lógicos atuais de alta integração (LSI e VLSI) atuais contribui para a maior ocorrência de falhas distribuidas devendo se dar mais atenção à sua ocorrência. Falhas distribuidas podem também ser causadas por falhas simples em elementos não lógicos de um computador, como por exemplo: fontes de alimentação, bases de tempo comuns, barramentos de dados e chaves de reconfiguração

O valor de uma falha é determinado quando os valores lógicos afetados assumem um valor constante, por exemplo amarrado ("stuck at") no nível lógico zero ou amarrado no nível lógico um durante toda a duração da falha. O valor de uma falha é indeterminado quando os valores lógicos afetados variam entre 0 e 1, porém não de acordo com as especificações de projeto.

A classificação quanto à causa distingue as falhas físicas, causadas por situações físicas adversas (perturbações internas ou externas ao sistema), de falhas humanas causadas pela interação do ser humano com a máquina seja na fase de projeto, construção, operação ou reparo.

Na metodologia de projeto a classificação de falhas quanto à duração (transientes e permanentes) é a mais importante porque requer diferentes abordagens de recuperação da máquina. Uma reinicialização da máquina é suficiente na maioria dos casos para corrigir erros causados por falhas transientes. Já no caso de falhas permanentes é necessária a substituição dos componentes danificados ou a reconfiguração dos circuitos.

A classificação quanto à duração e valor é aplicável tanto a falhas transientes como a falhas permanentes. Por ser interesse especial deste trabalho, abordaremos de maneira mais completa este tipo de falhas no item a seguir.

1.4. FALHAS PERMANENTES E FALHAS TRANSIENTES.

De acordo com a duração, os defeitos podem ser:

Permanentes quando causados por alterações permanentes dos parâmetros dos componentes. Por exemplo sinais de entrada amarrados em um determinado nivel lógico ("stuck at"), componentes curto circuitados, ligações abertas, etc.

Transientes, são defeitos cuja manifestação tem duração limitada. Na maioria dos casos estes defeitos são de curta duração e são causados por agentes externos ao sistema, como por exemplo interferência da rede ou eletromagnética ou alterações temporárias nos parâmetros dos componentes por aumento da temperatura ou vibrações.

Intermitentes ou pseudo transientes, são causados por defeitos permanentes nos componentes que se manifestam na presença de uma determinada combinação de variáveis lógicas. Esse tipo de defeitos ocorre frequentemente em circuitos integrados VLSI e pode tambem ocorrer em situações de temporização marginais.

As falhas permanentes são causadas por defeitos permanentes em componentes. A predição de falhas permanentes requer os seguintes passos:

- Estabelecer a probabilidade relativa de ocorrência de falhas locais e falhas distribuidas, identificando para estas últimas a extensão.
- Identificar outras possíveis fontes de falhas catastróficas como falhas na base de tempos principal, falhas na fonte de alimentação, barramentos, etc.
- Estabelecer a probabilidade relativa do modo de falha indeterminado.
- Determinar a confiabilidade esperada do sistema através da determinação da taxa de falha de todos os componentes.

A proximidade física dos elementos lógicos em circuitos LSI e VLSI aumenta a probabilidade de ocorrência de falhas distribuidas. Os efeitos mecânicos, térmicos e de radiação devem ser avaliados em termos de distribuição de falhas. A estimativa da probabilidade e extensão das falhas depende da tecnologia utilizada e do encapsulamento dos dispositivos semicondutores. Além das causas mencionadas, falhas distribuidas podem ser causadas por defeitos em alguns elementos críticos não lógicos, como por exemplo: bases de tempo comuns, fontes de alimentação, barramentos de dados e chaves de reconfiguração do computador. A identificação dos elementos e seus modos de falha serve para desenvolver técnicas para sua proteção durante a fase de projeto dos sistemas. Uma avaliação quantitativa da confiabilidade de um

SERVIÇO DE BIBLIOTECA

computador requer o conhecimento das taxas de falhas dos diversos componentes que constituem a máquina.

As falhas transientes ocorrem nos circuitos do computador e causam erros de processamento porém sem causar erros permanentes. Um caminho sistemático para predição das falhas transientes envolve os seguintes passos:

- As classes esperadas de falhas transientes são classificadas de acordo com sua extensão e valor.
- Os parâmetros dependentes do tempo (distribuição estatística de ocorrência e duração) são postulados para cada classe de falha transiente esperada.

A recuperação bem sucedida é então definida em relação a cada classe de falhas transientes e uma medida quantitativa para a recuperação é obtida pela análise do experimento.

As falhas intermitentes são por sua vez caracterizadas pelos seguintes parâmetros dependentes do tempo:

- **Duração** - No caso geral é uma variável aleatória cuja distribuição é governada pelo fenômeno que causa as falhas intermitentes. O limiar de duração para um determinado sistema tolerante a falhas é um intervalo de tempo determinado, tal que todas as falhas cuja duração ultrapasse esse limiar são consideradas como permanentes pelo mecanismo de recuperação de falhas.
- **Taxa de ocorrência das falhas intermitentes** - é uma variável aleatória cuja distribuição depende dos componentes e do ambiente. Um modelo conveniente para este parâmetro é a distribuição de Poisson.

A confiabilidade R(t) de um sistema de computação é função da taxa de falha de suas partes componentes, relacionando-se portanto somente as falhas de circuito do sistema. Uma medida diferente que indica a habilidade de um computador em tolerar

falhas intermitentes é a chamada Capacidade de Sobrevivência ("Survivability"), sendo definida da seguinte maneira:
Capacidade de Sobrevivência é a probabilidade de que um conjunto especificado de programas satisfaça as condições de sobrevivência no instante t = T, considerando-se que elas tenham sido satisfeitas no instante de tempo inicial, t = 0.

Para completar a especificação de um computador tolerante a falhas transientes, a recuperação de transientes com sucesso pode ser definida para cada classe de falhas transientes. Neste contexto, um computador consiste em todos os componentes de circuito, programas do sistema e programas de aplicação que devem estar operacionais durante um determinado tempo de missão. Os circuitos do computador não são danificados pelas falhas transientes, porém a falha ocorre pelo erro de processamento dos programas que estão sendo processados pelos circuitos.

A definição de recuperação bem sucedida varia de acordo com a severidade do defeito causado pelas falhas. Por exemplo a recuperação bem sucedida de uma interrupção temporária de energia elétrica ou um distúrbio eletromagnético de alta intensidade pode ser especificada como tolerância a falhas parciais, desde que não ocorram modificações nos programas e o seu processamento possa ser repetido a partir de uma reinicialização da máquina. Por outro lado, no caso de um bi-estável ("Flip-flop") intermitentemente ruidoso o tratamento de recuperação de falhas requer uma tolerância a falhas completa, para todo o conjunto de programas.

1.5. CAMINHOS PARA O AUMENTO DA CONFIABILIDADE

Existem dois caminhos básicos para aumentar a

confiabilidade de sistemas de computação: *Intolerância a Falhas* ("Fault Intolerance, Fault avoidance") e *Tolerância a Falhas* ("Fault Tolerance") [AVIZIENIS, 1978; SIEWIOREK e SWARZ, 1982].

A intolerância a falhas é baseada nas técnicas tradicionais de projeto digital sendo que o aumento de confiabilidade é conseguido pela utilização de componentes e técnicas de montagem especiais com maior confiabilidade. Após a construção do equipamento são realizados testes exaustivos nos circuitos (por exemplo ciclos térmicos e de vibração) e programas, procurando antecipar e corrigir possíveis erros. A filosofia da intolerância a falhas é eliminar todas as possíveis causas de falha durante o projeto e construção do sistema. É possível através do controle dos parâmetros físicos dos componentes e do aperfeiçoamento do processo de fabricação reduzir a taxa de falha dos componentes, entretanto após um certo limiar de taxa de falha os custos passam a ser proibitivos. Para complementar a intolerância a falhas são utilizadas técnicas eficientes de reparação para que o sistema seja consertado o mais rápidamente possível após ser detectada a falha. As modernas técnicas de reparo podem incluir nos circuitos: auto-diagnose [PEATMAN, 1981], análise de assinaturas [GORDON e NADIG, 1977] e emulação.

Embora seja possível determinar com razoável precisão as taxas de falha de lotes de componentes, individualmente não é possível prever o instante em que cada componente irá falhar. Pela complexidade dos circuitos utilizados nos modernos sistemas de computação e pelo grande número de interações nos programas é práticamente impossível construir-se grandes sistemas totalmente livres de falhas. O teste de todas as combinações possíveis que venham a causar falhas também é impossível e falhas podem ocorrer sob uma combinação não prevista de circunstâncias para

a qual os testes não foram realizados.

O caminho alternativo para se aumentar a confiabilidade de sistemas é a tolerância a falhas. Na tolerância a falhas é aceita a possibilidade de erros no sistema e são implementados recursos adicionais (redundância), tanto a nível de circuitos como de programas, que permitem que o sistema atenda os requisitos mínimos de computação, mesmo na ocorrência de falhas. Segundo **AVIZIENIS [1978]** a tolerância a falhas é o atributo de sobrevivência da máquina lógica e sua função é causar o retorno da máquina lógica dos estados errôneos para um comportamento especificado, assegurando a sobrevivência das atividades dos processos de informação. A redundância na tolerância a falhas pode assumir várias formas, como por exemplo circuitos para votação de sinais **[von NEUMAN, 1956]**, chaveamento de módulos de reposição para substituir um subsistema falho **[LOSQ, 1976; MATHUR e AVIZIENIS, 1970]** ou reexecução de programas após a detecção da falha.

A Intolerância e a Tolerância a falhas não devem ser considerados caminhos distintos, porém complementares e deve-se procurar as técnicas mais adequadas destes dois caminhos para conseguir o grau de confiabilidade desejado.

A utilização de técnicas para o aumento da confiabilidade durante a fase de projeto ou fabricação do equipamento de computação envolve inevitávelmente custos adicionais. Estes custos, entretanto, podem ser compensados pela confiança de operação mais segura e por uma possível diminuição dos custos de manutenção durante a vida útil do equipamento.

Na figura 1.1 é mostrado o custo de propriedade de um equipamento em função da confiabilidade, considerando-se o custo inicial do equipamento e o custo previsto para a manutenção. Verifica-se que o custo inicial do equipamento está diretamente

relacionado com a confiabilidade, sendo tanto maior quanto maior a confiabilidade. O custo de manutenção por sua vez diminui com a confiabilidade do equipamento. Existe um valor mínimo para o custo de propriedade do equipamento que é função do custo inicial e do custo de manutenção. O fabricante do equipamento deve ser capaz de estimar ainda durante a fase de projeto do equipamento, em primeiro lugar o custo da falha do equipamento e em segundo lugar o aumento da confiabilidade conseguido a um dado custo.

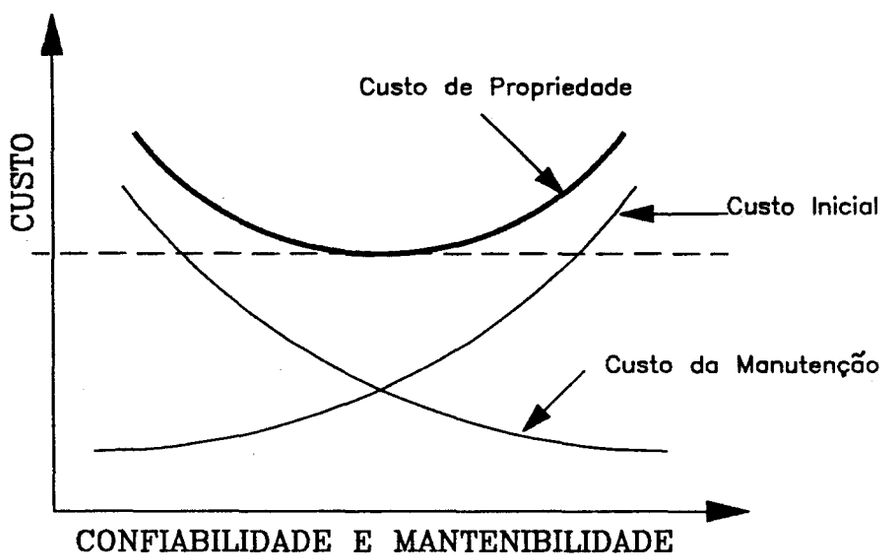


Figura 1.1: Custo de propriedade de um equipamento de computação.

1.6. PROPOSTA DO TRABALHO

No presente trabalho estão apresentadas alguns estudos e desenvolvimentos que realizamos como contribuição para o aumento da confiabilidade de sistemas com microcomputadores.

Neste capítulo apresentamos uma introdução sobre o tema de confiabilidade em sistemas de computação mostrando a

terminologia envolvida e os possíveis caminhos que normalmente são seguidos para aumentar a confiabilidade de sistemas de computação.

No segundo capítulo apresentamos os resultados obtidos no estudo do efeito de distúrbios da rede elétrica sobre a operação de microcomputadores. Neste capítulo é apresentado um sistema de testes especialmente desenvolvido para este estudo e os resultados obtidos utilizando o sistema de testes sobre um processador padrão.

No terceiro capítulo apresentamos o desenvolvimento de alguns sistemas com tolerância a falhas. São apresentados neste capítulo: um sistema de memória de pequeno porte utilizando códigos detetores corretores de erro, o desenvolvimento de uma arquitetura tolerante a falhas utilizando triplicação de módulos de processamento e o desenvolvimento de uma arquitetura tolerante a falhas baseada no sistema DUPLEX.

CAPÍTULO 2

ESTUDO SOBRE O EFEITO DE DISTÚRBIOS DA REDE ELÉTRICA NA OPERAÇÃO CONFIÁVEL DE MICROCOMPUTADORES.

2.1. INTRODUÇÃO

Falhas originadas por distúrbios da rede elétrica são consideravelmente frequentes em computadores. As consequências destas falhas podem ir do desagradável como a perda de um texto durante a edição, onde horas de trabalho do usuário são perdidas, até o catastrófico como corrupção de bancos de dados ou a completa disrupção da máquina.

Sistemas com microcomputadores são particularmente mais susceptíveis a distúrbios da rede elétrica do que outros equipamentos que não utilizam técnicas digitais. Várias formas de distúrbios, como ruídos da rede, picos, sobretensões e subtensões e outras irregularidades afetam a integridade dos sinais com informação binária e podem causar a falha de operação do equipamento. Os principais efeitos dos distúrbios são: alteração do estado lógico dos circuitos e mudança de bits nas memórias, em dispositivos de entrada/saída e na unidade central de processamento. Os distúrbios da rede podem levar a situações indesejáveis, como alteração de dados, desvios do fluxo dos programas, travamento de programas e saídas incorretas. Picos de tensão muito altos podem inclusive alterar os parâmetros físicos dos componentes induzindo a queima destes, levando à disrupção do computador.

Neste capítulo é apresentada uma abordagem sobre a caracterização dos distúrbios da rede elétrica e os resultados dos estudos realizados sobre o efeito destes distúrbios no comportamento de microcomputadores.

2.2. SOBRE A QUALIDADE DO SUPRIMENTO DE ENERGIA ELÉTRICA.

A qualidade da rede elétrica em regime permanente é estabelecida por padrões das companhias de eletricidade que delimitam desvios máximos permitidos para o nível de tensão, balanceamento de fases, distorção harmônica e estabilidade em frequência.

No Brasil, de acordo com a portaria 047 do Departamento Nacional de Águas e Energia Elétrica [DOU, 1978a] os limites considerados adequados para o fornecimento de energia elétrica são: ± 5% em relação à tensão nominal do sistema quando o atendimento for feito em tensão de transmissão, subtransmissão ou primária de distribuição e - 8,6% e 4% quando o atendimento for feito em tensão secundária de distribuição.

Distúrbios da rede elétrica, ou seja alterações nos níveis nominais de tensão, frequência ou na forma de onda senoidal são uma das causas da falha em computadores. Estes distúrbios podem ser gerados por: instabilidades nos geradores e nas linhas de transmissão, descargas elétricas na atmosfera e particularmente nas linhas de distribuição pelo chaveamento de cargas reativas, motores e outros equipamentos.

Os distúrbios elétricos que aparecem na alimentação de computadores são característicos do ambiente em que estes equipamentos são utilizados. Regiões rurais são mais susceptíveis a distúrbios devido a descargas elétricas da atmosfera nas linhas

de transmissão do que as cidades e a manutenção no caso de queima de fusíveis ou desligamento de chaves é usualmente mais demorada. Em ambientes industriais os distúrbios introduzidos por chaveamento de cargas reativas, particularmente motores podem ser consideravelmente amortecidos pela baixa impedância das linhas de transmissão. Em ambientes residenciais, as impedâncias das linhas de distribuição são maiores favorecendo a propagação dos distúrbios.

Para se ter uma idéia da magnitude dos distúrbios elétricos em alguns ambientes foi realizada, na década de 70, uma pesquisa nos E.U.A. para determinar o nível e a duração de distúrbios da rede elétrica em diversos ambientes residenciais e industriais **[MARTZLOFF e HAHN, 1970]**; foram registrados em redes de distribuição com tensão nominal de 120 V, distúrbios de até 5600 V durante a ocorrência de tempestades e de até 2500 V gerados por chaveamento de cargas elétricas. Outro autor **[WILLIAMSON, 1982]** cita que distúrbios de até 400 V foram registrados no sistema elétrico de um automóvel alimentado com bateria de 12 V.

Os tipos mais comuns de distúrbios da rede elétrica **[MARTZLOFF, 1987; RECHSTEINER, 1985]** são os seguintes:

(1) Variações de tensão - qualquer desvio ou oscilação do regime permanente da tensão elétrica acima ou abaixo de uma faixa pré-estabelecida.

(2) Ruido Elétrico - tensão de alta frequência superposta à tensão da rede elétrica, gerada por condições atmosféricas, sinais de rádio frequência e equipamentos que produzem faiscamento como motores com escovas e solda elétrica.

(3) Transientes ("Transients") e picos("Spikes")- aumento da tensão em menos de um ciclo seguido por oscilações amortecidas, causados por descargas elétricas atmosféricas,

chaveamento de cargas capacitivas ou indutivas, religamento da rede em caso de queda desta e equipamentos que produzem faiscamento.

(4) Falhas momentâneas - queda total da tensão por pelo menos 5 ms, que ocorrem como resultado de operações de chaveamento e falhas no equipamento das concessionárias.

(5) Sobretensões momentâneas - Aumento no valor nominal da tensão por uns poucos ciclos. São causados por diminuição de cargas elétricas e pelo chaveamento de linhas de alimentação.

(6) Subtensões momentâneas ou surtos ("Sags") - redução do valor nominal de tensão por mais de meio ciclo, causados por descargas elétricas atmosféricas e chaveamento de grandes cargas.

(7) Lacunas ("glitches" ou "dips") - redução no nível de tensão por menos de meio ciclo, causados por curto-circuitos breves e chaveamento de cargas.

(8) Interrupções ("Power outages") - Ausência prolongada de tensão, causada por acidentes relacionados à rede da concessionária e falhas nos transformadores.

Alguns dos distúrbios mencionados estão ilustrados na figura 2.1.

A observação de dados sobre os distúrbios da rede elétrica monitorados em diversos ambientes **[MARTZLOFF e HAHN, 1970; ALLEN e SEGALL, 1974; U.S. NAVY, 1977]** e a análise do efeito destes distúrbios sobre a operação de equipamentos eletrônicos, levou **KEY [1979]** a propor a seguinte classificação:

Tipo I: Distúrbios com duração entre 0,5 e 200 µS, seguidos ou não por oscilações amortecidas: frequências típicas de 0.2 a 5 Khz.

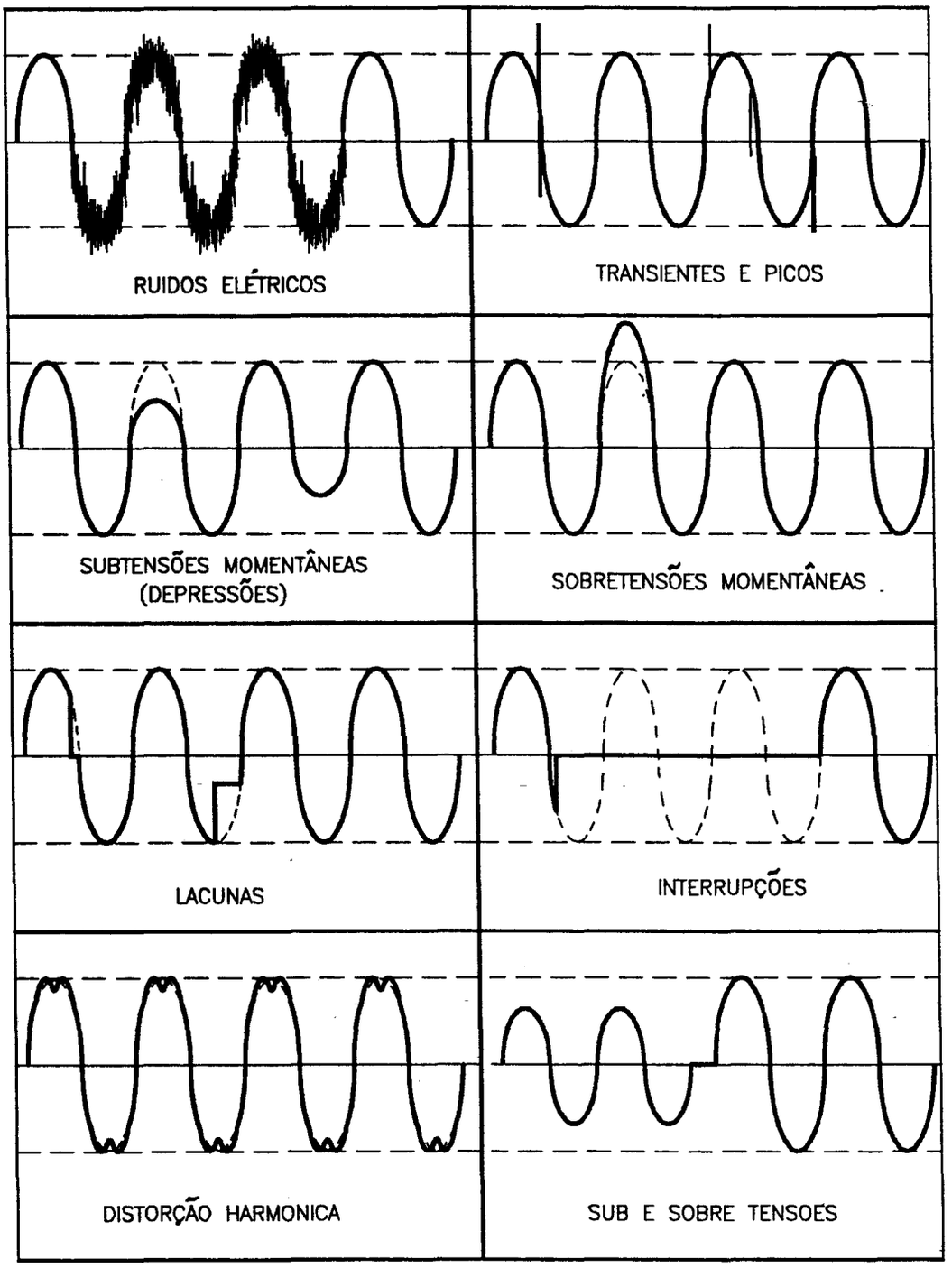


Figura 2.1: Principais distúrbios da rede elétrica.

Este tipo de distúrbios inclui transientes, interrupções de curta duração, ruído elétrico, picos e surtos. Verificou-se que estes distúrbios chegam a valores de 200% a 400%, e em alguns casos até maiores (picos), do valor nominal da tensão da rede.

Tipo II: Distúrbios com duração de 4 a 60 ciclos da frequência nominal de 60 Hz. Esta classe de distúrbios depende da distribuição de potência e inclui subtensões e surtos. Estes distúrbios normalmente assumem valores entre 80% a 110% do valor nominal da tensão da rede.

Tipo III: Interrupções longas com duração maior do que 2 s, podendo chegar até a algumas horas.

Apesar de algumas associações de fabricantes de computadores definirem limites de tolerância das suas máquinas à variação dos parâmetros da rede, não há uma definição formal do que deva ser uma alimentação adequada para computadores. **KEY [1979]** propôs uma envoltória de tolerância de tensão que ele considerou adequada para a alimentação de computadores; essa envoltória foi chamada de metas de projeto para fabricantes conscienciosos de computadores ("typical design goals of concious computer manufacturers"). A proposta de Key está ilustrada na figura 2.2.

De acordo com a envoltória proposta por Key, um computador pode tolerar distúrbios de grande amplitude e curta duração (menor do que 8 ms), diminuindo a tolerância para distúrbios de maior duração.

Na figura 2.3 está ilustrada a mesma envoltória proposta por Key, mostrando a classificação dos distúrbios também proposta por ele.

Analisando dados sobre a falha de computadores, Key identificou duas regiões de maior ocorrência de falhas: uma devido a sobretensões transientes de curta duração onde os distúrbios são gerados geralmente por chaveamento de cargas indutivas e

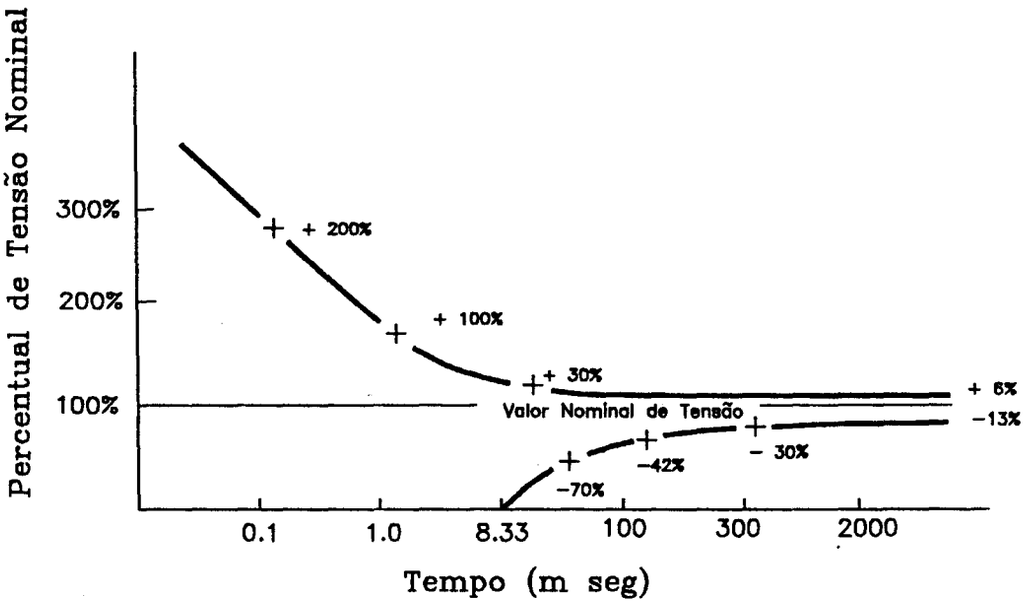


Figura 2.2: Limites de tolerância de tensão para computadores [KEY, 1979].

descargas elétricas atmosféricas e outra devido a subtensões de longa duração, causadas geralmente pelo chaveamento de grandes cargas, curto-circuitos e instabilidades na rede da concessionária.

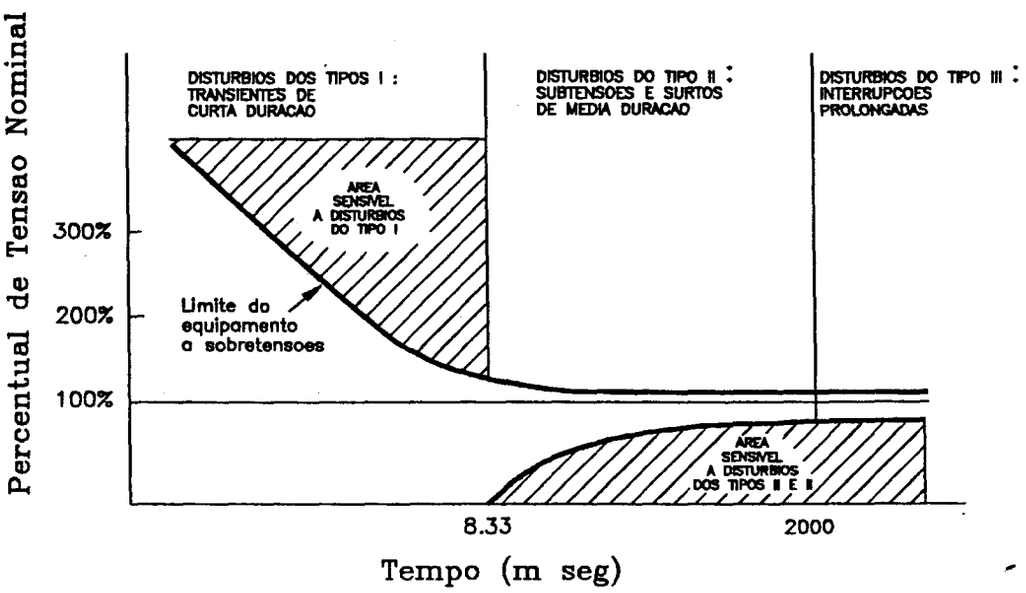


Figura 2.3: Regiões de susceptibilidade aos tipos de distúrbios definidos por KEY.

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC

No Brasil, existe uma portaria do Departamento Nacional de Águas e Energia Elétrica que normaliza a continuidade do fornecimento de energia elétrica [DOU,1978b] utilizando para isso os seguintes índices:

DEC - Índice de duração equivalente de interrupção por consumidor. Este índice expressa o intervalo de tempo, que em média, cada consumidor ficou privado do fornecimento de energia elétrica.

FEC - Índice de frequência equivalente de interrupções por consumidor. Este índice expressa o número de interrupções que em média cada consumidor do conjunto sofreu durante o intervalo de tempo considerado.

Na apuração dos índices acima devem ser consideradas todas as interrupções ocorridas em qualquer das partes do sistema elétrico e independente da sua natureza: programadas, acidentais, manobras, etc. Não são consideradas interrupções com duração inferior a 3 minutos.

Dados fornecidos pela Companhia de Força e Luz, indicam os seguintes índices obtidos para a região de São Carlos em 1989:

	DEC			FEC		
	MIN.	MAX.	MED.	MIN.	MAX.	MED.
1º Trim. 1989	1,47	5,51	2,94	1,21	3,31	2,14
2º Trim. 1989	1,27	4,74	2,47	0,92	3,27	1,60
3º Trim. 1989	1,40	5,12	2,97	1,32	5,17	2,46
4º Trim. 1989	0,97	2,91	2,41	0,90	2,56	1,78

Considerando: que do total de energia elétrica suprido

pelas concessionárias, a energia consumida pelos computadores é mínima (provavelmente menor do que 0.1% do total); que para a maioria dos consumidores, pequenos distúrbios da rede não causam maiores transtornos nos equipamentos ligados à rede (lâmpadas, motores, e equipamentos eletrônicos de consumo, como rádios, televisores, fornos de microonda, etc); seria extremamente oneroso e inclusive impraticável, na maioria dos casos que as concessionárias suprissem energia no grau de qualidade requerido pelos computadores. Por essa razão apesar de existir correlação entre os distúrbios da rede e a falha de computadores, as concessionárias de energia elétrica não aceitam qualquer responsabilidade nem oferecem qualquer remédio para as anomalias da rede. Existem também diversos distúrbios gerados nas redes de distribuição dos usuários e distúrbios gerados por condições atmosféricas como relâmpagos, vento, chuva e outros.

Os fabricantes de computadores por sua vez não garantem que os seus equipamentos vão operar corretamente na ocorrência de distúrbios da rede.

O dilema do usuário é o de conectar um equipamento extremamente sensível a uma rede extremamente ruidosa. A solução, embora nem sempre exequível é melhorar o suprimento de energia do computador e diminuir a susceptibilidade deste aos distúrbios da rede através do uso de equipamentos auxiliares de proteção

2.3. DIAGNÓSTICO E TESTE DE DISTÚRBIOS DA REDE EM MICROCOMPUTADORES

A maneira pela qual os distúrbios da rede afetam um determinado sistema de computação depende da natureza local dos distúrbios e das características do sistema. Um estudo completo

sobre o tema deve incluir as seguintes etapas [TRINDADE JR., 1987]:

(1)Caracterização através de medidas a longo prazo sobre a qualidade da rede elétrica nos diversos ambientes em que a máquina irá operar.

(2)Teste da vulnerabilidade das fontes de alimentação aos distúrbios elétricos caracterizados no passo 1.

(3) Pesquisa sobre a susceptibilidade do microcomputador aos distúrbios elétricos da fonte de alimentação.

(4) Proposta de soluções custo-efetivas para o problema.

Levantamentos de campo para avaliar a qualidade da rede em um determinado ambiente são geralmente iniciados com o objetivo de prevenir falhas em novos equipamentos a serem instalados ou explicar e corrigir falhas em equipamentos já existentes. Em qualquer um dos casos, o levantamento de campo sobre a qualidade da rede constitui uma das entradas do processo de decisão de prover ou não equipamento auxiliar de proteção e condicionamento da rede, antes ou depois que os distúrbios da rede tenham se tornado um problema.

Para uma caracterização completa da qualidade da rede elétrica, como sugerido em (1), diversos ambientes devem ser considerados e uma grande quantidade de dados deve ser coletada para se ter um tratamento estatístico adequado. A natureza dos distúrbios, assim como sua severidade e a taxa de incidência variam de local para local. Pode-se ter uma razoável avaliação dos distúrbios locais da rede ajustando alguns poucos dados coletados com informação sobre tipos de distúrbios de ambientes similares ao que se está analisando. Monitores de transientes que registram o nível e a frequência dos transientes são utilizados para essa finalidade.

A informação encontrada na literatura sobre a caracterização

dos ruídos de rede de acordo com o ambiente não é muito extensa. Dos trabalhos encontrados o mais abrangente é o de **MARTZLOF e GRUZS [1988]** que compilaram nove levantamentos de campo sobre distúrbios ocorridos na rede elétrica em diversos locais entre os anos 1970 e 1987. Desses levantamentos 4 cobriam todos os tipos de distúrbios e 5 cobriam apenas sobretensões transientes. O trabalho de Martzloff e Grusz é bastante completo pois trata de diversos pontos relevantes dos levantamentos de campo sobre a qualidade da rede, como a motivação, a interpretação dos tipos de distúrbios mais comuns, o equipamento normalmente utilizado e compara os dados compilados tanto do ponto de vista de resultados como de metodologia utilizada.

Um breve estudo para caracterização de alguns ambientes foi por nós realizado, sendo o resultado apresentado a seguir.

Dados sobre o desempenho de fontes de alimentação são poucos e usualmente não são publicados pelos fabricantes. É também inexistente na literatura, a informação sobre o comportamento de fontes de alimentação quando submetidas a distúrbios elétricos similares aos encontrados na rede. Para um estudo completo sobre o assunto seria necessário caracterizar cada tipo de fonte de alimentação verificando sua resposta quando submetida a distúrbios gerados por um sintetizador de transientes.

2.4. CARACTERIZAÇÃO DO RUÍDO DA REDE.

Uma caracterização completa do ruído da rede é de difícil execução, haja visto: o instrumental especializado necessário para as medidas, o tempo razoavelmente longo para que se possa colher um número considerável de dados e a diversidade de ambientes a

serem considerados.

O instrumento que deve ser utilizado para avaliar a qualidade da rede elétrica é o analisador de transientes. Esse equipamento pode assumir várias formas dependendo sua complexidade do número e quantidade de parâmetros que se deseja obter nas medidas. **MARTZLOFF e HAHN [1970]** desenvolveram um monitor de rede baseado em um osciloscópio modificado com uma câmara fotográfica adaptada e outro baseado em um contador eletromecânico que registrava transientes acima de uma determinada amplitude. Com esses monitores os autores analisaram diversos ambientes chegando a registrar surtos de tensão de amplitude até 2500 V em ambientes residenciais. Um outro monitor bastante mais complexo foi desenvolvido por **KAJIHARA [1972]**. Neste monitor o sinal da rede era continuamente armazenado em um tambor magnético e na ocorrência de um transiente o sinal registrado era passado para uma fita magnética para ser posteriormente analisado. **MARTZLOFF E GRUZS [1988]** apresentam no seu artigo os principais tipo de monitores de rede existentes, tecendo comentários sobre os mesmos. Fancher **[FANCHER, 1985]** apresenta os principais parâmetros medidos pelos modernos monitores de transientes de rede.

Para realizar um levantamento local sobre os distúrbios da rede, verificando principalmente picos e sobretensões, foi por mim desenvolvido um sistema simples cujo diagrama é mostrado na figura 2.4. O princípio deste monitor é a comparação dos níveis de tensão da rede com níveis de tensão obtidos a partir de uma tensão de referência. O circuito utilizado é constituido por um atenuador do sinal de rede, um filtro passa-alto com frequência de corte 375 Hz (para deixar passar sómente os transientes com componentes de frequência maior do que esse valor) e uma série de

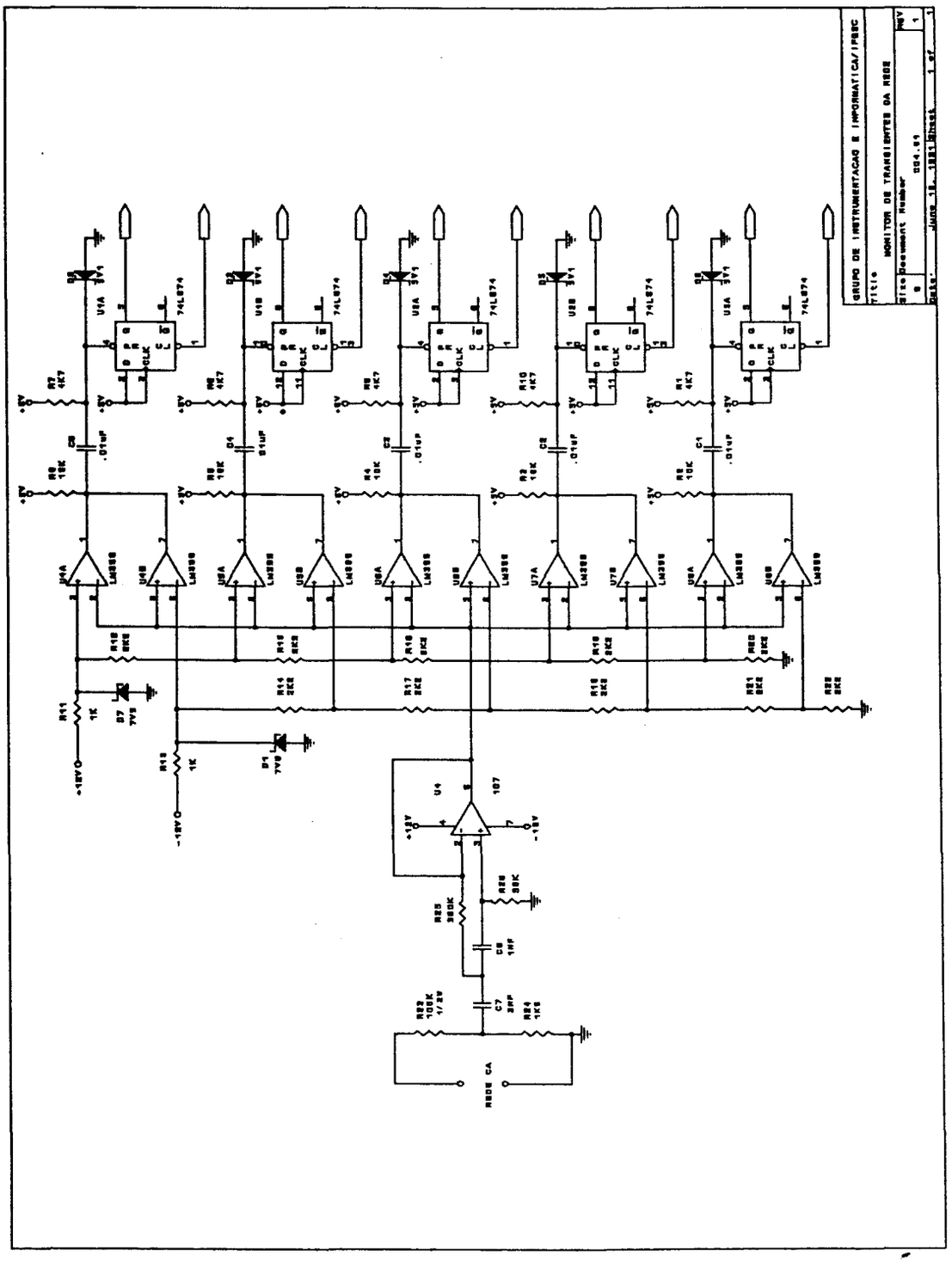


Figura 2.4: Monitor de transientes de rede.

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC
FÍSICA

comparadores que comparavam a saída do filtro com níveis de tensão pré-estabelecidos. As saídas dos comparadores foram ligadas a circuitos retentores que foram ligados a um microcomputador. As saidas dos retentores foram monitoradas pelo microcomputador e quando da ocorrência de um transiente este era registrado em um contador na memória. O circuito foi ajustado para detectar transientes nos seguintes níveis:

nível 1 - 152 a 302 V
nível 2 - 302 a 455 V
nível 3 - 455 a 605 V
nível 4 - 605 a 760 V
nível 5 - acima de 760 V

Para controle do monitor de transientes foi utilizado um microcomputador de 8 bits "Digibyte" e o programa desenvolvido para a monitoração permitia a contagem máxima de 9999 em qualquer um dos níveis, cessando a contagem quando atingia o valor máximo, também permitia a verificação da contagem de cada nível a qualquer instante.

Dois ambientes foram monitorados (ambos nas instalações do Departamento de Física e Ciência dos Materiais), o primeiro foi o laboratório do Grupo de instrumentação eletrônica e o segundo foi a Oficina Mecânica.

O primeiro ambiente foi monitorado durante 240 horas, tendo-se obtido os seguintes resultados:

40 transientes do nível 1 - 0.17 transientes/hora
6 transientes do nível 2 - 0.025 transientes/hora
1 transiente do nível 3 - 0.004 transientes/hora
70 transientes do nível 4 - 0.290 transientes/hora

5 transientes do nível 5 - 0.020 transientes/hora

Os valores acima foram obtidos através da contagem dos contadores tomando-se os valores 3 a 4 vezes por dia durante o período normal de trabalho. Durante o período de testes, quatro vezes o contador do nível 4 e uma vez o contador do nível 5 atingiram a contagem máxima (9999). Esses valores foram desconsiderados da tabela acima, tendo possívelmente sido gerados por ruído de alta frequência proveniente de centelhamento ao ligar alguma carga. Verificou-se também existir correlação entre o ligar determinadas cargas (lâmpadas fluorescentes, soldadores elétricos) e o registro de transientes.

O segundo ambiente foi monitorado durante 312 horas tendo-se obtido os seguintes resultados:

83 transientes do nível 1 - 0.266 transientes/hora
21 transientes do nível 2 - 0.067 transientes/hora
1 transiente do nível 3 - 0.003 transientes/hora
126 transientes do nível 4 - 0.400 transientes/hora
15 transientes do nível 5 - 0.048 transientes/hora

Durante o período monitorado, 14 vezes o contador do nível 4 e 38 vezes o contador do nível 5 atingiram a contagem máxima. Durante o mesmo período, o microcomputador deixou de operar 10 vezes devido aos distúrbios da rede.

Pelos dados obtidos, verifica-se que o segundo ambiente é extremamente mais ruidoso do que o primeiro, devido básicamente ao ligamento e desligamento de cargas com potência consideravel, como motores de máquinas operatrizes, aparelhos de solda a arco e um grande número de lâmpadas fluorescentes.

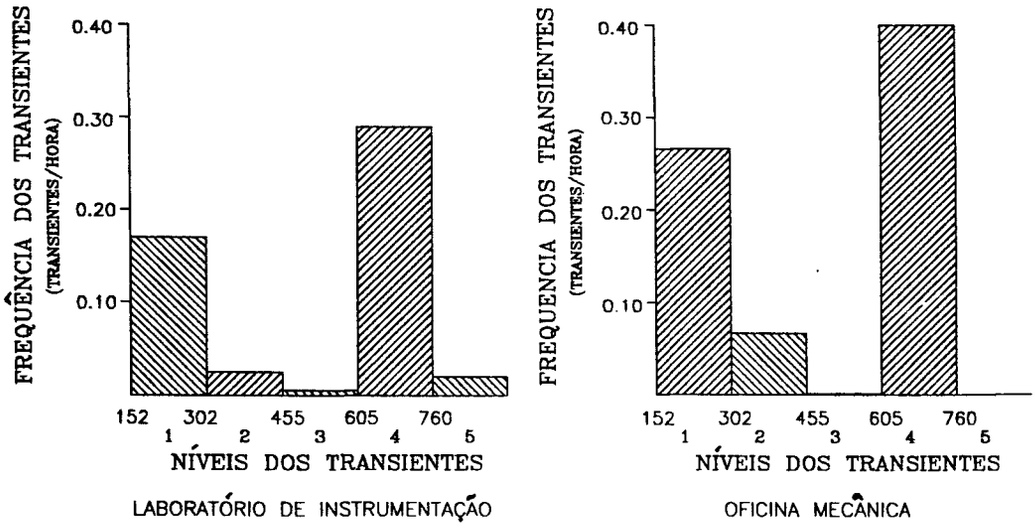


Figura 2.5: Frequência dos transientes observados no Departamento de Física e Ciência dos Materiais de São Carlos.

2.5. Monitor microprocessado de transientes da rede.

O monitor descrito na seção anterior é bastante limitado, registrando apenas sobretensões da rede dentro de uma estrita faixa de valores. Para poder caracterizar de maneira mais completa os distúrbios da rede, está sendo desenvolvido sob minha orientação um monitor de distúrbios da rede, controlado por um microprocessador de 16 bits (8086) **[SILVA, 1991]**. Os circuitos utilizados são de baixo consumo (CMOS) e o instrumento deverá operar com baterias para ficar independente da alimentação da rede. A arquitetura do monitor está ilustrada na figura 2.6 e consiste dos seguintes blocos funcionais:

Divisor de tensão - Consiste de um divisor resistivo para reduzir a tensão de entrada (CA da rede) do valor nominal para valores

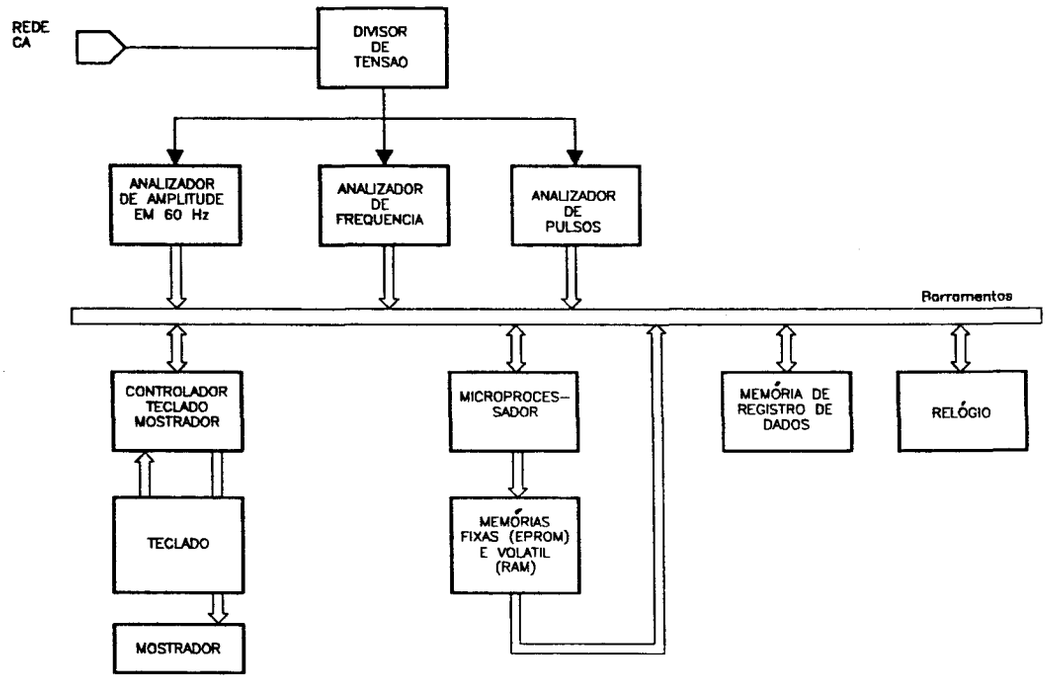


Figura 2.6: Diagrama de blocos monitor de transientes de rede microprocessado.

toleráveis pelos circuitos eletrônicos.

Analisador de amplitude em 60 Hz - A aquisição da amplitude em 60 Hz é feita de forma direta através de um conversor analógico digital de 8 bits. A tensão da rede será constantemente monitorada pelo microprocessador e quando houver uma situação de sub ou sobretensão (abaixo de 85% ou acima de 110% do valor nominal), o histórico da anomalia será registrado. Este circuito também deverá monitorar a tensão das baterias de alimentação e os valores de pico dos transientes.

Analisador de frequências - O analisador de frequências tem as funções de verificar se ocorrem desvios na frequência de 60 Hz e detectar a presença de interferências de rádio-frequência. Nos circuitos do analisador de frequência foram utilizados quatro filtros ativos: um para 60 Hz e tres para frequências nas faixas de 0,6 a 10Khz, 10Khz a 100Khz e 100Khz a 1Mhz. Os pulsos de saída dos filtros são contados por contadores binários.

Analisador de pulsos - O analisador de pulsos tem como função monitorar a duração, a frequência e a amplitude de transientes de curta duração (até 50 µs com picos de tensão iguais ou inferiores a 6KV.

Circuitos do microprocessador - Os circuitos do microprocessador ("hardware") consistem básicamente de um microprocessador 80C86-2 com os circuitos de apoio: gerador de pulsos de relógio, retentores para demultiplexação do barramento de dados e endereços e memórias permanentes ("EPROMS") com programas e voláteis ("RAMS"). A interface com o usuário é feita através de um teclado com teclas numéricas e de funções e por um visor de cristal líquido. O circuito foi construido com lógica de baixo consumo ("CMOS") sendo alimentado por duas fontes para tolerar falhas na alimentação.

Memória de registro de dados - Os parâmetros medidos são guardados em uma memória volátil de baixo consumo (CMOS) removível. Foi utilizada uma bateria de 3,6V para preservar a informação durante o transporte. Os dados coletados serão lidos e posteriormente processados por um microcomputador tipo IBM-PC através de uma interface adequada.

2.6. ENSAIOS SOBRE O EFEITO DE DISTÚRBIOS DA REDE ELÉTRICA EM MICROCOMPUTADORES.

A determinação da susceptibilidade de um equipamento aos distúrbios da rede elétrica pode ser obtida através da aplicação de distúrbios com características definidas e verificando-se o efeito desses distúrbios no equipamento sob teste. Uma vez aplicado o

distúrbio que seja capaz de provocar uma falha na operação do equipamento existem duas possibilidades básicas para verificar o efeito das falhas:

- Comparar o comportamento do equipamento com um comportamento padrão préviamente definido. Neste caso deve-se dispor de um padrão de excitações e respostas específico para o equipamento sob teste.

- Comparar o comportamento do equipamento com um equipamento considerado padrão, ambos executando simultaneamente a mesma função.

A segunda possibilidade caracteriza-se por uma maior independência em relação ao equipamento sob teste.

Nos dois métodos é necessário que os equipamentos envolvidos nos testes tenham todas as suas partes componentes, e sujeitas a falhas, exercitadas contínuamente. No caso de computadores, as excitações são providas por programas que exercitam as diferentes secções do mesmo. Os diversos sinais do computador devem ser continuamente monitorados durante a execução do programa para verificar se o funcionamento está correto.

No presente trabalho escolheu-se a segunda alternativa e a determinação do funcionamento correto do equipamento sob teste se faz por comparação com outro equipamento idêntico tomado como padrão. Os dois equipamentos operam síncronamente para efeito de comparação. Não é necessário, no caso, conhecer com exatidão as respostas do equipamento aos distúrbios aplicados e uma classe de distúrbios pode servir, com pequenas alterações, a toda uma classe de equipamentos. No caso específico de

computadores, tanto o computador sob teste como o tomado como padrão executam síncronamente o mesmo programa.

Baseado nas considerações acima foi desenvolvido um sistema de testes para verificar o comportamento de sistemas digitais quando submetidos a distúrbios da rede, o qual será apresentado a seguir.

2.7. SISTEMA DE TESTES PARA VERIFICAR O COMPORTAMENTO DE MICROCOMPUTADORES SUJEITOS A DISTÚRBIOS DA REDE ELÉTRICA.

O sistema de testes desenvolvido para testar a susceptibilidade de sistemas digitais a distúrbios da rede é mostrado no diagrama geral de blocos da figura 2.7. O sistema foi especialmente projetado para utilização com microcomputadores e outros equipamentos digitais utilizando microprocessadores. Isso permite testar uma classe bastante ampla de equipamentos .

Como requisito de teste considera-se que o microprocessador a ser testado disponha de uma entrada para acionamento de uma interrupção não mascarável (INT) e uma entrada que permita colocá-lo em estado de espera (ESP).

O sistema de testes utiliza duas réplicas idênticas do equipamento, uma que será submetida aos distúrbios e outra tomada como padrão. As réplicas utilizadas serão a partir de agora chamadas módulo 1 e módulo 2. O módulo 1 é chamado padrão enquanto o módulo 2 é o que sofre a influência dos distúrbios.

Durante os testes, os módulos 1 e 2 executam síncronamente o mesmo programa . Este sincronismo dos módulos é conseguido pela utilização de circuitos comuns para geração do sinal

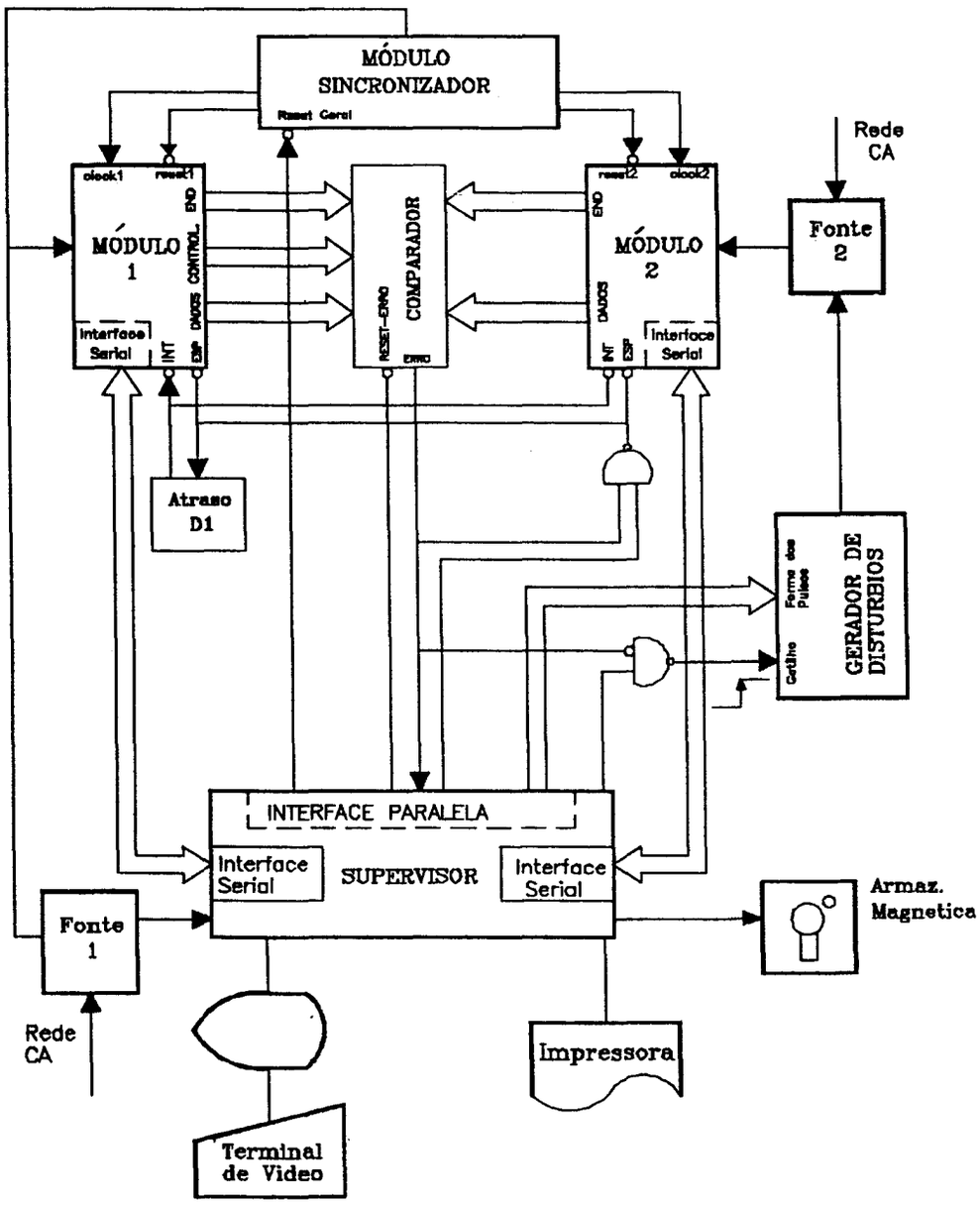


Figura 2.7: Diagrama de blocos do sistema de testes sobre o efeito de distúrbios da rede elétrica.

de base de tempos ("clock") e reinicialização ("reset"). Esses circuitos estão contidos no bloco sincronizador. O sincronismo exato dos módulos sob teste é importante em consequência do método utilizado na detecção das falhas.

A detecção de diferenças no funcionamento dos módulos sob

teste é feita através de um circuito comparador que compara os principais sinais desses módulos. No caso de equipamentos com microprocessadores é feita a comparação dos barramentos de dados e endereços. A comparação dos barramentos é a única forma viável para a detecção de alterações no funcionamento de microprocessadores, haja visto a impossibilidade de acesso direto aos subsistemas lógicos internos destes.

A ocorrência de uma diferença de funcionamento entre os módulos é informada a um computador supervisor. O supervisor controla através de portas paralelas todo o sistema de testes provendo os sinais para reinicialização utilizados na sincronização dos módulos sob teste e comparador. O supervisor recebe do comparador o sinal de ERRO que indica qualquer diferença constatada entre os barramentos do módulo sob teste. A forma e a aplicação dos distúrbios aos quais está sujeito o módulo 2 também são controladas pelo supervisor.

No caso de sistemas com microcomputadores, a troca de dados entre os módulos sob teste e o supervisor, após ter sido detectada uma falha entre os barramentos dos módulos, é importante para se determinar a ocorrência de uma falha.

O sistema de testes é controlado por um microcomputador supervisor que fornece estatísticas sobre as falhas detectadas. Ele viabiliza também a implantação de uma série de funções adicionais do sistema de testes. Entre essas, cita-se a possibilidade de identificar a natureza da falha e a troca dos programas executados nos módulos em teste.

Ao supervisor cabe ainda o controle do gerador de distúrbios para gerar sequências pré estabelecidas de distúrbios ao microcomputador sob teste. É possível nessa configuração, controlar as características dos distúrbios gerados através de um programa executado no bloco supervisor.

O supervisor pode emitir relatórios sobre as falhas no sistema em teste que poderão permitir estudo de técnicas para contrapor essas falhas.

2.8. MECANISMO DE DETECÇÃO E IDENTIFICAÇÃO DAS FALHAS.

Para a detecção e identificação das falhas é realizada uma comparação dos barramentos e uma coleta de dados dos módulos sob teste. A confirmação da falha é feita fazendo-se a comparação do conteúdo da memória de leitura e escrita e dos registradores internos dos dois módulos sob teste. Esta comparação é feita pelo supervisor após os dados terem sido coletados através das portas de comunicação serial. Através desta comparação consegue-se determinar a natureza da falha. De uma forma geral, as falhas são motivadas por uma alteração no barramento de dados ou de endereços, por algum dos circuitos conectados a esses barramentos. Esta última afirmação somente é valida, se o único meio de interferência considerado para os distúrbios forem as linhas de alimentação dos circuitos onde eles foram aplicados.

A comparação dos barramentos deve ser síncrona, ou seja os barramentos devem ser comparados nos instantes em que eles contém informações válidas. A comparação assíncrona, embora conceitualmente mais simples e imediata, é de dificil implementação devido aos ruídos de curta duração ("glitches") normalmente presentes nos barramentos e devido também ao fato que alguns circuitos geram ruídos adicionais de maior duração quando os barramentos se encontram no terceiro estado ("tri state").

Um comparador síncrono não sofre dos inconvenientes apontados para os assíncronos e por essa razão foi adotado neste

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFUSP
FÍSICA

trabalho. Nesse tipo de comparador deve existir uma lógica que indique os instantes em que deve ser efetuada a comparação, isto é, os instantes nos quais existem dados ou endereços válidos nos barramentos. Esse circuito utiliza os sinais do barramento de controle do módulo não sujeito à ação dos distúrbios aplicados (módulo 1) para determinar os instantes em que devem ser efetuadas as comparações. É evidente que um comparador desse tipo torna-se específico para o microprocessador para o qual foi projetado, ou quando muito, para uma família de microprocessadores de um determinado fabricante. A filosofia de projeto permanece entretanto inalterada para os diversos processadores existentes.

A transferência do conteúdo dos registradores internos e da memória dos módulos para o supervisor é feita através de portas seriais RS-232C. A adoção desse padrão facilita a implementação da comunicação entre o supervisor e eventuais equipamentos comerciais sob teste, haja visto que portas seriais são comumente encontradas nesses equipamentos.

A detecção e identificação de uma falha inicia-se quando o comparador detecta uma diferença entre os barramentos de dados e de endereços dos módulos 1 e 2, o que é indicado pelo sinal "Erro". Esse sinal inibe a geração de novos distúrbios, faz a solicitação de estados de espera ("ESP") aos módulos e informa o supervisor da ocorrência de uma possível falha. Após um periodo de tempo t, contado a partir da geração do sinal "ESP", é feita uma solicitação de interrupção não mascarável "INT" aos módulos. A rotina de atendimento dessa interrupção cuida então da transferência dos dados das memórias e registradores internos para o supervisor, os dados dos módulos são comparados e um relatório com as diferenças encontradas é elaborado e impresso.

O mecanismo descrito foi implementado visando minimizar

o número de instruções executadas pelo processador, após a ocorrência de uma falha. Isso evita que a falha se propague tornando difícil a determinação de sua origem. Outros mecanismos são possíveis e podem ser mais simples, se não for necessário conhecer a origem da falha. Pode-se também rastrear o efeito de uma falha implementando-se um circuito que permita a execução de um número pré determinado de instruções após a detecção de uma diferença nos barramentos. Outra forma consiste na execução de rotinas que permitam reiniciar o processamento dos módulos a partir do estado em que ele foi interrompido pela execução da interrupção não mascarável.

Toda uma sequência de testes pode ser pré-programada no supervisor e após a emissão dos relatórios relativos a uma falha, pode-se tentar uma nova sincronização para que a sequência de testes seja terminada ou repetida. Esta automatização do plano de testes abre a possibilidade de sequências longas de testes, sem a interferência de um operador. Nesse último caso deve-se, entretanto, prever mecanismos de proteção para desligamento do sistema no caso de testes mais severos que possam implicar na destruição dos circuitos.

Na figura 2.8 mostramos um fluxograma simplificado de uma sequência automática de testes controlada pelo supervisor.

2.9. DESCRIÇÃO DOS CIRCUITOS IMPLEMENTADOS

Na definição dos circuitos utilizados para os testes de susceptibilidade, procurou-se observar, dentro do possível, a sua independência em relação às características dos módulos sob teste. Esse cuidado é importante no sentido de minimizar as modificações do sistema, para o teste de um equipamento diferente daquele que

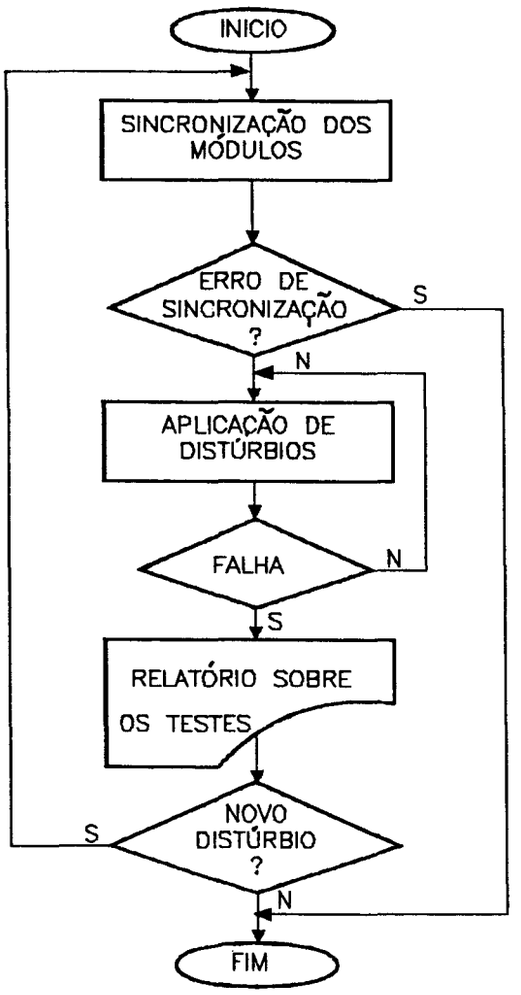


Figura 2.8: Diagrama simplificado do programa executado no supervisor durante os ensaios.

vinha sido testado. A seguir descreveremos a nível de diagrama de blocos os circuitos construídos.

2.9.1. Microcomputador supervisor.

O diagrama de blocos do supervisor é mostrado na figura 2.9. Foi utilizado neste trabalho um microcomputador desenvolvido pelo laboratório de instrumentação eletrônica do DFCM (micro LIE - UCP64D), com as seguintes características:

- Microprocessador Z80A - com base de tempos 4 Mhz.
- 66 Kbytes de memória de escrita e leitura ("RAM") e 4

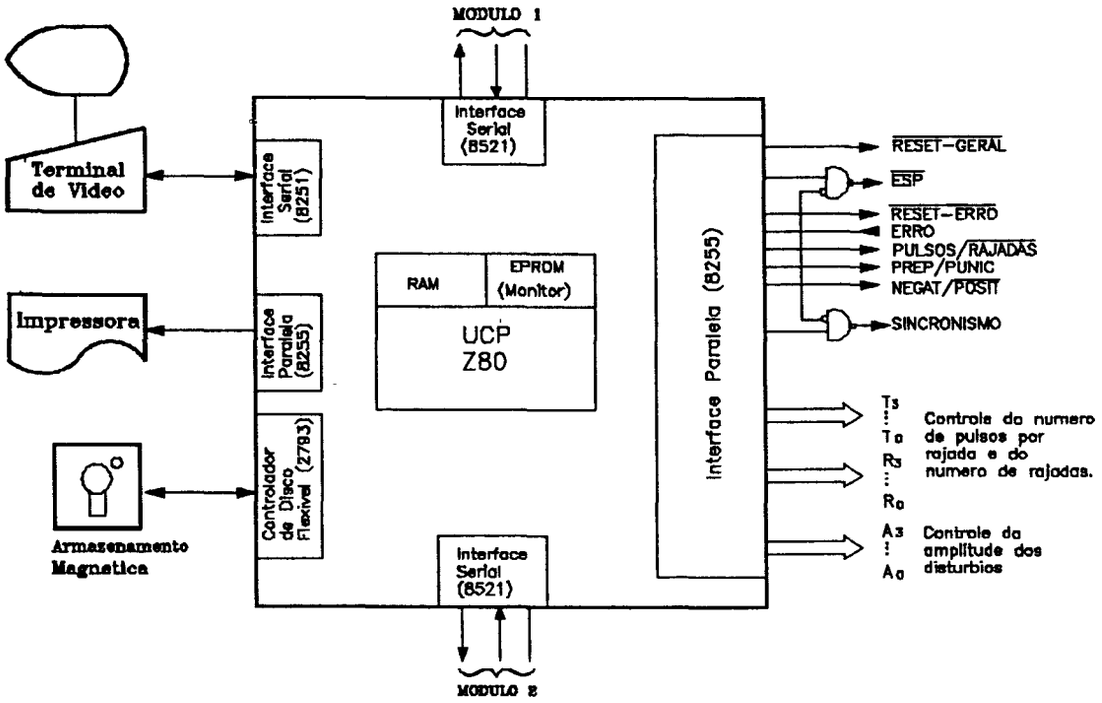


Figura 2.9: -Diagrama de blocos do módulo supervisor.

kbytes de memória permanente ("EPROM").

- Teclado e monitor de vídeo.
- Unidade de disco flexível de 5 1/4", Dupla face - Dupla densidade (360 K).
- 1 Porta serial RS-232C
- 1 Impressora paralela

Foi utilizado o sistema operacional CPM para controlar o sistema.

Foram ainda acrescentados ao microcomputador para poder controlar o sistema de testes 2 portas seriais e 3 portas paralelas de 8 bits (Intel 8255). As portas seriais foram utilizadas para comunicação com os módulos sob teste e as portas paralelas foram utilizadas para controle dos outros módulos do sistema.

2.9.2. Os módulos sob teste.

O diagrama em blocos dos módulos dos microcomputadores utilizados para os testes de susceptibilidade (módulos sob teste) é apresentada na figura 2.10.

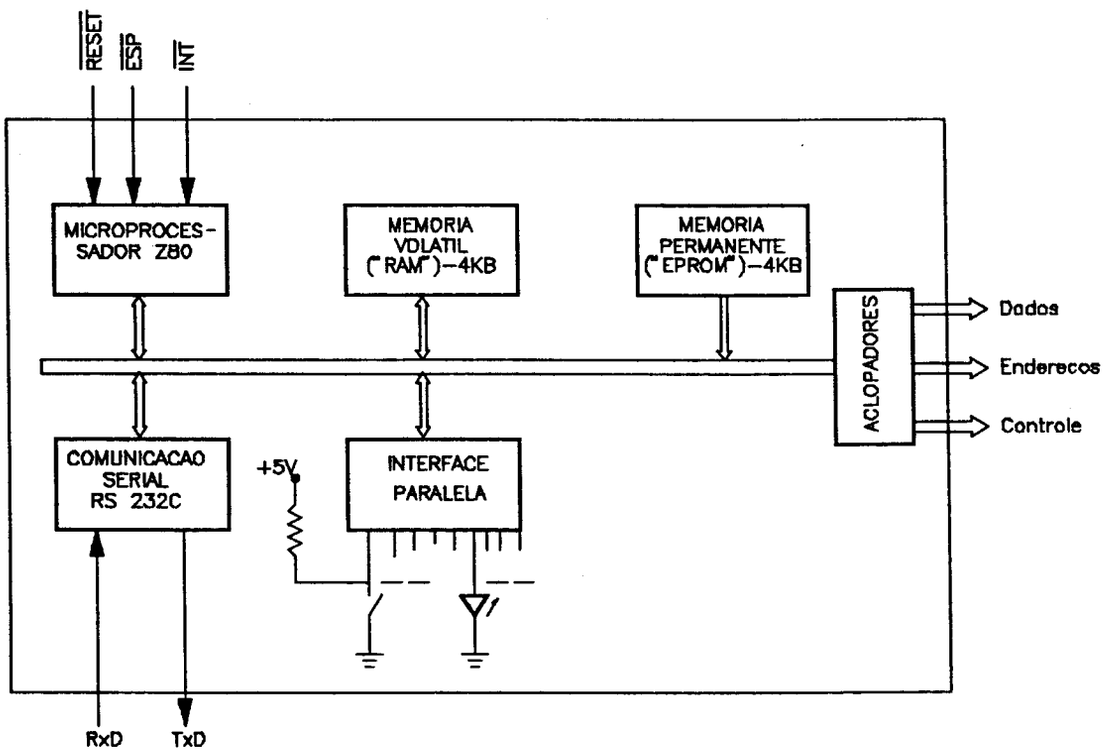


Figura 2.10: Diagrama de blocos simplificado dos módulos sob teste.

Os módulos sob teste foram baseados no microprocessador de 8 bits Z80A, foram utilizadas duas memórias permanentes de 2 Kbytes (2716), duas memórias de escrita e leitura estáticas (6116), uma porta de comunicação serial (8251 A) e duas portas de comunicação paralela (Z80 - PIO).

As portas paralelas foram utilizadas para simular sinais de entrada e saída, sendo uma delas ligada a oito chaves bipolares e a outra ligada a oito diodos emissores de luz.

A taxa de transferência para a comunicação serial foi gerada a partir de um sinal de relógio e fixada em 4800 bauds. A módulo

sob teste recebe do sincronizador os sinais de reinicialização e relógio, os quais são comuns aos dois módulos sob teste. Os barramentos de dados, endereços e controle de cada módulo , são disponíveis ao comparador através de acopladores ("buffers").

2.9.3. Comparador.

O comparador detecta diferenças entre os barramentos de dados e endereços dos módulos sob teste. O diagrama de blocos do comparador é mostrado na figura 2.11.

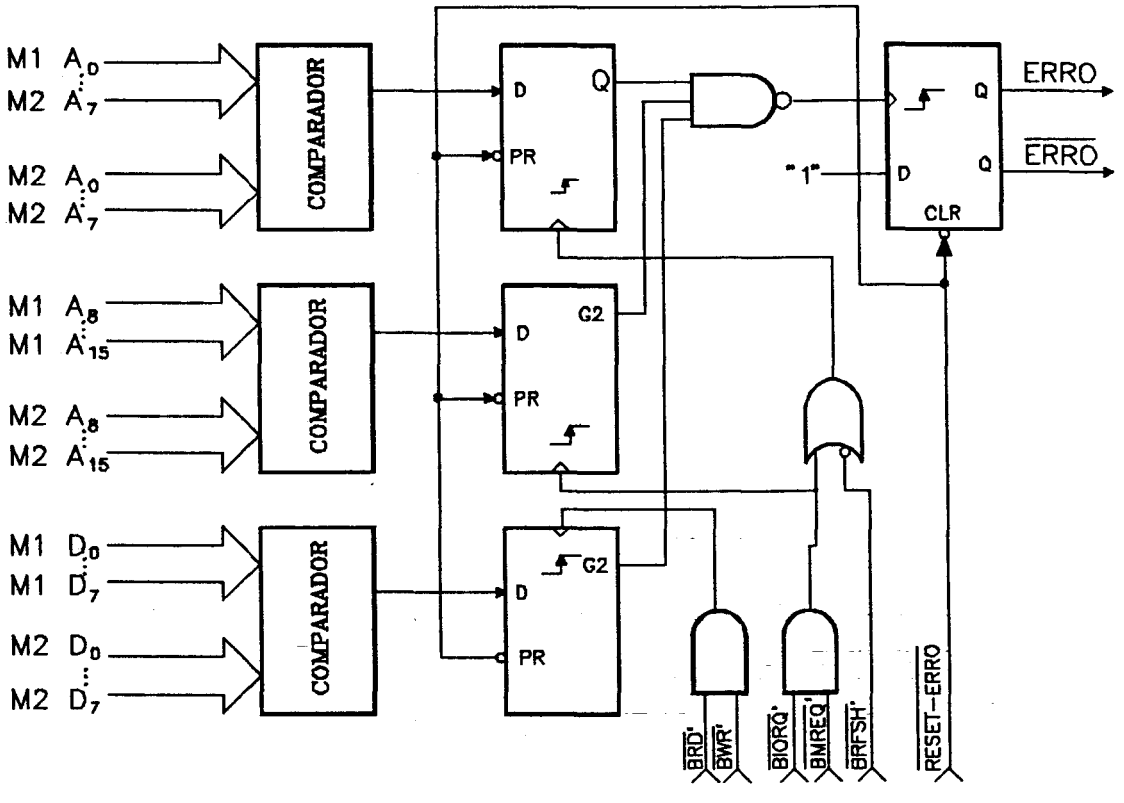


Figura 2.11: Diagrama de blocos do circuito comparador.

A comparação dos barramentos é feita linha a linha, utilizando-se portas-coincidência com saídas em coletor aberto, tendo sido

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC
FÍSICA

utilizadas 24 portas para essa comparação. As saídas das portas foram agrupadas em tres grupos: Endereços A_1 .... A_6, endereços A_7 ... A_{15} e dados D_0 ... D_7, cada grupo dando origem a uma linha que indica diferenças entre os grupos de sinais. Essas linhas são conectadas a entradas de retentores ("flip-flops") do tipo D, que armazenam o conteúdo delas quando ocorre uma transição positiva nas entradas de relógio. A saída de qualquer um desses retentores pode acionar um quarto retentor, que indica, finalmente, a ocorrência de uma diferença entre os barramentos de dados e endereços dos módulos sob teste no instante em que houve o armazenamento de informação na saída das portas comparadoras.

Para cada um dos tres grupos de sinais comparados, existe um retentor que é acionado em sincronismo com sinais do barramento de controle do módulo não sujeito aos distúrbios aplicados (módulo 1). Dessa forma, uma diferença entre os barramentos de dados ocasiona um sinal ERRO do comparador, caso ela ocorra na subida do sinal de leitura ou do sinal de escrita do microprocessador do módulo 1. Similarmente, diferenças nos barramentos de endereços são consideradas nas subidas dos sinais de acesso à memória, ($\overline{MREQ'}$), ou a dispositivos de entrada e saída ($\overline{IORQ'}$), do mesmo processador. O barramento de endereços doi dividido, para efeito de comparação, devido aos endereços para refrescamento de memórias dinâmicas que a UCP Z80 coloca nas linhas A_0 ..... A_6. Como alguns fabricantes não especificam o estado das outras linhas de endereço durante o ciclo de refrescamento, foi inibida a comparação dessas linhas nessa condição.

2.9.4. Gerador de distúrbios

O gerador de distúrbios é o módulo responsável pela geração dos distúrbios que são aplicados aos módulos sob teste. Os

possíveis distúrbios que podem ser gerados por este circuito são: pulsos isolados, pulsos repetidos e rajadas de pulsos ("burst") com duração e taxas de repetição programáveis.

O gerador de distúrbios foi dividido em tres blocos funcionais: gerador de pulsos, fonte programável de tensão e circuito de chaveamento dos pulsos conforme o diagrama de blocos da figura 2.12.

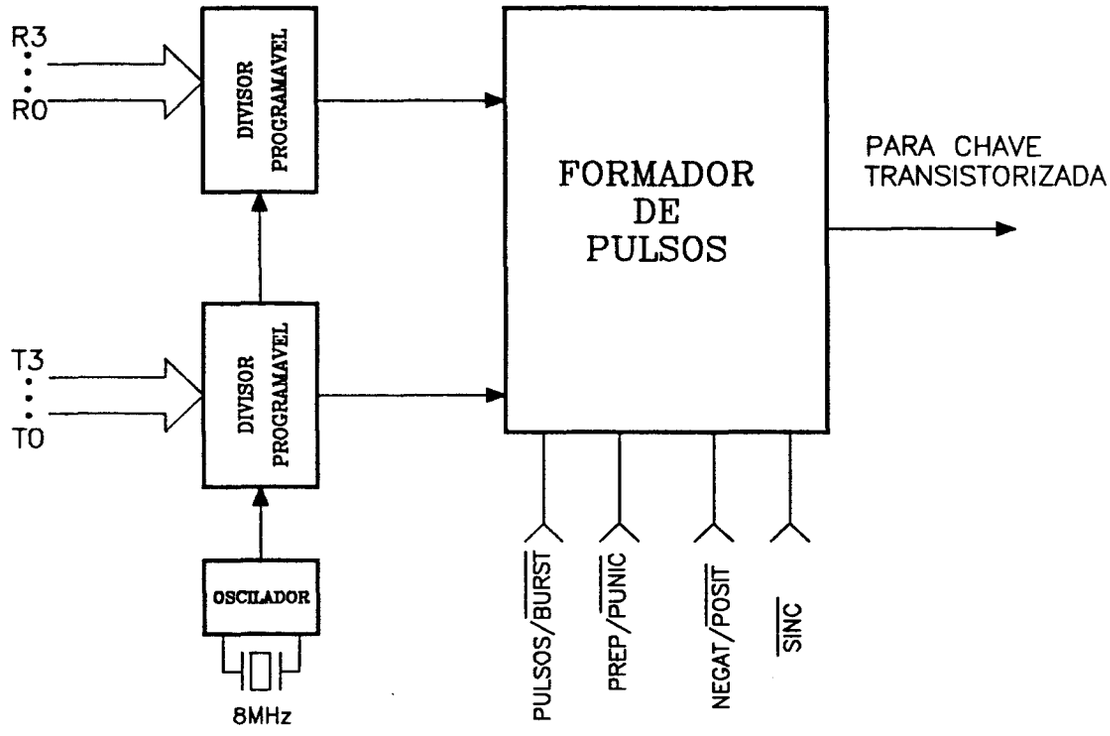


Figura 2.12: Diagrama de blocos do gerador de distúrbios.

Os períodos e as taxas de repetição dos pulsos são derivados de um oscilador a cristal e a forma dos pulsos é controlada por sinais comandados pelo supervisor. Os sinais do gerador de pulsos controlam a aplicação de um valor de tensão programável nas linhas de alimentação de um dos módulos sob teste, através de uma chave eletrônica. A forma de onda dos pulsos está ilustrada

na figura 2.13.

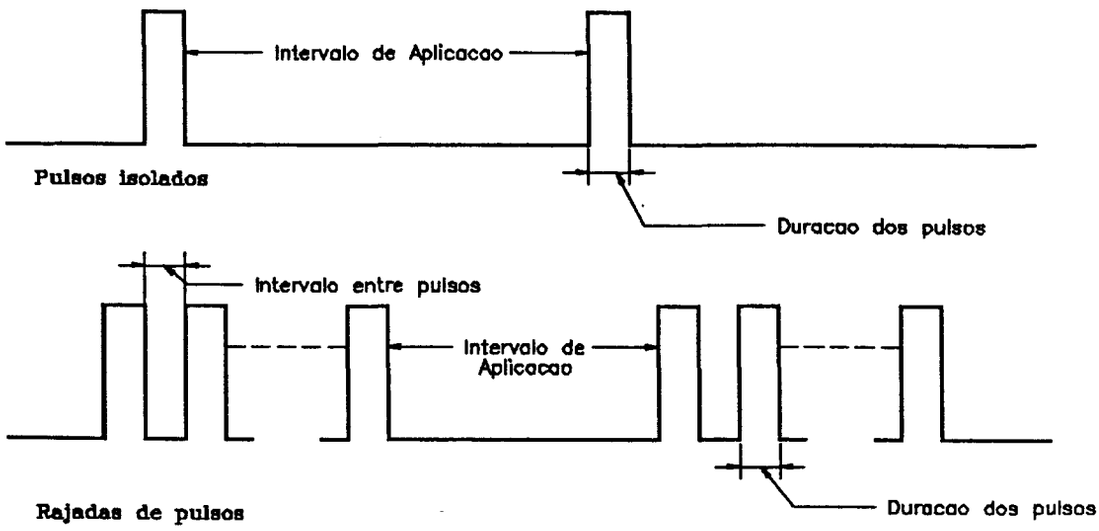


Figura 2.13: Forma de onda dos distúrbios aplicados.

Como fonte de tensão programável foi utilizada uma fonte de corrente contínua marca TEKTROL mod. TCA 120-05. Essa fonte pode fornecer tensões contínuas entre 0 e 125 volts com correntes de até 5 amperes. O limitador de corrente foi ajustado para a máxima corrente, isto é, 5 A. A tensão de saída da fonte foi programada pela adição de resistores externos comandados através de relês pelo supervisor. O circuito foi projetado de maneira que a tensão de saída da fonte pudesse ser programada entre 0 e 75 volts, em passos de 5 volts. Na implementação utilizou-se resistores variáveis, permitindo que outros valores para os degraus de tensão possam ser selecionados.

Para que a tensão obtida na fonte pudesse ser aplicada de forma controlada nas linhas de alimentação do módulo sujeito aos distúrbios, foi utilizada uma chave eletrônica de potência construída com transistor do tipo VMOS. O acoplamento dos distúrbios com a linha de alimentação do equipamento sob teste foi realizada de maneira direta e em paralelo com a fonte de alimentação do equipamento. Foi prevista também a possibilidade

de inversão da polaridade dos distúrbios.

2.10. DESCRIÇÃO DOS PROGRAMAS UTILIZADOS.

Para tornar operacional o sistema de testes foram desenvolvidos alguns programas utilizando a linguagem montadora do microprocessador Z80. Dois blocos de rotinas foram desenvolvidos, um para o módulo supervisor e outro para os módulos sob teste. A seguir são descritos os programas desenvolvidos ressaltando a função desempenhada por cada um deles no sistema.

2.10.1. Rotinas do Supervisor.

A operação do sistema de testes é controlada por um conjunto de rotinas que são executadas pelo microcomputador supervisor. Cabe a essas rotinas:

- Sincronizar a execução das rotinas de testes dos módulos.
- Controlar a aplicação de uma sequência de distúrbios com características programáveis e pré definidas.
- Controlar a transferência dos dados das memórias voláteis e dos registradores dos módulos sob teste para a memória local do supervisor.
- Comparar os dados transferidos emitindo um relatório sobre as discrepâncias constatadas.
- Resincronizar os módulos sob teste, alertando o operador caso essa resincronização não seja possível.

2.10.2. Rotinas dos módulos sob teste.

Os módulos sob teste tem um programa monitor residente em memória permanente do tipo "EPROM". A execução do programa monitor depende de condição pré definida das chaves ligadas às portas paralelas, se esta condição não for encontrada o programa monitor não é executado. O programa monitor foi utilizado nas fases de desenvolvimento e depuração dos módulos sob teste.

Assim que o módulo sob teste entra em operação é executada a rotina de inicialização (caso não esteja executando o programa monitor) que inicializa os dispositivos de entrada/saída e preenchem com um valor constante (55H) as memórias voláteis.

Após a inicialização dos módulos, a rotina de teste, também residente na memória permanente, é transferida para a memória volátil e então executada. A rotina de testes executa um laço infinito dentro do qual são exercitados, para escrita e leitura, a área de dados da memória volatil e as portas paralelas (onde estão ligadas as chaves e os diodos emissores de luz).

A rotina de testes só deixa de ser executada quando a CPU recebe uma interrupção não mascarável, resultante de uma diferença entre os barramentos dos módulos sob teste, detectada pelo comparador. Quando isto ocorre, o programa é desviado para uma rotina de atendimento da interrupção que executa as seguintes funções: salva na memória volátil o conteúdo dos registradores da UCP, transfere mediante um comando do supervisor via porta serial os conteúdos das memórias voláteis, aguarda um comando do supervisor para reinicialização dos módulos.

2.11. ENSAIOS SOBRE A SUSCEPTIBILIDADE AOS DISTÚRBIOS DA REDE.

2.11.1. Considerações gerais sobre os distúrbios aplicados.

Os distúrbios aplicados durante os testes foram pulsos isolados ou rajadas de pulsos (conforme a Figura 2.13), sobrepostos à tensão de alimentação dos módulos sob teste. As tensões de aplicação dos pulsos variaram entre 5V e 75V em degraus de 5V, acima da tensão nominal de alimentação (5V) no caso dos pulsos positivos e de -2V a -10V em degraus de -2V, abaixo da tensão nominal de alimentação, no caso dos pulsos negativos.

Devido à limitação de corrente da fonte de alimentação utilizada para gerar os distúrbios, ocorreu um amortecimento significativo dos distúrbios de curta duração pelos capacitores de desacoplamento.

Em um dos ensaios realizados foram retirados todos os circuitos integrados de um dos módulos sob teste e foram aplicados distúrbios com amplitude fixa de 75V e largura variável. A forma de onda observada está ilustrada na Figura 2.14. Os valores de tensão de pico foram medidos e estão indicados na tabela 2.1.

Tabela 2.1: Redução da amplitude dos pulsos devido aos capacitores de desacoplamento.

Largura dos distúrbios (μseg)	Tensão de pico medida (V)
1,2	4,0
2,4	6,0
4,6	10,0
8,0	23,0
14,0	32,0
28,0	58,0
72,0	72,0

Podemos observar que devido ao alto conteúdo harmônico dos pulsos de curta duração, estes são fortemente absorvidos pelos capacitores de desacoplamento. Esse fato reforça a importância do

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC
FÍSICA

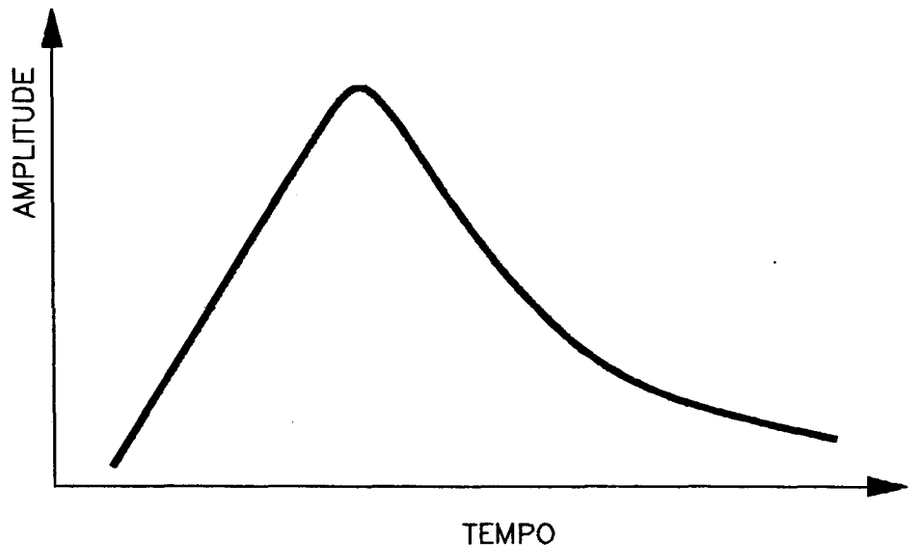


Figura 2.14 Forma de onda dos distúrbios amortecidos pelos capacitores de desacoplamento.

uso desses capacitores na eliminação de distúrbios de curta duração.

Sete sequências de distúrbios foram aplicadas de acordo com a tabela 2.2. As duas primeiras sequências aplicadas foram de pulsos isolados positivos ordenados por energia crescente para evitar a queima indiscriminada de componentes. As sequências de 3 a 6 foram também de pulsos positivos (e rajadas de pulsos) e os distúrbios foram ordenados fixando-se a duração (entre 1,2μseg e 260μseg) e variando-se a amplitude (entre 5 e 75V). Na última sequência foram aplicados pulsos negativos, também ordenados por largura e amplitude crescentes. As durações dos distúrbios (em μseg) foram: 1,2; 2,4; 4,6; 8; 14; 28; 72; 132 e 260.

2.11.2. Resultados dos ensaios.

Os resultados obtidos pela aplicação das sequências de teste estão apresentados na tabela 2.3. Algumas das sequências foram

Tabela 2.2: Características das sequências de teste.

Sequência de teste	Tipo dos Distúrbios	Polaridade	Nº de Aplicações de cada Distúrbio	Intervalo de Aplicação	Ordem de Aplicação
1	Pulsos isolados	Positiva	10	10 seg	Energia Crescente
2	Pulsos isolados	Positiva	50	15 seg	Energia Crescente
3	Pulsos isolados	Positiva	10	10 seg	Amplitude e duração Crescentes
4	Rajada de pulsos 4 pulsos separados 50 µseg	Positiva	10	10 seg	Amplitude e duração Crescentes
5	Rajada de pulsos 16 pulsos separados 50 µseg	Positiva	10	10 seg	Amplitude e duração Crescentes
6	Rajada de pulsos 16 pulsos separados 15 µseg	Positiva	10	10 seg	Amplitude e duração Crescentes
7	Pulsos isolados	Negativa	10	10 seg	Amplitude e duração Crescentes

aplicadas mais de uma vez, como mostrado na coluna "Teste".

Todas as sequências de teste às quais foi submetido o módulo 2 não foram completadas. Dois tipos de falhas de componentes foram observadas: queima de componentes e travamento de memórias. Na tabela 2.3 pode também ser observada a duração e a amplitude do último distúrbio aplicado.

As queimas de memórias foram mais frequentes na etapa de depuração e validação do sistema. Para evitar a queima indiscriminada de componentes foram tomados cuidados na ordenação e no intervalo entre aplicações dos distúrbios.

As memórias utilizadas nos módulos sob teste são de tecnologia CMOS. Nestes dispositivos, o substrato de silício forma junções P-N com os elementos do circuito CMOS criando um retificador controlado de silício ("SCR - silicon controlled rectifier") parasita. Este "SCR" pode ser disparado em determinadas circunstâncias, no nosso caso através dos distúrbios aplicados, resultando em um aumento muito grande de corrente no

Tabela 2.3: Resultados da aplicação dos distúrbios no microprocessador sob teste.

Sequência de testes	Teste	Classe de falha identificada A	B	C	D	Total de falhas detectadas	Ultimo distúrbio aplicado Duração	Amplitude	Número total de distúrbios aplicados
	1	00	01	00	01	02	28µseg	10V	261
	2	00	01	00	00	01	28µseg	10V	261
1	3	00	01	00	00	01	14µseg	10V	171
	4	00	01	00	00	01	14µseg	10V	171
	5	00	02	00	00	02	14µseg	15V	301
2	1	01	00	01	03	05	28µseg	10V	1301
3	1	00	10	00	02	12	8µseg	35V	511
	2	00	03	02	02	07	8µseg	70V	581
4	1	00	05	00	08	13	8µseg	35V	513
5	1	00	05	00	21	26	8µseg	35V	513
6	1	00	02	00	41	43	4.6µseg	60V	411
7	1	27	00	00	12	39	2.4µseg	- 8V	81

dispositivo. Esse fenômeno é o travamento de memórias que mencionamos, denominado na literatura de "latch-up" [PAWLOSKI ET AL., 1985; HOROWITZ e HILL, 1989]. A manifestação desse efeito se dá por um curto circuito da fonte de alimentação do módulo sob teste e um aquecimento pronunciado da memória. Muitas das memórias que apresentaram esse efeito quando socorridas a tempo, com a fonte de alimentação desligada e esperando um tempo de resfriamento, voltaram a funcionar.

As seguintes classes de diferenças, detectadas entre os módulos, foram identificadas:

A - Alteração dos conteúdos das RAMs ou do conteúdo dos registradores da CPU.

B - Travamento de memórias ("latch-up")

C - Destruição (queima) de componentes semicondutores.

D - Falhas por razões não identificadas.

As falhas de classe A estão relacionadas com modificações nas áreas de armazenamento volátil. Nesta situação os distúrbios afetam diretamente ou os barramentos ou os dispositivos.

As falhas de classe B estão relacionadas com o travamento de memórias. Externamente observou-se aquecimento dos componentes e curto-circuito na fonte de alimentação.

As falhas de classe C estão relacionadas com a destruição (queima) de dispositivos semicondutores. A maioria dos componentes queimados utilizavam tecnologia MOS.

Falhas de classe D são discordâncias detectadas pelo comparador que não afetam o processamento dos módulos sob teste. Devemos observar que os barramentos dos módulos sob teste de alguma maneira foram afetados, embora aparentemente não se tenha observado nenhuma falha.

Analisando a tabela 2.3 verificamos que quando da aplicação de distúrbios positivos, as falhas que mais provocaram discordâncias no comparador foram motivos não identificados e travamento de memórias com 1,56% e 0,62% das discordâncias detectadas respectivamente.

A distribuição das falhas detectadas para os pulsos positivos é a seguinte:

Alteração no conteúdo do armazenamento volátil	0,9%
Travamento de memórias ("latch-up")	27,3%

Destruição de componentes semicondutores	2,7%
Motivos não identificados	69,1%

Excluindo as falhas não identificadas, a maior parte das falhas (30%) causou problemas nos circuitos. Verificamos também que as rajadas de pulsos (sequências 4, 5 e 6) causaram mais problemas do que os pulsos isolados. Também é importante observar que durante os ensaios foram atingidos valores de tensão consideravelmente maiores do que o valor nominal da fonte de alimentação, pelo menos 2 vezes e chegando a ser 14 vezes maior. Isto pode também ser visto na tabela 4 pelo registro do último distúrbio aplicado.

No caso dos distúrbios negativos o efeito dos distúrbios foi consideravelmente diferente. A maior parte dos distúrbios que causaram problemas o fizeram pela alteração dos dados nas memórias ou nos registradores, seguidas por execução incorreta dos programas. Níveis baixos de energia foram suficientes para interromper a operação do microcomputador. Em nenhum caso os distúrbios negativos provocaram problemas de circuito no microcomputador.

O ensaio com pulsos negativos revalida o trabalho de **CORTES e Mc CLUSKEY [1986]** sobre o efeito de distúrbios da alimentação em circuitos CMOS. Neste trabalho CORTES mostrou que os distúrbios negativos da alimentação causam problemas de temporização em circuitos digitais, causando atrasos na propagação.

2.12. PROPOSTAS PARA MINIMIZAR O EFEITO

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC
FÍSICA

DOS DISTÚRBIOS EM EQUIPAMENTOS COM MICROPROCESSADORES.

Resumindo os efeitos dos distúrbios da rede em sistemas com microprocessadores, vimos que: os distúrbios positivos causam problemas a nível de circuitos induzindo frequentemente a queima de componentes e ao travamento de memórias e outros dispositivos de tecnologia CMOS; já os distúrbios negativos afetam a função de processamento alterando o fluxo dos programas, podendo levar a erros de programação e até mesmo à interrupção da operação do sistema.

Algumas soluções simples podem introduzir uma melhora considerável em sistemas com microprocessadores frente aos distúrbios da rede. A utilização de capacitores de desacoplamento pode reduzir a amplitude dos distúrbios de curta duração a níveis aceitáveis. A utilização correta de filtros de linha e supressores de transientes (por exemplo varistores) pode introduzir melhoras frente aos distúrbios positivos [MARTZLOFF, 1987]. Para os distúrbios negativos, de curta duração, uma solução simples consiste em aumentar a energia armazenada nas fontes de alimentação utilizando capacitores maiores.

Soluções mais completas para o problema podem envolver custos muito maiores. A nível de condicionamento da rede podem ser utilizados transformadores de isolação, reguladores automáticos de tensão e fontes ininterruptíveis de tensão [CLARK, 1985; CROSTHWAIT, 1985].

Para aplicações críticas, técnicas de tolerância a falhas tanto a nível de circuitos como de programação devem ser utilizadas. Algumas destas técnicas são descritas no artigo de GERKE e KIMMEL [1986].

A susceptibilidade de sistemas aos distúrbios da rede está

diretamente relacionada com o consumo de potência. Quanto menor a potência consumida mais efetivos e de menor custo são os esquemas elétricos e eletrônicos de proteção. Existe uma tendência, com a evolução da tecnologia, de se reduzir a potência consumida pelos dispositivos semicondutores, por exemplo o consumo de potência do microprocessador de 8 bits 8085 é de 1,5W, do microcomputador de 16 bits 8086 é de 2,5W, já o do 80C86 (8086 com tecnologia CMOS) é de apenas 1W. Essa tendência de redução do consumo dos dispositívos é contraposta, entretanto, pelo desenvolvimento de dispositivos mais rápidos e com maior escala de integração. Por exemplo, temos informações de um microprocessador RISC, que está sendo desenvolvido com tecnologia bipolar, com capacidade de processamento de 600 MIPS e consumo de potência de 100W.

2.13. CONCLUSÕES

Neste capítulo estão apresentadas algumas contribuições realizadas para o estudo sobre o efeito de distúrbios da rede elétrica na operação de sistemas com microprocessadores.

Mostramos inicialmente alguns conceitos relevantes sobre a qualidade do suprimento de energia, como a terminologia utilizada e a classificação dos distúrbios.

Apresentamos também os resultados de um levantamento local sobre os distúrbios da rede no qual se monitorou o aparecimento de picos e sobretensões. Como complementação deste trabalho apresentamos uma arquitetura de um monitor microprocessado de transientes de rede que está em fase de desenvolvimento.

Para poder avaliar a susceptibilidade de sistemas com

microprocessadores aos transientes da rede, desenvolvemos um sistema de testes dedicado. Este sistema utiliza duas réplicas idênticas do equipamento a ser testado, uma submetida aos distúrbios e outra tomada como padrão. Os barramentos dos módulos sob teste são comparados para detectar possíveis diferenças quando estes são submetidos aos distúrbios.

Utilizando o sistema de testes, submetemos um módulo microcomputador a diversas sequências de distúrbios, com amplitude, polaridade e duração prédefinidos e registramos o efeito desses distúrbios no módulo sob teste.

Pudemos verificar que distúrbios com amplitude superior à tensão nominal de alimentação do módulo tem um efeito destrutivo, levando frequentemente à destruição de dispositivos semicondutores e travamento de memórias CMOS. Distúrbios com amplitude menor do que o valor nominal de alimentação, tem o efeito de alterar o processamento, podendo inclusive interromper a operação da máquina.

Foram apontadas algumas soluções para o problema. O nível de tolerância aos distúrbios que pode ser atingido é diretamente proporcional ao custo. Soluções mais completas envolvem certamente custos maiores.

Em se mantendo o atual nível de desempenho dos sistemas o problema tenderia a ficar menos crítico, pois há uma tendência de redução de potência dos dispositivos semicondutores. Entretanto, existe uma tendência oposta de se aumentar o desempenho dos sistemas através de dispositivos mais rápidos e com maior capacidade de processamento e de armazenamento. O preço pago por isto é um maior consumo de potência. Isso de certa maneira reaviva a importância do tema que tratamos neste capítulo.

A proteção de sistemas contra os distúrbios da rede deve

continuar , por muito tempo, sendo uma das preocupações dos projetistas e fabricantes de sistemas digitais.

CAPÍTULO 3

DESENVOLVIMENTO DE ALGUNS SISTEMAS CONFIÁVEIS.

3.0. INTRODUÇÃO.

Quando uma falha ocorre em um sistema SIMPLEX (um só módulo de processamento), o sistema permanece inoperante até que a falha seja corrigida. Para aumentar a confiabilidade destes sistemas utilizando a técnica de tolerância a falhas devem ser incluidos elementos redundantes no sistema, que deverão substituir os elementos que venham a falhar.

Neste capítulo apresentamos alguns sistemas e estudos que realizamos utilizando técnicas de tolerância a falhas. Apresentamos o desenvolvimento de memórias utilizando codificação redundante para detectar e corrigir erros, onde a redundância é implementada através de bits adicionais na memória. Mostramos o desenvolvimento de uma arquitetura triplicada em anel que é baseada em um conceito tradicional de tolerância a falhas: o de redundância modular múltipla. Mostramos também um estudo sobre a recuperação de falhas transientes em sistemas com microprocessadores e a proposta de uma arquitetura DUPLEX/HÍBRIDA.

3.1. DETECÇÃO E CORREÇÃO DE ERROS EM MEMÓRIAS PARA MICROCOMPUTADORES.

3.1.1. Considerações sobre a confiabilidade de memórias a semicondutor.

A confiabilidade de sistemas de computação está diretamente relacionada à confiabilidade das memórias utilizadas. Cada memória pode ser considerada por si só um sistema funcional sendo sua menor unidade uma célula de armazenamento. As estruturas internas da memória possuem mecanismos inerentes de falha que afetam as células de armazenamento individualmente.

As memórias a semicondutor são organizadas com as células de armazenamento em conjuntos de linhas e colunas. As falhas nestas memórias podem ser de natureza permanente ou transiente e podem afetar uma simples célula, linhas, colunas, combinações de linhas e colunas e em algumas ocasiões parte da memória ou mesmo todo o dispositivo [MARSTON, 1980].

As falhas permanentes ocorrem por defeitos físicos inerentes do dispositivo como curto-circuitos, ligações abertas ou com mau contacto, micro-trincas e outras.

As falhas transientes são caracterizadas por serem de natureza aleatória, não recorrentes e por não destruirem as células da memória. Estas falhas são normalmente causadas por distúrbios externos (elétricos e eletro-magnéticos), problemas de temporização, temperatura e outros efeitos que possam fazer com que o dispositivo opere fora dos seus limites normais de tolerância. Outra causa de falhas transientes são radiações ionizantes, como as particulas alfa, que alteram a carga em regiões da memória com substratos de alta impedância [NATIONAL, 1987]. Na ocorrência de uma falha transiente, a informação corrompida pode ser detectada durante uma operação de leitura da memória e para corrigí-la é necessário recuperar a informação correta e reescrevê-la na memória.

A crescente complexidade e capacidade de armazenamento das memórias a semicondutor tem radicalmente diminuido as dimensões dos seus elementos aumentando a susceptibilidade desses dispositivos a distúrbios externos e à radiação. Para contornar esse problema, esquemas especiais para a detecção e correção de erros são utilizados, esquemas esses baseados na redundância de informação. A seguir apresentamos alguns esquemas que são normalmente utilizados para detecção e correção de erros em memórias.

3.1.2. Codificação para detecção e correção de erros.

Pelo acréscimo de alguns bits a uma palavra de memória é possível detectar se ocorreu alguma alteração nos dados armazenados. Esses bits são chamados códigos redundantes. Alguns dos códigos redundantes permitem não apenas identificar se a palavra lida está errada, mas também identificar quais os bits errados e a partir destes reconstruir o dado original. Esses códigos são chamados detectores/corretores de erro.

Dois conceitos importantes sobre os códigos redundantes são a eficiência e a distância **[ALTNETHER, 1979]**.

A **eficiência de um código** (E) é definida como a razão entre o número de bits da palavra codificada e o número de bits da palavra original.

$$E = \frac{N}{M} = \frac{M + K}{M}$$

Onde: N é o número de bits da palavra codificada, M é o número de bits da palavra original e K é o número de bits redundantes.

A **distância** é definida como o número de bits que diferem entre

duas palavras. Por exemplo:

```
1  1  0 | 0  1 |   dois bits diferem
1  1  0 | 1  0 |     distância = 2
```

A distância mínima para que seja possível a detecção de erros deve ser 2.

Existem diversos códigos para detecções e/ou correção de erros. A classificação destes códigos de acordo com **SIEWIOREK e SWARZ [1982]** é a seguinte: códigos M de N ("M-of-N"), códigos de paridade, soma de palavras ("checksums"), códigos aritméticos e códigos cíclicos. Destes o mais utilizado é a paridade que inclui o código de Hamming.

A **paridade** é um dos códigos com distância mínima 2 e envolve a utilização de um bit extra de informação chamado bit de paridade. As palavras codificadas podem ter paridade par, que corresponde a um número par de bits "1" na palavra ou paridade ímpar que corresponde a um número ímpar de bits "1". As paridades ímpar e par são utilizadas distintamente, nunca se misturando. Por exemplo, no caso de paridade par o bit de paridade será "1" se a palavra contiver um número par de bits iguais a "1" e "0" se contiver um número ímpar de bits iguais a "1".

No caso da paridade par, o bit de paridade (C) é gerado pela função ou-exclusivo de todos os bits da palavra.

$$C = b_0 \oplus b_1 \oplus b_2 \oplus \ldots\ldots\ldots\ldots\ldots b_n$$

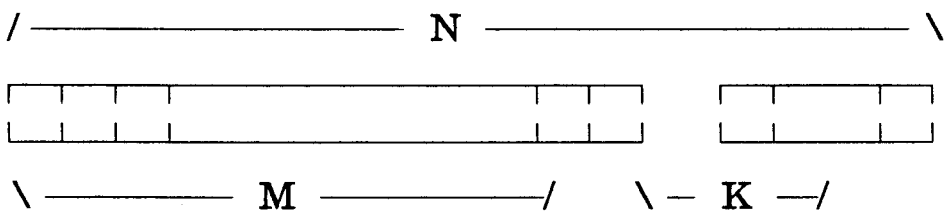
Antes da palavra ser armazenada na memória, o bit de paridade é gerado e salvo. Quando a palavra é lida da memória, o bit de paridade é novamente gerado e comparado com o bit armazenado, a não coincidência indica alteração no dado.

As limitações do uso da paridade são decorrentes do fato que

múltiplos erros podem passar desapercebidos e que embora a paridade detecte o erro, ela não permite a sua correção.

Os **códigos de Hamming**, que utilizam uma técnica similar à da paridade, tem sido os códigos mais utilizados para correção de erros em memórias.

Para uma palavra de memória com M bits, K bits extras são utilizados e adicionados à palavra formando a palavra codificada.



O código de Hamming é bastante similar à paridade. A diferença está na relação entre os K bits adicionais e os M bits de dados. Os M bits são armazenados na memória de dados e os K bits são armazenados em outra memória adicionada ao sistema.

Os K bits armazenados na memória são comparados, pela função ou-exclusivo, com os K' bits gerados a partir da palavra lida da memória. O resultado da comparação é chamado palavra "syndrome" e contém informações para indicar se houve ou não erro e para especificar onde ocorreu o erro. Depois de interpretada esta informação, a correção é feita invertendo-se o bit errado.

```
  X X X X X X ........... X   K
⊕ X X X X X X ........... X   K'
  X X X X X X ........... X   syndrome
```

A palavra syndrome tem K bits e pode assumir 2^K valores, compreendidos entre 0 e $2^K - 1$. Um desses valores é zero, e é o que vai indicar que não foi detectado erro. Os $2^k - 1$ valores restantes

indicarão qual dos N bits está errado. Desta última relação pode-se obter o tamanho de K. Para corrigir um bit errado em uma palavra contendo M bits de dados, temos:

$$2^K-1 \geq N \qquad mas \quad N=M+K$$

$$2^K \geq M+K+1$$

Existem duas variantes do código de Hamming, o que detecta e corrige 1 bit de erro, chamado de única detecção/única correção e o que detecta 2 bits de erro e corrige 1, chamado de dupla detecção/única correção ou código de Hamming modificado.

A tabela 3.1 mostra a relação entre M e K para vários comprimentos de palavra de memória e a tabela 3.2 mostra a distância mínima entre palavras nas duas variantes do código de Hamming mencionadas.

Todos os bits, incluindo os K bits adicionais que são chamados de **bits de verificação** são identificados pela sua posição na palavra. As posições são representadas por números binários e os bits de verificação assumem a posição dos números binários que tem um único bit igual a "1". A tabela 3.3 mostra as posições dos bits de dados e de verificação na palavra codificada.

Os bits de verificação são obtidos igualando as potências de 2 nas posições binárias. Cada bit de verificação será determinado pela função ou-exclusivo (paridade) dos bits que tiverem "1" na linha de posição. Os bits de verificação são gerados utilizando as seguintes equações:

$$C1 = M1 \oplus M2 \oplus M4 \oplus M5 \oplus M7 \oplus M9 \oplus M11 \oplus M12 \oplus M14 \oplus M16$$
$$C2 = M1 \oplus M3 \oplus M4 \oplus M6 \oplus M7 \oplus M10 \oplus M11 \oplus M13 \oplus M14$$
$$C4 = M2 \oplus M3 \oplus M4 \oplus M8 \oplus M9 \oplus M10 \oplus M11 \oplus M15 \oplus M16$$
$$C8 = M5 \oplus M6 \oplus M7 \oplus M8 \oplus M9 \oplus M10 \oplus M11$$
$$C16 = M12 \oplus M13 \oplus M14 \oplus M15 \oplus M16$$

Tabela 3.1: Relação entre o código redundante (K) e o tamanho da palavra (M) para os códigos de Hamming.

	única detecção/ única correção		dupla detecção/ única correção	
K	≤ M ≥		≤ M ≥	
4	4	11	1	3
5	12	26	4	10
6	27	57	11	25
7	58	120	26	56
8	121	245	57	119

Tabela 3.2: Distância mínima para diversos códigos detectores e/ou corretores de erro.

distância mínima	processo utilizado
2	detecção de 1 erro apenas (paridade)
3	única detecção/única correção
4	dupla detecção/única correção

Os bits de verificação são armazenados na memória com os bits de dados, formando uma mensagem de 21 bits.

M16 M15 M14 M13 M12 C16 M11 M10 M9 M8 M7 M6 M5 C8 M4 M3 M2 C4 M1 C2 C1

Quando ocorre uma leitura na memória, os bits de verificação são recalculados e comparados com os armazenados na memória,

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC

Tabela 3.3: Posicionamento dos bits de dados e de verifição na palavra codificada.

16	15	14	13	12		11	10	9	8	7	6	5		4	3	2		1				Bits de dados
					C16								C8				C4		C2	C1		Bits de posição
21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		Posição
1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	2^0	C1
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	2^1	C2
1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	2^2	C4
0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	2^3	C8
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2^4	C16

gerando a palavra syndrome de 5 bits.

$$S1 = M1 \oplus M2 \oplus M4 \oplus M5 \oplus M7 \oplus M9 \oplus M11 \oplus M12 \oplus M14 \oplus M16 \oplus C1$$
$$S2 = M1 \oplus M3 \oplus M4 \oplus M6 \oplus M7 \oplus M10 \oplus M11 \oplus M13 \oplus M14 \oplus C2$$
$$S4 = M2 \oplus M3 \oplus M4 \oplus M8 \oplus M9 \oplus M10 \oplus M11 \oplus M15 \oplus M16 \oplus C4$$
$$S8 = M5 \oplus M6 \oplus M7 \oplus M8 \oplus M9 \oplus M10 \oplus M11 \oplus C8$$
$$S16 = M12 \oplus M13 \oplus M14 \oplus M15 \oplus M16 \oplus C16$$

Se nenhum bit estiver errado, a palavra syndrome terá todos os seus bits iguais a "0". Se um dos bits de verificação estiver errado, a palavra syndrome terá um bit "1" na posição correspondente a este bit e os demais bits iguais a "0". Caso ocorra uma inversão de um bit na palavra armazenada na memória, a palavra syndrome irá indicar a posição do bit errado na palavra de 21 bits, que pode ser corrigido por inversão.

Por exemplo, consideremos que o bit M13 tenha sido alterado por algum distúrbio externo. Neste caso a palavra syndrome resultante da comparação dos bits de verificação será 1 0 0 1 0.

Para que este bit seja corrigido deve ser utilizada a seguinte equação:

$$M13C = S16.S8.S4.S2.S1 \oplus M13 = 1.0.0.1.0 \oplus M13$$

onde M13C é o bit corrigido

De maneira similar as equações de correção para os outros bits de dados podem ser determinadas.

Um esquema para a detecção e correção de erros simples para uma memória de 8 bits é apresentado na figura 3.1. A implementação deste esquema requer a síntese através de circuitos lógicos das funções que geram os bits de verificação, a palavra syndrome e a decodificação para correção. Isto será apresentado a seguir.

Apesar do código de Hamming ser simples, sua implementação por circuitos apresenta alguns problemas. O número de bits de dados submetidos à função ou-exclusivo para gerar os bits de verificação é variavel. No exemplo citado varia de 10 bits, para C1, a 5 bits para C16. O atraso de propagação de uma porta ou-exclusivo varia com o seu número de entradas, sendo maior em uma porta com 10 entradas do que em uma com 5. O sistema deverá sempre esperar a máxima propagação, o que reduzirá a sua velocidade. Equalizando o número de bits de verificação seria otimizada a velocidade dos circuitos codificadores.

Um outro problema que ocorre é que no código de Hamming estudado, dois bits incorretos podem fazer com que um bit correto seja indicado como incorreto. Devido a estas razões uma modificação do código de Hamming que detecta erros duplos e corrige erros simples é a mais utilizada para detecção e correção de erros em memórias.

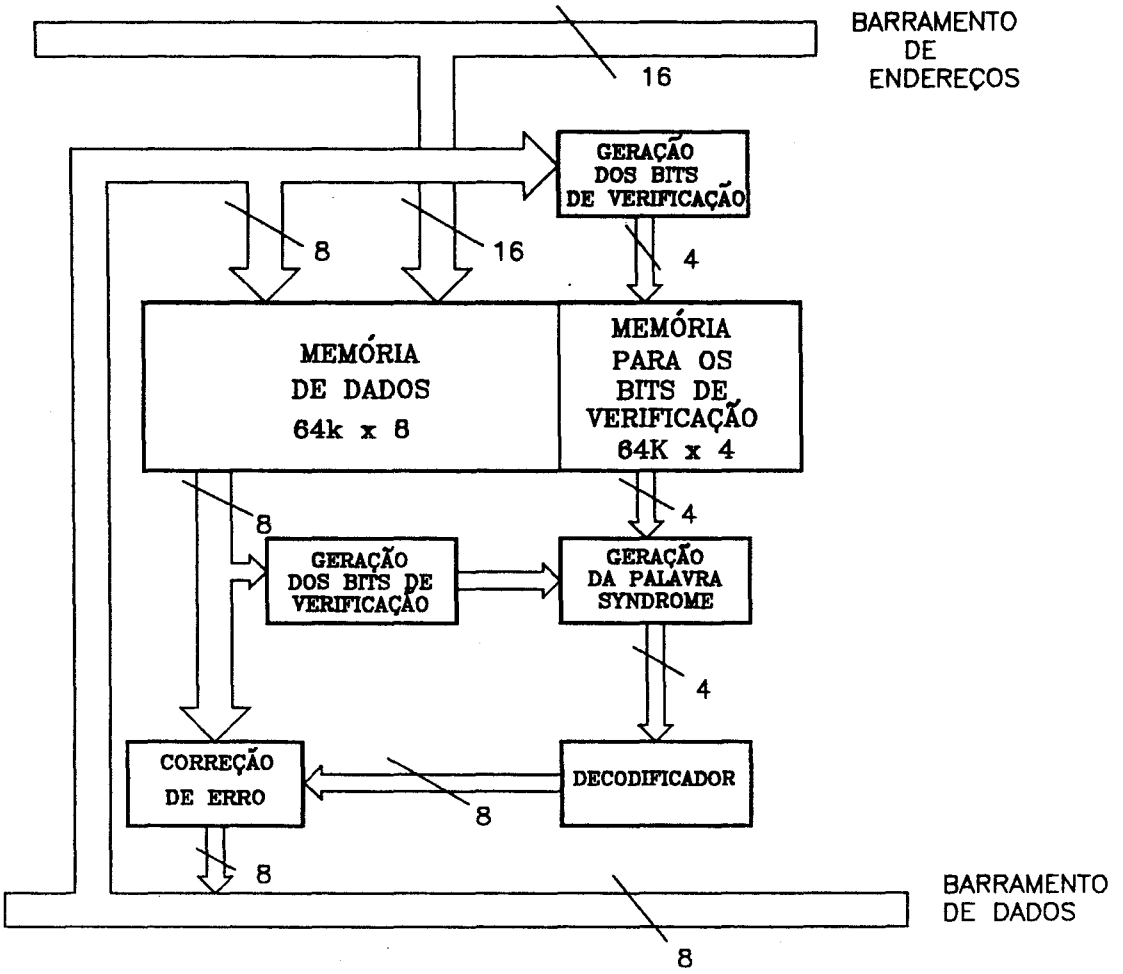


Figura 3.1: Esquema de detecção e correção de erros simples em uma memória com palavra de 8 bits.

Uma das possibilidades da ocorrência de duplo erro é através de dois bits de verificação errados. Utilizando diretamente o código de Hamming, o circuito implementado poderia corrigir um outro bit diferente do errado. A técnica de detecção de duplo erro elimina essa possibilidade separando as palavras codificadas por uma distância mínima de quatro. Como resultado, cada bit da palavra de dados fica protegido por um número impar de bits de verificação, de tal forma que a palavra syndrome tenha sempre um número ímpar de bits.

A detecção de duplo erro será feita verificando-se a paridade

dos bits que compõem a palavra syndrome. Sempre que for verificada a paridade par identificar-se-á a ocorrência de duplo erro. Portanto para se obter a capacidade de dupla detecção é necessário acrescentar um bit aos bits de verificação. Na tabela 3.1 é mostrado o número total de bits de verificação (K) em função do comprimento da palavra (M) para esta codificação.

As palavras syndrome para a dupla detecção/única correção são obtidas de maneira similar ao código de Hamming. A diferença básica é que neste código a palavra syndrome sempre tem um número ímpar de bits "1" e que esta não mapeia diretamente os bits da palavra de dados.

ALTNETHER [1979] propõe um método empírico para obtenção das palavras syndromes onde todos os estados possíveis dos bits de verificação são listados e os estados que não contiverem um número ímpar de bits "1" são eliminados da lista. No caso de palavras de 16 bits, as possíveis palavras syndrome são:

C6	C5	C4	C3	C2	C1
1	1	1	0	0	0
1	1	0	1	0	0
1	1	0	0	1	0
1	1	0	0	0	1
1	0	1	1	0	0
1	0	1	0	1	0
1	0	1	0	0	1
1	0	0	1	1	0
1	0	0	1	0	1
1	0	0	0	1	1
0	1	1	1	0	0
0	1	1	0	1	0
0	1	1	0	0	1
0	1	0	1	1	0
0	1	0	1	0	1

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

Na listagem acima foram eliminadas as palavras syndrome que tem mais do que tres bits iguais a "1", de maneira que cada palavra de dado terá tres bits protegendo-a.

Como existem 20 palavras syndrome e são necessárias apenas 16 para codificar palavras de 16 bits, escolhe-se dentre elas um conjunto tal que os bits de verificação sejam obtidos pela função ou-exclusivo de um número igual de bits. Obtem-se dessa maneira combinações como as apresentadas na tabela 3.4 onde são mostradas as palavras syndrome e os bits de verificação.

Tabela 3.4 - Posicionamento dos bits de dados e de verificação para o código de Hamming modificado.

M16	M15	M14	M13	M12	M11	M10	M9	M8	M7	M6	M5	M4	M3	M2	M1	C1	C2	C3	C4	C5	C6	
1	0	0	0	1	1	1	0	1	0	1	0	0	1	0	1	1	0	0	0	0	0	C1
0	1	1	0	0	0	1	1	0	1	0	1	0	0	1	1	0	1	0	0	0	0	C2
1	1	0	1	0	1	0	1	0	0	0	0	1	1	1	0	0	0	1	0	0	0	C3
1	1	1	1	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	C4
0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	C5
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	1	C6

Da tabela 3.4 obtem-se as seguintes equações para geração dos bits de verificação e palavras syndrome, respectivamente:

$$C1 = M1 \oplus M3 \oplus M6 \oplus M8 \oplus M10 \oplus M11 \oplus M12 \oplus M16$$
$$C2 = M1 \oplus M2 \oplus M5 \oplus M7 \oplus M9 \oplus M10 \oplus M14 \oplus M15$$
$$C3 = M2 \oplus M3 \oplus M4 \oplus M9 \oplus M11 \oplus M13 \oplus M15 \oplus M16$$
$$C4 = M4 \oplus M5 \oplus M6 \oplus M12 \oplus M13 \oplus M14 \oplus M15 \oplus M16$$
$$C5 = M7 \oplus M8 \oplus M9 \oplus M10 \oplus M11 \oplus M12 \oplus M13 \oplus M14$$
$$C6 = M1 \oplus M2 \oplus M3 \oplus M4 \oplus M5 \oplus M6 \oplus M7 \oplus M8$$

$$S1 = M1 \oplus M3 \oplus M6 \oplus M8 \oplus M10 \oplus M11 \oplus M12 \oplus M16 \oplus C1$$
$$S2 = M1 \oplus M2 \oplus M5 \oplus M7 \oplus M9 \oplus M10 \oplus M14 \oplus M15 \oplus C2$$
$$S3 = M2 \oplus M3 \oplus M4 \oplus M9 \oplus M11 \oplus M13 \oplus M15 \oplus M16 \oplus C3$$
$$S4 = M4 \oplus M5 \oplus M6 \oplus M12 \oplus M13 \oplus M14 \oplus M15 \oplus M16 \oplus C4$$
$$S5 = M7 \oplus M8 \oplus M9 \oplus M10 \oplus M11 \oplus M12 \oplus M13 \oplus M14 \oplus C5$$
$$S6 = M1 \oplus M2 \oplus M3 \oplus M4 \oplus M5 \oplus M6 \oplus M7 \oplus M8 \oplus C6$$

A detecção de duplo erro será feita gerando-se um bit de paridade da palavra syndrome. Caso seja detectado um número par de zeros significará que ocorreu duplo erro.

$$DBE = S6 \oplus S5 \oplus S4 \oplus S3 \oplus S2 \oplus S1$$

Quando não ocorrer nenhum erro a palavra syndrome terá todos os seus bits iguais a zero. Quando ocorrer um único erro este será detectado e corrigido por inversão e novamente escrito na memória. No caso de ocorrência de duplo erro, este será indicado pelo sinalizador DBE, sendo a posição indicada pela palavra syndrome não correta. O esquema de dupla detecção/simples correção para uma memória de 8 bits está apresentado na figura

3.2.

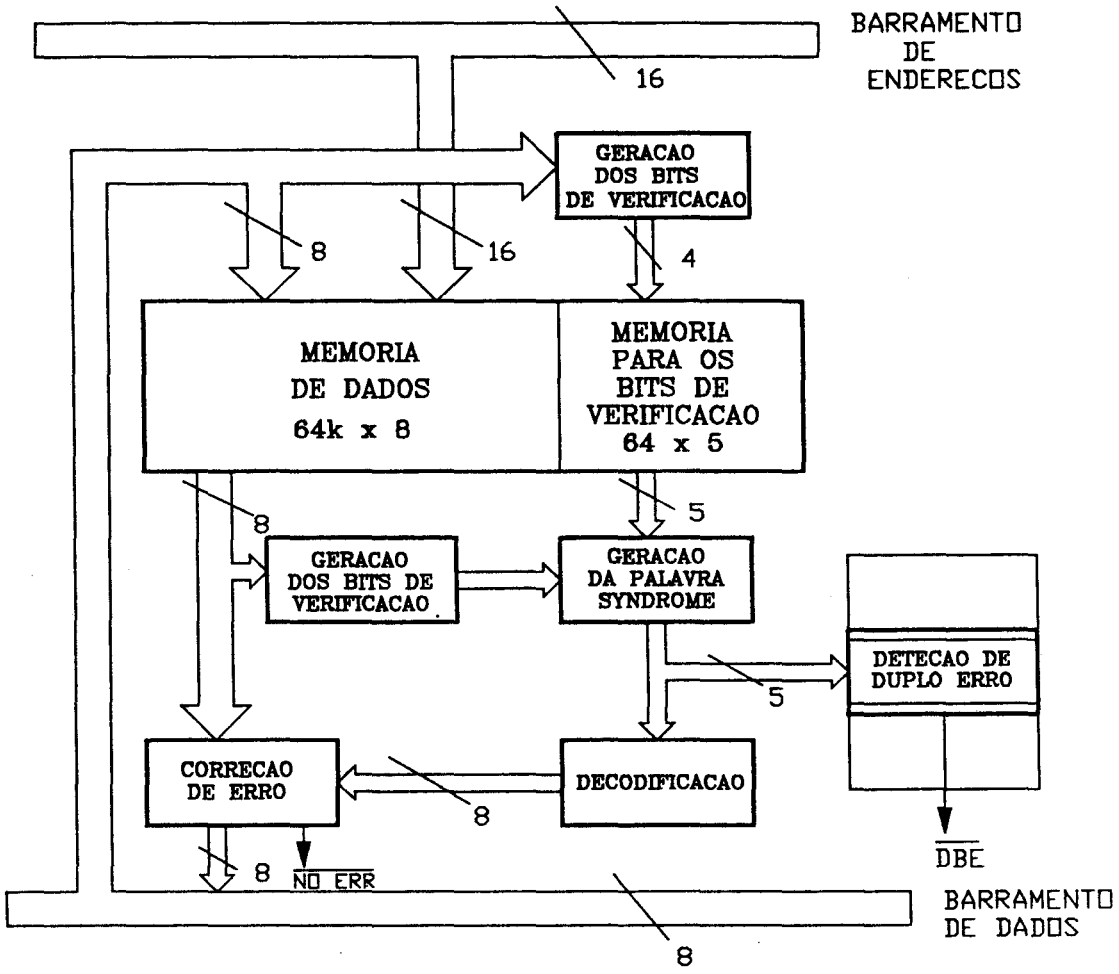


Figura 3.2: Esquema de dupla detecção/simples correção em uma memória com palavra de 8 bits.

Utilizando os principios aqui descritos, implementamos os esquemas de única detecção/única correção e de dupla detecção/única correção para uma memória de 8 bits que estão apresentados a seguir.

3.1.3. Implementação do esquema de única detecção/única correção para uma memória de 8 bits.

A técnica de única detecção/única correção foi implementada

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC
FÍSICA

em um circuito de memória dinâmica de 64K x 8 bits constituido de memórias 4164 e possuindo um controlador para refrescamento destas memórias (INTEL 8203). O esquema utilizado é aquele mostrado na figura 3.1. Foram implementados os circuitos lógicos, utilizando tecnologia TTL, para geração dos bits de verificação, geração da palavra syndrome e decodificação para correção do erro. Para armazenamento dos bits de verificação foram acrescidos 4 bits adicionais para cada palavra, formando uma memória de 64K x 4 bits.

As equações para geração dos bits de verificação no caso de 8 bits de dados são as seguintes:

$$C1 = M1 \oplus M2 \oplus M4 \oplus M5 \oplus M7$$
$$C2 = M1 \oplus M3 \oplus M4 \oplus M6 \oplus M7$$
$$C4 = M2 \oplus M3 \oplus M4 \oplus M8$$
$$C8 = M5 \oplus M6 \oplus M7 \oplus M8$$

A mensagem formada dos bits de dados com os bits de verificação será a seguinte:

M8 M7 M6 M5 C8 M4 M3 M2 C4 M1 C2 C1

As equações para geração da palavra syndrome são:

$$S1 = M1 \oplus M2 \oplus M4 \oplus M5 \oplus M7 \oplus C1$$
$$S2 = M1 \oplus M3 \oplus M4 \oplus M6 \oplus M7 \oplus C2$$
$$S4 = M2 \oplus M3 \oplus M4 \oplus M8 \oplus C4$$
$$S8 = M5 \oplus M6 \oplus M7 \oplus M8 \oplus C8$$

As equações para geração dos bits de verificação e da palavra syndrome possuem elementos comuns e são geradas em instantes

diferentes. Os bits de verificação são gerados quando da escrita de dados na memória e a palavra syndrome é gerada quando um dado é lido da memória. Por essa razão utilizou-se o mesmo circuito para gerar os dois, de acordo com as seguintes equações:

$$X1 = M1 \oplus M2 \oplus M4 \oplus M5 \oplus M7 \oplus (C1 \,.\, R/\overline{W})$$
$$X2 = M1 \oplus M3 \oplus M4 \oplus M6 \oplus M7 \oplus (C2 \,.\, R/\overline{W})$$
$$X4 = M2 \oplus M3 \oplus M4 \oplus M8 \oplus (C4 \,.\, R/\overline{W})$$
$$X8 = M5 \oplus M6 \oplus M7 \oplus M8 \oplus (C8 \,.\, R/\overline{W})$$

O circuito utilizado foi um gerador de paridade (74LS280), onde uma das entradas (a correspondente ao bit de verificação lido da memória) passa por uma porta E (74LS08). Durante a escrita da leitura o sinal de escrita/leitura ($R/\overline{W}$) está no nível lógico "0", fazendo com que a entrada do gerador de paridade também fique no nivel lógico "0". A saída do gerador de paridade neste caso será o bit de verificação. Um diagrama simplificado do circuito utilizado é mostrado na figura 3.3.

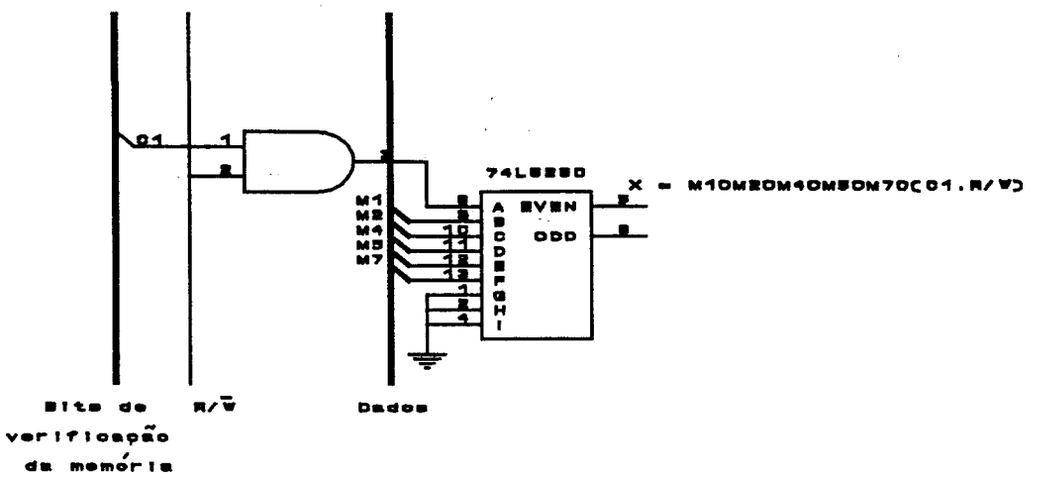


Figura 3.3: Circuito para geração dos bits de verificação e da palavra syndrome.

Tendo-se a mensagem de 12 bits (M8 M7 M6 M5 C8 M4 M3

M2 C4 M1 C2 C1) e gerada a palavra syndrome (S8 S4 S2 S1), as equações para correção da mensagem no caso de um erro é a seguinte:

$$M8C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M8$$
$$M7C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M7$$
$$M6C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M6$$
$$M5C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M5$$
$$C8C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus C8$$
$$M4C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M4$$
$$M3C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M3$$
$$M2C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M2$$
$$C4C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus C4$$
$$M1C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M1$$
$$C2C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M2$$
$$C1C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M1$$

A correção dos bits de verificação não é necessária já que o microcomputador não os utiliza como dados. As equações acima foram implementadas utilizando circuitos decodificadores (74LS138), dos quais as saídas foram escolhidas da sua tabela verdade, e portas ou exclusivo (74LS86). O circuito implementado é mostrado na figura 3.4. Um diagrama completo da memória está apresentado no apêndice A.

O circuito implementado foi conectado a um microprocessador de 8 bits baseado na UCP Z80. Foram realizados diversos testes para verificar o desempenho do circuito, simulando-se falhas de dados através de ligações abertas e medindo-se a temporização dos sinais. O circuito satisfez plenamente as expectativas.

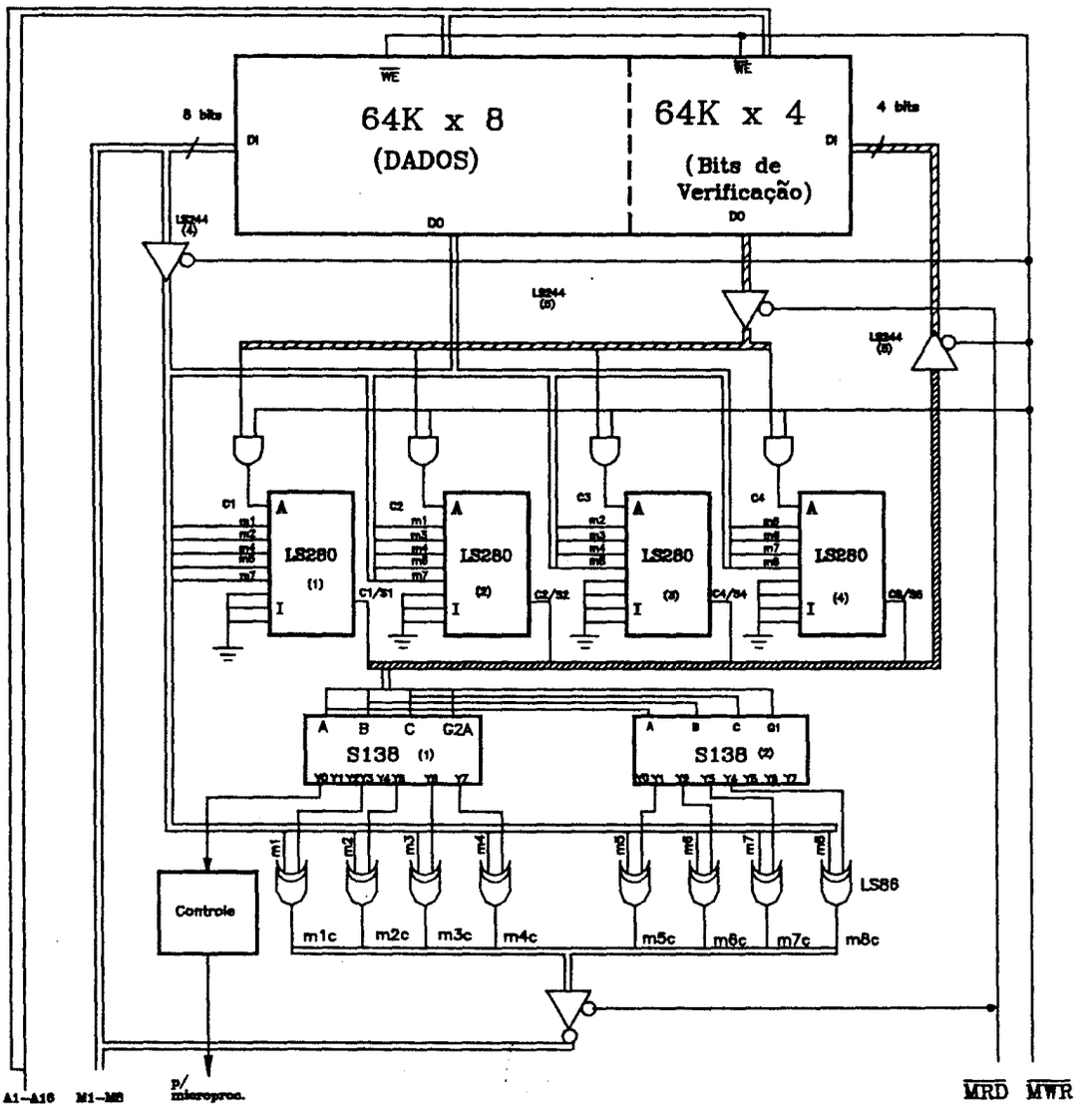


Figura 3.4: Diagrama simplificado da memória com detecção/correção de erros utilizando o código de Hamming.

3.1.4. Implementação do esquema de dupla detecção/única correção para uma memória de 8 bits.

A técnica de dupla detecção/única correção foi implementada no mesmo circuito de memória dinâmica de 8 bits mencionado no item anterior. O esquema utilizado é aquele mostrado na figura 3.2 e foram implementados os circuitos lógicos, utilizando tecnologia

TTL, para geração dos bits de verificação, geração da palavra syndrome e decodificação para correção do erro. Para armazenamento dos bits de verificação foram acrescidos 5 bits adicionais para cada palavra, formando uma memória de 64K x 5 bits [ALTNETHER, 1979]. Para obter as possíveis palavras syndrome, todos os estados possíveis dos bits de verificação foram listados eliminado-se todos os estados que não continham tres bits "1". Neste caso, as possíveis palavras syndrome são:

C5	C4	C3	C2	C1
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0

Eliminando duas colunas já que apenas 8 são necessárias, montamos a tabela de mapeamento das palavras syndrome e bits de verificação (tabela 3.5). Observe-se que uma das linhas não possui o mesmo número de bits "1" que as demais. Apesar deste inconveniente temos um circuito bastante uniforme em termos do tempo de propagação.

Da tabela 3.5 são obtidas as equações para geração dos bits de verificação e palavras syndrome:

$C1 = M1 \oplus M2 \oplus M6 \oplus M7 \oplus M8$
$C2 = M2 \oplus M3 \oplus M4 \oplus M5 \oplus M7$
$C3 = M1 \oplus M2 \oplus M3 \oplus M5 \oplus M6$
$C4 = M4 \oplus M5 \oplus M6 \oplus M7 \oplus M8$
$C5 = M1 \oplus M3 \oplus M4 \oplus M8$

Tabela 3.5 - Mapeamento das palavras syndrome e bits de verificação.

M8	M7	M6	M5	M4	M3	M2	M1	C1	C2	C3	C4	C5	
1	1	1	0	0	0	1	1	1	0	0	0	0	C1
0	1	0	1	1	1	1	0	0	1	0	0	0	C2
0	0	1	1	0	1	1	1	0	0	1	0	0	C3
1	1	1	1	1	0	0	0	0	0	0	1	0	C4
1	0	0	0	1	1	0	1	0	0	0	0	1	C5

$S1 = M1 \oplus M2 \oplus M6 \oplus M7 \oplus M8 \oplus C1$
$S2 = M2 \oplus M3 \oplus M4 \oplus M5 \oplus M7 \oplus C2$
$S3 = M1 \oplus M2 \oplus M3 \oplus M5 \oplus M6 \oplus C3$
$S4 = M4 \oplus M5 \oplus M6 \oplus M7 \oplus M8 \oplus C4$
$S5 = M1 \oplus M3 \oplus M4 \oplus M8 \oplus C5$

As funções acima foram implementadas com circuitos geradores de paridade (74LS280) de maneira similar ao circuito de única detecção/única correção.

Para a detecção de duplo erro também foi usado um circuito gerador de paridade que gera o sinal de indicação de duplo erro (DBE) da equação abaixo.

$$DBE = S1 \oplus S2 \oplus S3 \oplus S4 \oplus S5$$

Tendo-se a mensagem de 12 bits as equações para correção da mensagem no caso de um erro são as seguintes:

$$M8C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M8$$
$$M7C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M7$$
$$M6C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M6$$
$$M5C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M5$$
$$M4C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M4$$
$$M3C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M3$$
$$M2C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M2$$
$$M1C = S8 \,.\, S4 \,.\, S2 \,.\, S1 \oplus M1$$

As equações acima foram implementadas utilizando circuitos decodificadores (74LS138), dos quais as saidas foram escolhidas da sua tabela verdade, e portas ou exclusivo (74LS86). O circuito implementado está mostrado na figura 3.5. O diagrama completo da memória esta apresentado no apêndice A.

O circuito implementado foi conectado a um microprocessador de 8 bits baseado na UCP Z80. Foram realizados diversos testes similares aos realizados no esquema de única deteção/única correção para verificar o desempenho do circuito, tendo o circuito satisfeito plenamente as espectativas.

3.2 - MICROCOMPUTADOR TOLERANTE A FALHAS COM ARQUITETURA EM ANEL

3.2.1. Aspectos gerais da arquitetura em anel.

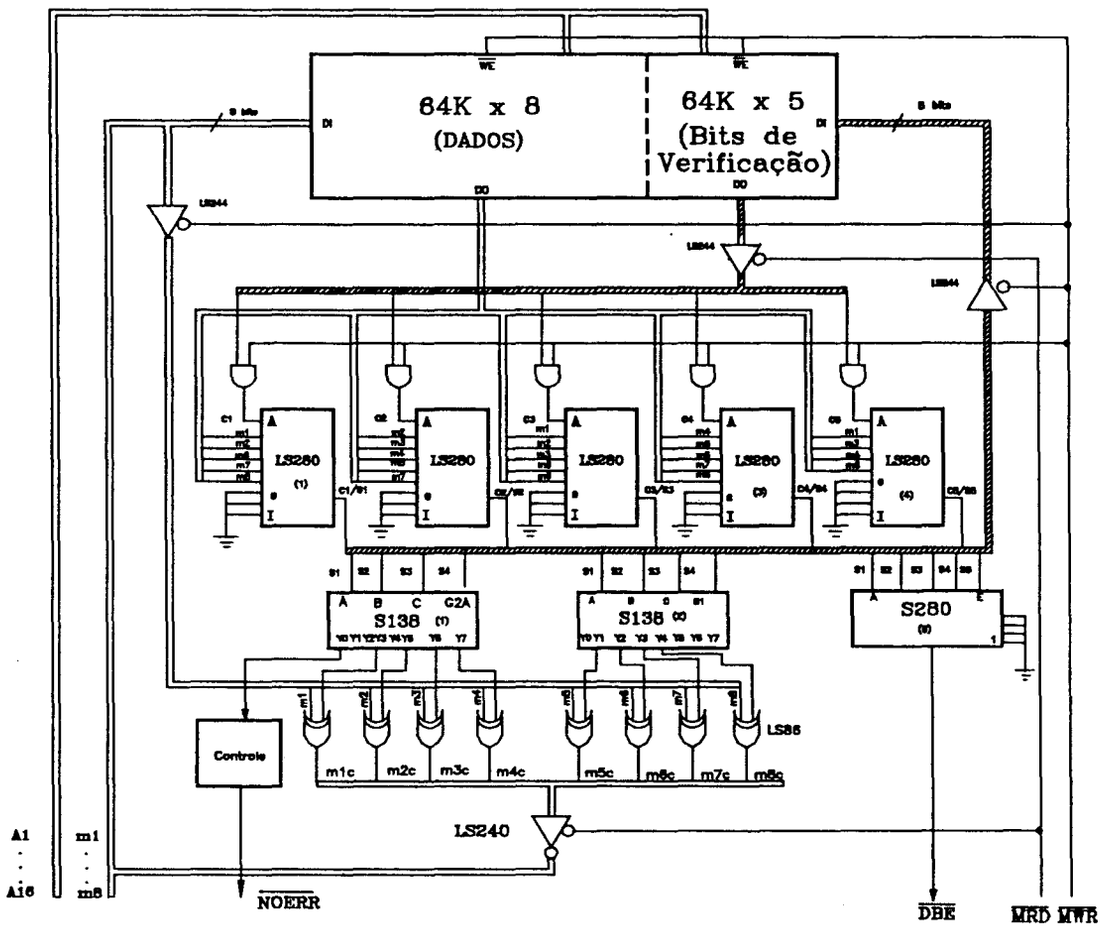


Figura 3.5: Diagrama simplificado da memórida com dupla detecção/simples correção (código de Hamming modificado).

A arquitetura em anel, inicialmente desenvolvida na minha tese de doutorado **[OBAC-RODA, 1983]** é uma variação dos sistemas com redundância modular multipla (N modular redundant structure, NMR") **[MATHUR e de SOUZA, 1975]**.

A arquitetura em anel é constituida por módulos idênticos de processamento conectados por barramentos de comunicação formando um anel "loop". Um diagrama de blocos desta arquitetura é apresentado na figura 3.6. Os módulos possuem entradas comuns, recebendo os dados de uma ou mais fontes. Os dados são processados e o resultado do processamento é passado para os outros módulos através do canal de comunicação, havendo

circulação de dados através do anel.

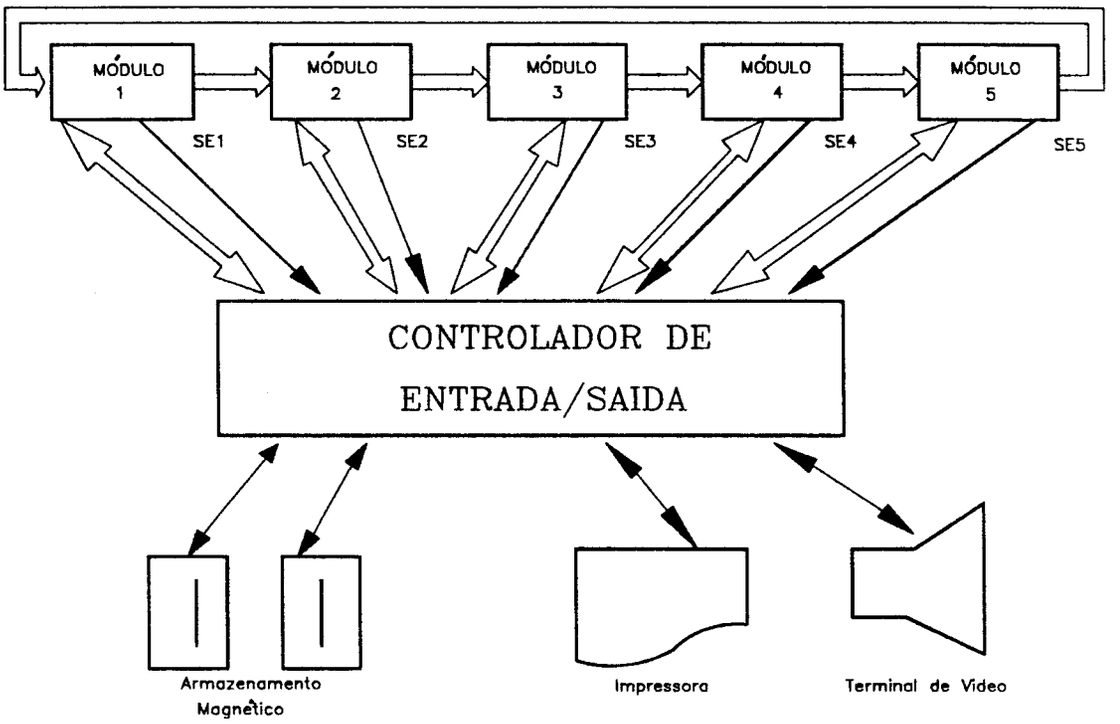


Figura 3.6: Arquitetura em anel com 5 módulos de processamento.

A verificação da validade dos dados (concordância ou discordância) é realizada por um processo de votação majoritária [von NEUMAN, 1956] e o resultado da votação em cada módulo é indicado por um sinal de condição de erro. O estado lógico dos sinalizadores de erro é utilizado por uma lógica externa aos módulos para a seleção da saida do sistema dentre um dos módulos de operação correta.

Na tese de doutorado foram analisados diversos aspectos teóricos e de implementação da arquitetura em anel, pelo presente autor. A topologia desta arquitetura foi estudada e duas estratégias de votação sobre os dados foram identificadas: a votação majoritária, onde cada módulo executa o processo de votação sobre os dados de todos os módulos do sistema e a votação

parcial de dados onde o processo de votação é realizado sobre uma parcela dos dados disponíveis.

Foram deduzidas as equações da confiabilidade para as diversas configurações da arquitetura em anel e comparou-se sua confiabilidade com a de sistemas com redundância modular múltipla para ambos os esquemas de votação. Para o esquema com votação majoritária verificou-se que ocorre uma diminuição da confiabilidade com o aumento do número de módulos. A razão disto é o bloqueio na comunicação pelos módulos que falharam. Foram propostos alguns esquemas para minimizar o problema: utilizando múltiplos barramentos e sistema de comunicação independente dos módulos de processamento. O esquema com votação parcial de dados, embora aparentemente mais confiável apresenta dificuldades de implementação para um número grande de módulos (maior do que 3).

Da análise feita sobre a confiabilidade de sistemas em anel, concluiu-se que o sistema em anel com 3 módulos foi o que apresentou melhores resultados e maiores facilidades de implementação.

Para validar o estudo realizado, construiu-se um protótipo da arquitetura em anel utilizando microcomputadores de uma só pastilha do tipo Intel 8748.

Diversos aspectos da arquitetura em anel ficaram por ser analisados. Por essa razão foi proposta e realizada a continuação do trabalho na forma de trabalho de doutorado sob minha supervisão [**FISCHER, 1990**]. Os principais aspectos deste trabalho são descritos a seguir.

3.2.2. Características da arquitetura implementada.

Alguns aspectos críticos da arquitetura em anel que ficaram

por ser analisados e que nortearam o desenvolvimento do presente trabalho foram:

O **sistema de comunicação** deve ser eficiente e se possível independente da operação dos módulos de processamento.
Deve haver **sincronização** do processo de intercâmbio dos dados entre os módulos e na votação dos dados.
A **indicação em cada módulo do resultado da votação** deve ser correta e segura assim como a **habilitação do módulo que dá saida ao sistema.**
Eliminação de possíveis **pontos comuns vulneráveis a falhas**.

O sistema implementado tem 3 módulos, tendo sido utilizado o esquema de votação parcial de dados (comparação de dados de 2 módulos) por programação. As principais características do sistema de acordo com **[FISCHER, 1990]** são:

1) O sistema pode tolerar a falha de apenas um dos módulos.
2) O sistema é modular e pode operar sem um dos módulos ativados. Isto facilita a manutenção e aumenta a disponibilidade do sistema.
3) O sistema é transparente ao usuário. Este não precisa saber que está programando um computador tolerante a falhas.
4) O sistema operacional escolhido foi o CP/M **[HOGAN, 1983]** adaptado para a arquitetura em anel. Isto facilita a execução de programas já existentes pela máquina tolerante a falhas.
5) Os módulos são conectados por dutos paralelos de 8 bits com circulação unidirecional de dados no sentido horário.
6) O resultado de cada processamento interno é enviado aos outros módulos.
7) A comparação dos dados entre os módulos é implementada por

programação. Cada módulo compara o dado recebido do módulo adjacente com o resultado de seu próprio processamento.
8) O resultado da comparação é indicado por um sinalizador de erro que irá controlar a saida do sistema.
9) Sómente um dos módulos, de operação correta, alimenta a saída do sistema.
10) É considerado módulo falho aquele que transmite dados incorretos ou é incapaz de enviar dados.
11) Cada módulo tem sincronização independente, não havendo referência de tempo comum.

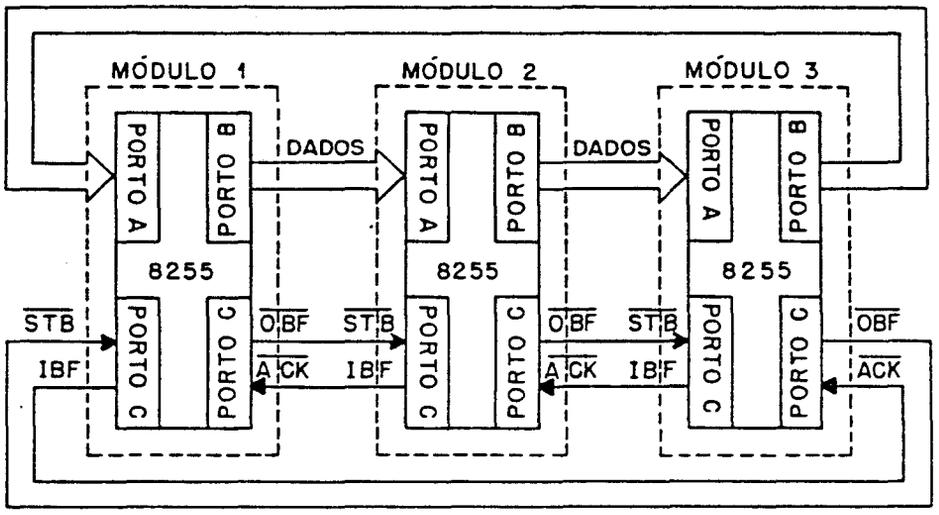
3.2.3. Detalhes da implementação do microcomputador tolerante a falhas em anel.

Cada módulo do sistema em anel é uma unidade de processamento completa, constituida por um microprocessador Z80, 1 bloco de memória permanente ("EPROM") com 32 Kbytes e 1 bloco de memória volátil ("RAM") com 64 Kbytes dividido em 2 páginas de 32 Kbytes. O módulo está baseado no microcomputador MICRO-LIE desenvolvido no Departamento de Física e Ciências dos Materiais/USP por **TRAINA [1986]**.

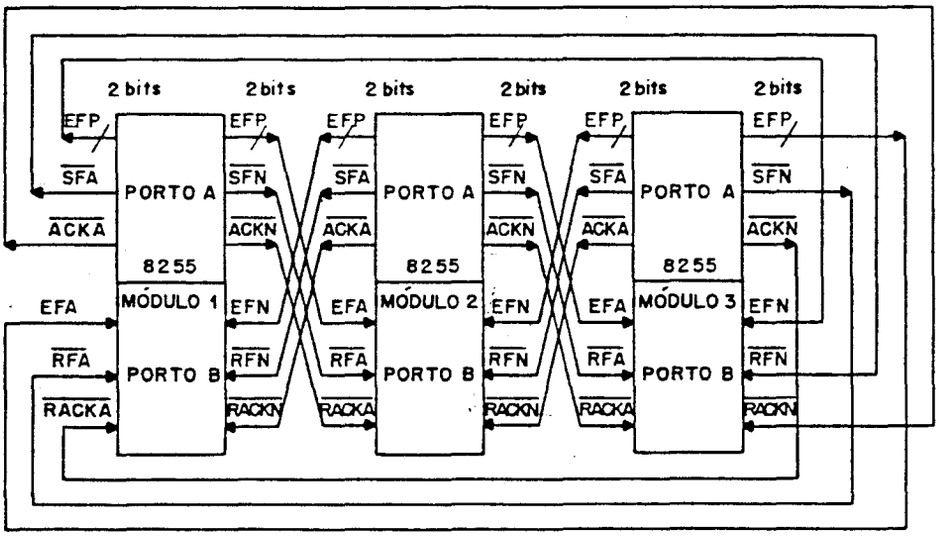
Cada módulo possue um programa monitor residente na memória permanente assim como o sistema básico de entrada e saída ("BIOS") do CP/M. Foram utilizados, com algumas modificações introduzidas, o monitor desenvolvido para o microcomputador MICRO-LIE **[MACHADO TRAINA, 1987]**, assim como a configuração básica desta máquina: esquema de relógio, configuração de blocos de memória e endereçamento de periféricos.

Os módulos se comunicam no anel através de portas paralelas do tipo INTEL 8255. O subsistema de comunicação de

dados envolve os esquemas de intercâmbio dos resultados do processamento entre os módulo e a transmissão dos resultados da votação através de sinalizadores de erro. Estes esquemas estão ilustrados na figura 3.7.



a) Transmissão dos dados para votação. (*)



b) Transmissão dos sinalizadores de erro. (*)

Figura 3.7: Esquema de transmissão no subsistema de comunicação do anel.
(*) Legendas na próxima página.

Legenda da Figura 3.7

$\overline{OBF}$ - Indica que há um dado pronto no porto.
$\overline{STB}$ - Carrega o dado no porto de entrada.
EFP - Sinalizador de erro gerado pelo módulo.
EFA - Sinalizador de erro recebido do módulo anterior.
$\overline{SFN}$ - Indica para o próximo módulo que o sinalizador de erro foi ativado.
$\overline{ACKN}$ - Indica que o sinalizador de erro do próximo módulo foi aceito.
$\overline{RACKN}$ - Recepção de reconhecimento do próximo módulo.
$\overline{RFA}$ - Recepção de aviso que ativou o sinalizador anterior.
$\overline{ACK}$ - Indica que o dado foi aceito.
IBF - Indica que o dado foi carregado na entrada.
EFN - Sinalizador de erro recebido do proximo módulo.
$\overline{SFA}$ - Indica para o módulo anterior que o sinalizador de erro foi ativado.
$\overline{ACKA}$ - Indica que o sinalizador de erro do módulo anterior foi aceito.
$\overline{RACKA}$ - Recepção de reconhecimento do módulo anterior.
$\overline{RFN}$ - Recepção de aviso que o sinalizador do próximo módulo foi ativado.

Para sincronizar os módulos foi utilizada a técnica de transmissão com reconhecimento ("handshake") para a transmissão dos dados e dos sinalizadores de erro. Nesta técnica quando o dado está disponível para ser transmitido, o módulo transmissor o coloca na saída de dados do subsistema de comunicação e ativa a linha de reconhecimento indicando que os dados na saída do subsistema são válidos. O receptor por sua vez mostra que recebeu os dados enviando um sinal de reconhecimento.

A seleção do módulo que deve alimentar a saída do sistema é feita através da configuração dos sinalizadores de erro de acordo com a tabela 3.6. Esta tabela foi construida considerando o sentido horário de circulação dos dados. Cada sinalizador de erro (EF) é constituido de 2 bits. Quando há concordância dos dados no processo de votação, o valor do sinalizador é 01 e quando há discordância é 10. Isto procura evitar que falhas nos circuitos indiquem incorretamente o resultado da votação, já que no caso de falha é mais provável que os bits sinalizadores fiquem colados em apenas um nível lógico 00 ou 11.

A seleção do módulo de saída poderia ser realizada por circuitos [OBAC RODA, 1983] ou por programação. Nesta implementação foi realizada por programação. Para isto é

Tabela 3.6: Seleção do módulo de saída de acordo com os valores lógicos dos indicadores de erro.

EF_A	EF_B	EF_C	Módulo de saída
01	01	01	Módulo A
10	01	01	Módulo B
10	01	10	Módulo B
10	10	01	Módulo C
01	10	01	Módulo C
01	01	10	Módulo A
01	10	10	Módulo A

necessário que cada módulo conheça os sinalizadores de erro dos outros módulos e através destes determine se deve ou não acessar o barramento de E/S (entrada e saída).

O subsistema de E/S é constituido pelos circuitos de acoplamento com o barramento de E/S (entrada e saída) e foi proposto, embora não tenha sido implementado, um mecanismo que denominamos linha bastão para controle do acesso do módulo ao barramento. A linha bastão é utilizada para priorizar o acesso dos módulos a um mesmo periférico em operação de leitura.

O acesso ao periférico é feito por um esquema de seleção por programação. Cada módulo gera um número aleatório que é contado decrescentemente até zero. Então, o módulo lê a linha bastão e se ela estiver livre (nível lógico 1), o módulo sinaliza o acesso ao barramento de E/S, mudando o nível lógico da linha bastão para zero. Os demais módulos só terão acesso a um periférico de E/S quando o módulo que estiver lendo o periférico liberar a linha bastão. O esquema utilizado para a linha bastão

está apresentado na figura 3.8.

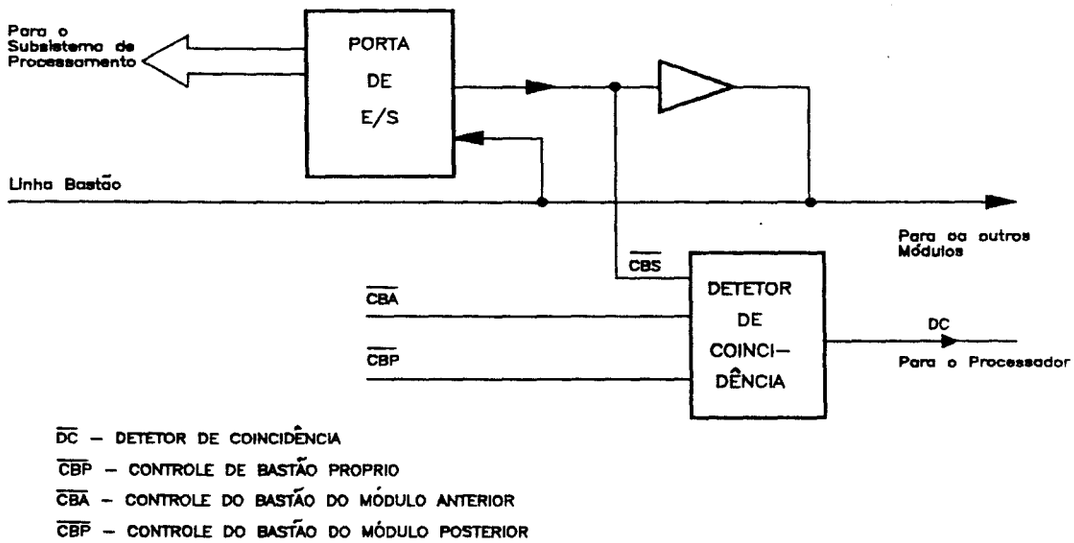


Figura 3.8: Esquema da "Linha Bastão".

Se dois módulos tentarem, simultaneamente, o acesso ao barramento de E/S, o circuito detetor de coincidência sinaliza a ocorrência e cada módulo reinicializa o processo de acesso ao barramento, gerando um outro número aleatório.

A configuração do conector dos módulos foi projetada de modo que cada módulo leia a sua posição e possa identificar qual a sua posição no anel (A, B ou C).

Dos periféricos de E/S, o armazenamento magnético (acionador de disco flexivel) é um caso crítico, pois o número de acessos de cada módulo é muito grande. Foram identificadas tres formas distintas de implementar o armazenamento magnético:

a) Um disco único com setores comuns para os tres módulos.
b) Um disco único com setores independentes para cada módulo.
c) Um disco para cada módulo.

As duas primeiras alternativas tem como principal vantagem

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC
FÍSICA

o custo menor e a facilidade de operação (apenas 1 disco no sistema). As suas principais desvantagens são a de ser um ponto vulnerável de falha no sistema e a de requerer modificações consideráveis nos programas de gerenciamento de entrada/saída. Por facilidades de implementação adotou-se a alternativa c. A configuração do sistema em anel implementado está mostrada na figura 3.9.

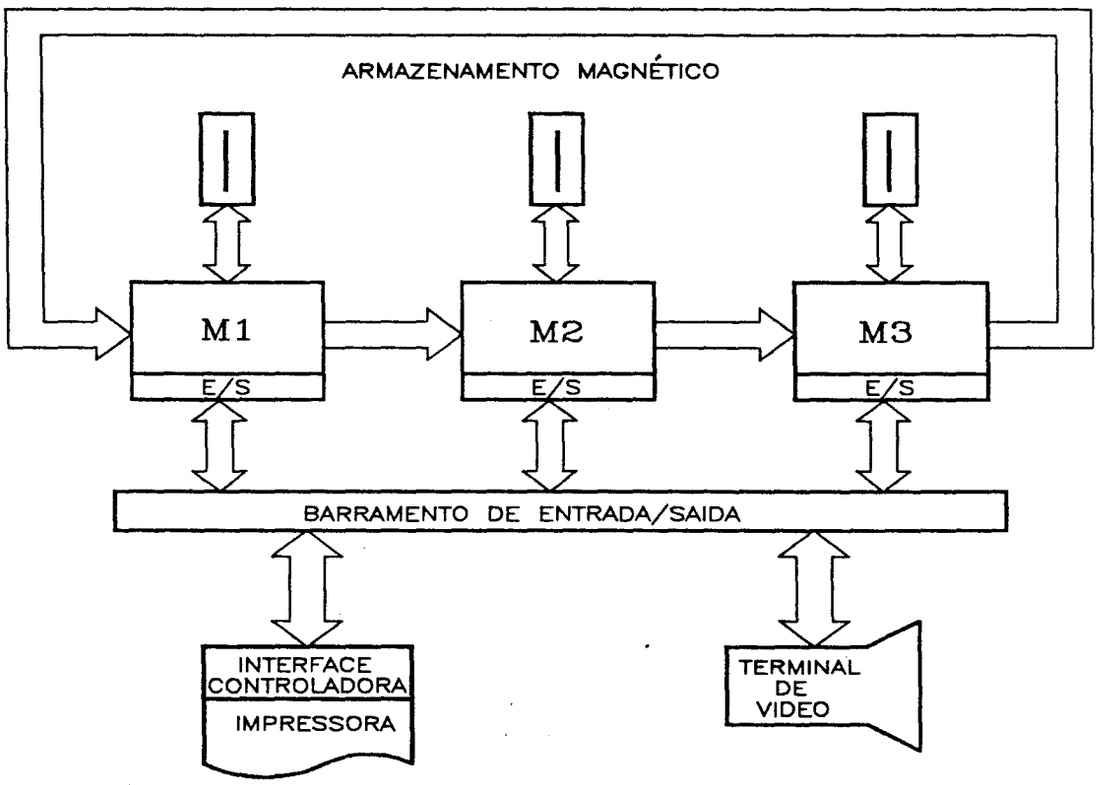


Figura 3.9: Configuração do microcomputador tolerante a falhas com arquitetura em anel.

3.2.4. Programas de gerenciamento para o microcomputador tolerante a falhas, em anel.

Uma vez definida a implementação do sistema com tres acionadores de disco flexível independentes, procurou-se

INFORMÁTICA

implementar o sistema de gerenciamento com o mínimo de alteração dos programas de gerenciamento de E/S desenvolvidos para o MICRO-LIE.

As alterações efetuadas estão divididas em dois grupos: aquelas relacionadas com as diferenças de arquitetura entre o MICRO-LIE e o sistema em anel e as associadas à implantação dos programas de gerenciamento da tolerância a falhas.

As alterações devido às diferenças entre o MICRO-LIE e o sistema em anel tratam de diferenças de endereçamento nas portas de entrada e saída e nas memórias e formas de acesso ao armazenamento magnético. As rotinas desenvolvidas, embora trabalhosas, não apresentam detalhes relevantes de implementação.

O segundo tipo de alterações estão relacionadas aos programas de gerenciamento do sistema em anel. Procurou-se fazer o mínimo possível de alterações nos programas de gerenciamento de E/S disponíveis, sem comprometer o desempenho do sistema. As rotinas de gerenciamento devem tratar com os seguintes casos: entrada de dados, saída de dados para o vídeo ou outro dispositivo de saída, votação e sincronização e transmissão dos dados.

A entrada de dados via teclado do terminal de vídeo é feita via interface de comunicação RS232, disponível em cada módulo. O acionamento de qualquer tecla é passado aos tres processadores. Neste caso é utilizada a capacidade intrínseca de sincronização do porto serial. O acesso a outros dispositivos de entrada e saída poderia ser realizado através do mecanismo "linha bastão", já descrito, no entanto não foi implementado.

O sistema em anel realiza a votação sobre os dados de saída. É neste caso que os programas de gerenciamento exercem sua função principal. Na rotina básica de entrada e saída ("BIOS") é

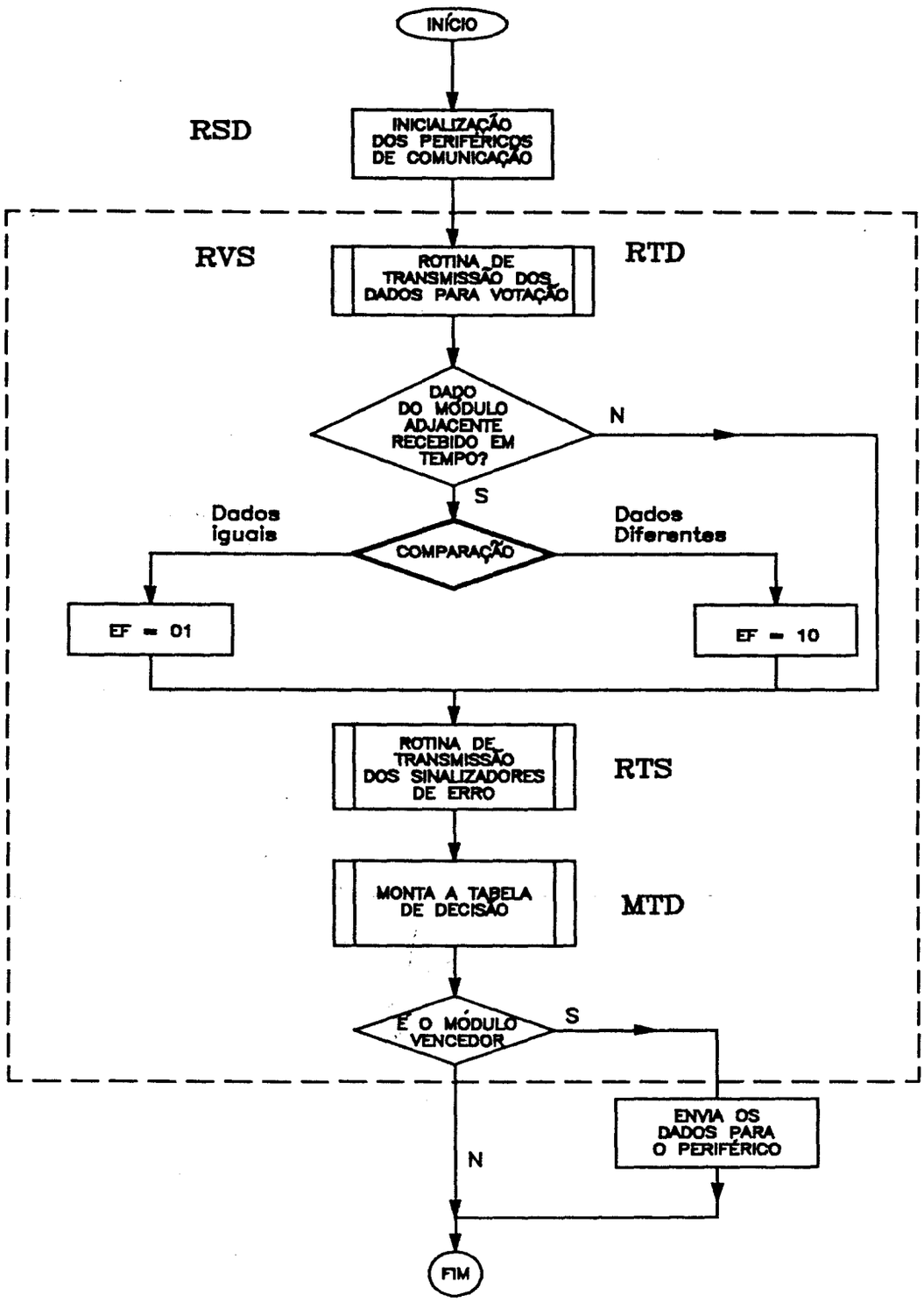


Figura 3.10: Diagrama de fluxo simplificado dos programas de transmissão de dados e votação do microcomputador, em anel.

feita uma chamada para a rotina de saída de dados do sistema em anel (RSD). Um diagrama de fluxo simplificado desta rotina é apresentado na figura 3.10.

A RSD após a inicialização dos portos de comunicação do anel e dos portos de transmissão dos sinalizadores de erro, chama a rotina de votação e sincronização (RVS). Após a execução da RVS, a rotina RSD recebe a informação do módulo que venceu a votação e habilita o envio de dados deste módulo para a saída.

A RVS tem a função de executar a sincronização dos módulos e a votação (comparação) dos dados, indicando o resultado do processo de votação através dos sinalizadores de erro. A RVS chama a rotina de transmissão dos dados para a votação (RTD) e aguarda um certo tempo para o recebimento do dado do módulo adjacente. Caso o dado não seja recebido o processo de votação é cancelado. A RVS coordena também a chamada da rotina de transmissão dos sinalizadores de erro (RTS). A RVS decide qual foi o módulo vencedor e aciona os sinalizadores correspondentes, consultando a tabela de decisão fornecida pela rotina "monta tabela de decisão" (MTD). O módulo vencedor manda então o dado para o periférico de saída.

3.2.5. Sincronização no microcomputador tolerante a falhas, em anel.

No microcomputador tolerante a falhas desenvolvido não há uma base de tempos comum para a sincronização. A base de tempos comum seria um ponto vulnerável de falhas. Sua não utilização, entretanto, pode gerar dessincronismos entre os módulos, devido a diferenças de tempo de acesso a periféricos de entrada e diferenças de fase nos próprios relógios de cada módulo. Estes dessincronismos entre os módulos diminui a probabilidade

de ocorrência de falhas correlatas. É necessário manter, entretanto, o sincronismo entre os módulos a nível de tarefas (sincronismo frouxo). Os processos de entrada de dados, processamento, votação e saída de dados devem ser realizados em paralelo pelos módulos, seguindo o mesmo fluxo de processamento e sobre a mesma versão dos dados [**WENSLEY, 1983**]. O dessincronismo, a nível de tarefa de um dos módulos pode fazer com que este seja considerado como falho.

As duas formas que tem sido utilizadas para a sincronização de sistemas redundantes são: através de uma referência comum de tempo, implementada por circuitos ou por programas [**DAVIES e WAKERLY, 1978; LAMPORT e MELLIAR SMITH, 1985; BABAOGLU e DRUMMOND, 1987**]; ou através de ponto comum de encontro [**CHANDE et al., 1989**]. A segunda alternativa foi utilizada na implementação do sistema em anel.

O ponto comum de encontro nos processos é o ponto onde cada processador tem o resultado do seu processamento disponível para a votação. A partir desse ponto, o módulo transmite o seu dado e fica em estado de espera aguardando o aviso de reconhecimento do receptor. Simultâneamente o módulo monitora a chegada do dado do módulo anterior. Sómente após o recebimento do reconhecimento de sua transmissão e do envio do seu reconhecimento de recepção para o módulo anterior é que será iniciada a rotina de votação. Este processo sincroniza os módulos, fazendo com que todos executem simultâneamente a rotina de votação. Esta é executada sobre a mesma versão dos dados. Um diagrama mostrando este processo de sincronismo é apresentado na figura 3.11.

Caso um dos módulos venha a perder a sincronização devido a uma falha transiente, este poderá ser novamente colocado em operação através de um processo de recuperação por programação

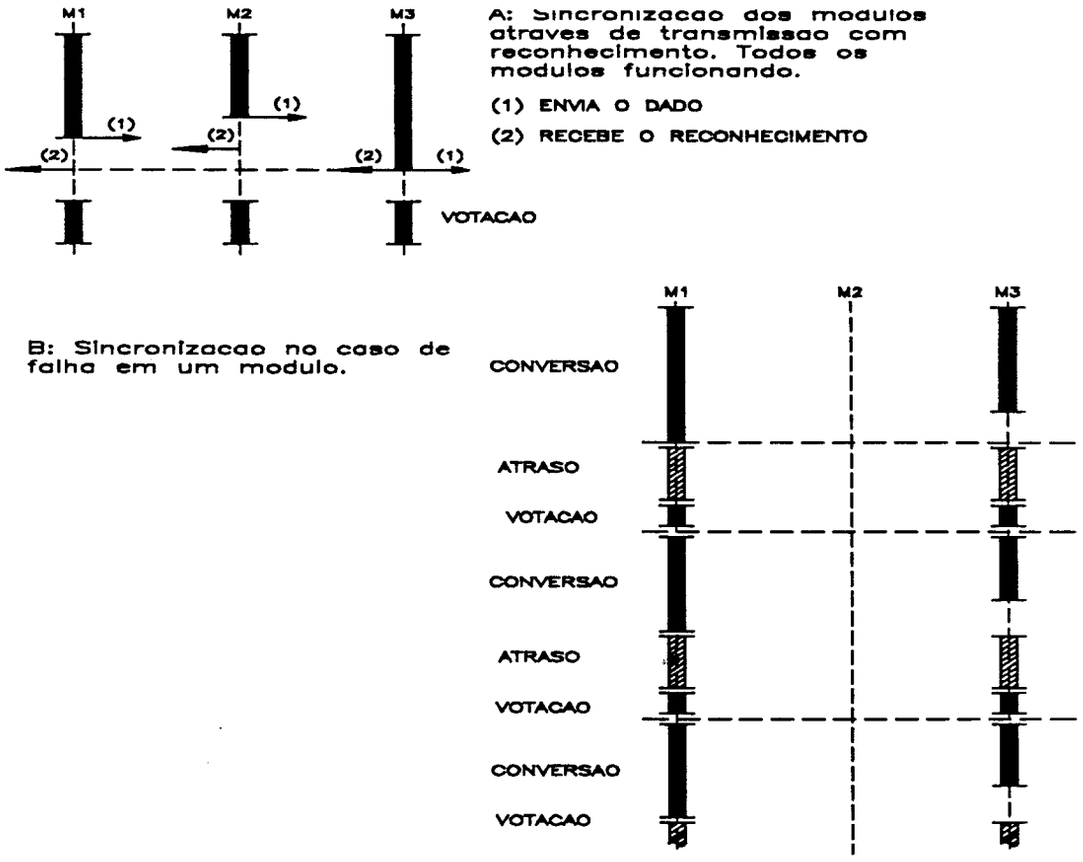


Figura 3.11: Sincronização no microcomputador tolerante a falhas com arquitetura em anel.

ou pela reinicialização do sistema.

3.2.6. Considerações sobre a confiabilidade e o desempenho do microcomputador tolerante a falhas, em anel.

O sistema implementado, pela utilização de módulos triplicados, é uma forma de redundância estática. A detecção dos erros e a recuperação da falha não são identificáveis na saída do sistema. Na realidade ocorre, como será mencionado adiante, uma degradação no desempenho do sistema com uma redução da velocidade de processamento, sem entretanto alterar as funções básicas do sistema.

A votação sobre um número parcial de dados, no caso a votação de 2 sobre 3 dados (comparação), nada mais é do que a implementação distribuida da votação majoritária, cuja função é:

$$M=AB+BC+AC$$

Quando uma falha ocorre, ela é detectada pela comparação dos dados em cada módulo.

A identificação do módulo que falhou pode ser feita através da análise dos sinalizadores de erro. Testes adicionais podem ser implementados para identificar o local de ocorrência da falha: se no processador, no subsistema de comunicação ou na transmissão dos dados. Na implementação realizada foi executado apenas o teste para verificar erros na transmissão.

Falhas maliciosas do tipo em que informações conflitantes, enviadas por um mesmo processador para processadores diferentes por rotas independentes [**FRISON e WENSLEY, 1982**], não ocorrem na transmissão de dados do sistema implementado. Neste sistema, sendo a circulação de dados unidirecional e a votação de dados parcial, não existe a possibilidade de envio de informações conflitantes para processadores distintos. No caso de circulação bidirecional, uma falha maliciosa poderia fazer com que os processadores não chegassem a um consenso.

Cada módulo recebe os sinalizadores de erro dos outros módulos e envia os seus sinalizadores para esses módulos. Através da configuração dos sinalizadores é identificado o módulo que venha a falhar e é escolhido o módulo que dará saída ao sistema. Falhas maliciosas podem ocorrer durante a transmissão dos sinalizadores de erro. Para minimizar a possibilidade de ocorrência deste tipo de falhas, utilizou-se um mecanismo de auto-teste da transmissão e dois bits para os sinalizadores de erro.

O funcionamento da arquitetura em anel foi verificado para várias situações de falhas simuladas nos módulos de

processamento, tendo-se obtido resultados satisfatórios.

O tempo de processamento aumenta quando um dos módulos do anel sofre uma falha. Esse aumento no tempo de processamento depende do número de ciclos de espera para sincronização que é ajustado (por programação) por um contador interno. Quando os módulos operam em completo sincronismo, não há perda de tempo com ciclos de espera.

Determinou-se quanto o sistema de gerenciamento e votação atrasa a saída de dados em relação ao sistema SIMPLEX. O tempo de gerenciamento foi determinado para os seguintes casos: operação normal do sistema e falha total de um módulo. No primeiro caso para um sinal de relógio de 4Mhz, o valor obtido foi de 360 µseg. e no segundo caso para o mesmo sinal de relógio e considerando que o contador faz 65635 contagens (pior caso) antes de sair do laço de contagem, o valor obtido foi de 3,4 seg. Este tempo pode, entretanto, ser reduzido diminuindo o valor do contador. Por exemplo para 10 contagens, o tempo cai para 792 µseg.

Alguns pontos importantes que devem ser ressaltados sobre a arquitetura implementada são: a facilidade de comunicação entre os módulos; a flexibilidade na implementação de algoritmos para a votação, já que esta é feita por programação, e a possibilidade de expandir o sistema para uma estrutura com maior número de módulos ("NMR") utilizando o esquema de sincronização frouxa.

O microcomputador tolerante a falhas, em anel, embora não tenha sido desenvolvido com o objetivo de uma aplicação imediata, é um "laboratório" para o estudo e desenvolvimento de novos mecanismos e sistemas tolerantes a falhas. Existem diversos detalhes que poderão ser melhor explorados e alguns deles são relacionados a seguir.

No sistema implementado, para efeito de falhas considerou-se

como fazendo parte do módulo o subsistema e o canal de comunicação de dados e de indicadores de erro. Embora na proposta inicial tenha sido previsto um subsistema de comunicação independente do módulo isso não foi realizado. Um dos trabalhos que poderão ser realizados é a implementação do subsistema de comunicação independente do módulo e a implementação da tolerância a falhas nesse subsistema.

A proposta da linha bastão para acesso a periféricos comuns de entradas também é um trabalho interessante que poderá ser realizado.

Outros trabalhos que poderão ser realizados estão relacionados com o aumento de desempenho da arquitetura, seja através de mecanismos mais ágeis para realizar os processos de sincronização, transmissão dos dados e acesso ao barramento de saída, seja pela utilização de circuitos e microprocessadores mais eficientes.

3.3. ARQUITETURA TOLERANTE A FALHAS DUPLEX-HÍBRIDA

3.3.1. Sistema DUPLEX

Uma das formas mais simples de se implementar a tolerância a falhas é através da utilização de dois módulos de processamento. Sistemas com essa configuração, referenciados como sistemas DUPLEX tem sido extensivamente tratados na literatura. Sistemas DUPLEX tem sido utilizados em diversas aplicações tais como centrais telefônicas automáticas [KRAFT e TOY, 1981; OSSFELDT e JONSSON, 1980], aplicações de alta segurança [THOMPSON e NEWSTEAD, 1982] e sistemas de uso geral [DELLACORNA et al., 1980].

Existem dois modos básicos de operação dos sistemas DUPLEX:

No primeiro modo, os dados dos dois módulos são continuamente comparados por um detetor de discordância (comparador) até que uma falha permanente seja detectada em um dos módulos. Quando isto ocorre, o módulo que falhou é identificado por um circuito testador externo, ou rotinas de auto-teste e é colocado fora de operação. O sistema procede então sua operação com o módulo que não apresentou defeito. O sistema cessa sua operação quando o segundo módulo falha. Este tipo de operação é chamado DUPLEX/SIMPLEX, representado por DS(t). Uma forma alternativa do DUPLEX/SIMPLEX opera com um módulo ativo e um módulo desenergizado (Figura 3.12). Nesta forma existe um subsistema externo aos módulos que verifica qual dos módulos está operando e seleciona a saída deste módulo como a saída do sistema. Quando o módulo ativo deixa de operar corretamente o supervisor aciona o outro módulo e seleciona a saída deste módulo como a nova saída do sistema. Os possíveis mecanismos que tem sido utilizados para sinalizar ao supervisor que a unidade ativa falhou são: o relógio cão de guarda ("watch-dog timer") [LU, 1982; PEARSON e PREECE, 1982] e a indicação de erro; neste último as unidades de processamento possuem circuitos e programas adicionais que monitoram as diversas partes funcionais da unidade sinalizando ao supervisor quando um erro irrecuperável é detectado. O relógio cão de guarda é implementado no circuito supervisor através de um contador que é zerado periodicamente pela unidade de processamento ativa quando esta unidade está operando corretamente. Antes de enviar o comando para zerar o contador, a unidade ativa verifica o seu estado interno através de um programa de auto-teste. A confiabilidade do sistema DUPLEX/SIMPLEX é dada pela

expressão:

$$R(DS) = R_M^2 - 2R_M\ (1-R_M)$$

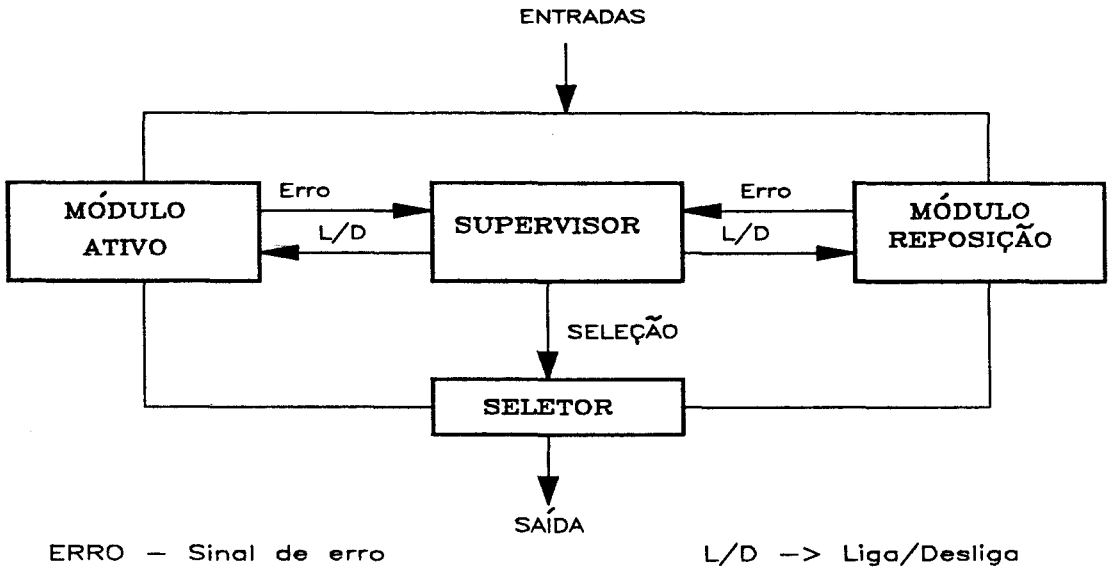


Figura 3.12: Sistema DUPLEX/SIMPLEX com chaveamento automático.

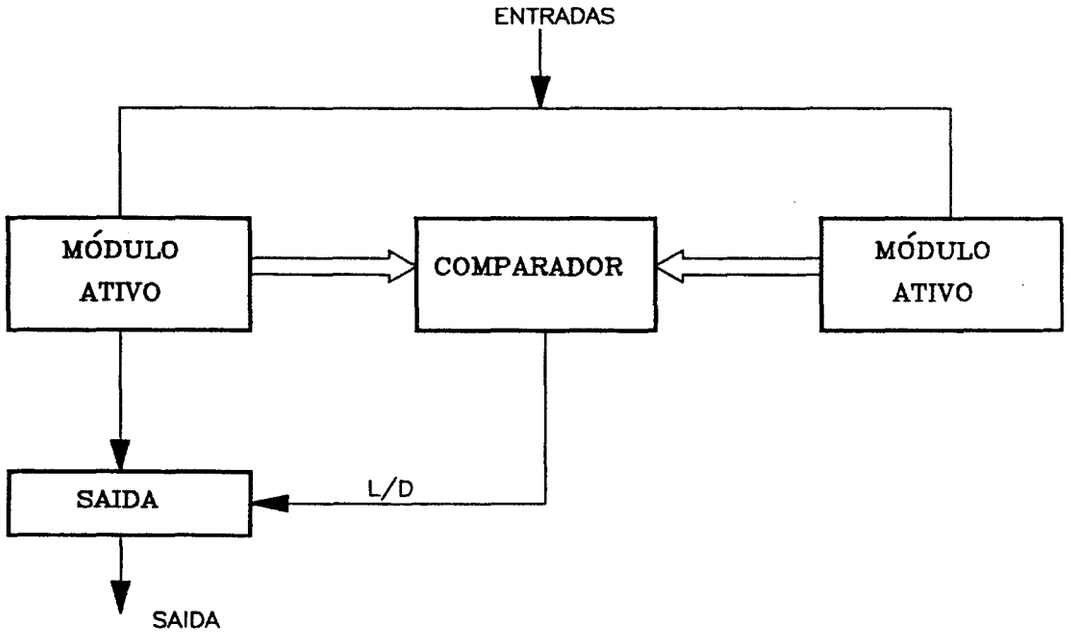
Considera-se que cada módulo tem uma taxa de falha constante λ, assim sendo a confiabilidade de cada módulo é:

$$R_m = e^{-\lambda t}$$

No outro modo, chamado simplesmente DUPLEX, representado por D(t), o sistema opera com dois módulos ativos, os dados destes módulos são continuamente comparados e no caso de uma falha permanente o sistema cessa sua operação (Figura 3.13).

Existem duas classes de sistemas DUPLEX com dois módulos ativos: sistemas fortemente conectados e sistemas fracamente conectados.

Nos sistemas fortemente conectados as unidades de processamento operam sincronamente, executando o mesmo programa a partir de uma base de tempos comum.



L/D -> Liga/Desliga

Figura 3.13: Sistema DUPLEX.

Em arquiteturas fracamente conectadas as unidades de processamento possuem o seu próprio relógio e não operam síncronamente. As duas unidades executam os mesmos programas ou programas equivalentes e em pontos específicos do programa mandam os dados calculados para o comparador. Este é o mesmo mecanismo utilizado no microcomputador tolerante a falhas com arquitetura em anel descrito na secção anterior. A vantagem da arquitetura fracamente conectada em relação à fortemente conectada consiste no fato de que na primeira não existem pontos de falha simples como o relógio e o circuito de comparação.

A confiabilidade do sistema DUPLEX é dada pela expressão:

$$R(D) = R_m^2$$

que é a confiabilidade dos dois módulos em série.

Em termos de confiabilidade, o DUPLEX/SIMPLEX é melhor

do que o DUPLEX normal, entretanto a detecção de um erro requer mecanismos adicionais de teste que muitas vezes são difíceis de serem implementados.

3.3.2. Arquiteturas DUPLEX-HÍBRIDAS.

O conceito de redundância híbrida introduzido por **MATHUR e AVIZIENIS [1970]** consiste em uma combinação de um sistema tolerante a falhas básico, contendo N módulos (Duplex, N=2; TMR, N=3; NMR, N módulos) associado a um conjunto de S módulos de reposição (figura 3.14). Quando for identificada a falha em um dos módulos do subsistema básico, este será substituido por um módulo idêntico disponível em um banco de módulos de reposição, retornando ao estado original. Assim cada módulo que falha é substituido por um dos módulos de reposição até que todos os módulos de reposição sejam utilizados. O estado final do sistema é o do sistema tolerante a falhas básico e a operação do sistema termina quando este último sistema falhar. Falhas transientes devem ter um tratamento especial no chaveamento dos módulos de reserva, para evitar que módulos em perfeito funcionamento sejam colocados fora de operação quando sujeitos a este tipo de falhas. Isto será abordado no item que se segue.

Neste trabalho será estudado o sistema DUPLEX/HÍBRIDO, representado por DH(S), desenvolvido a partir do sistema DUPLEX adicionando-se um banco com S módulos de reposição.

A análise da confiabilidade do sistema DH(S) foi realizada em trabalho anterior [OBAC-RODA, 1983] e está apresentada no apêndice B. Considerando uma configuração DH(S) em que os módulos do sistema DUPLEX tem taxa de falha constante λ e que os módulos de reserva são considerados inativos, tendo uma taxa de falha μ ($\mu<\lambda$) a expressão da confiabilidade do sistema DH(S)

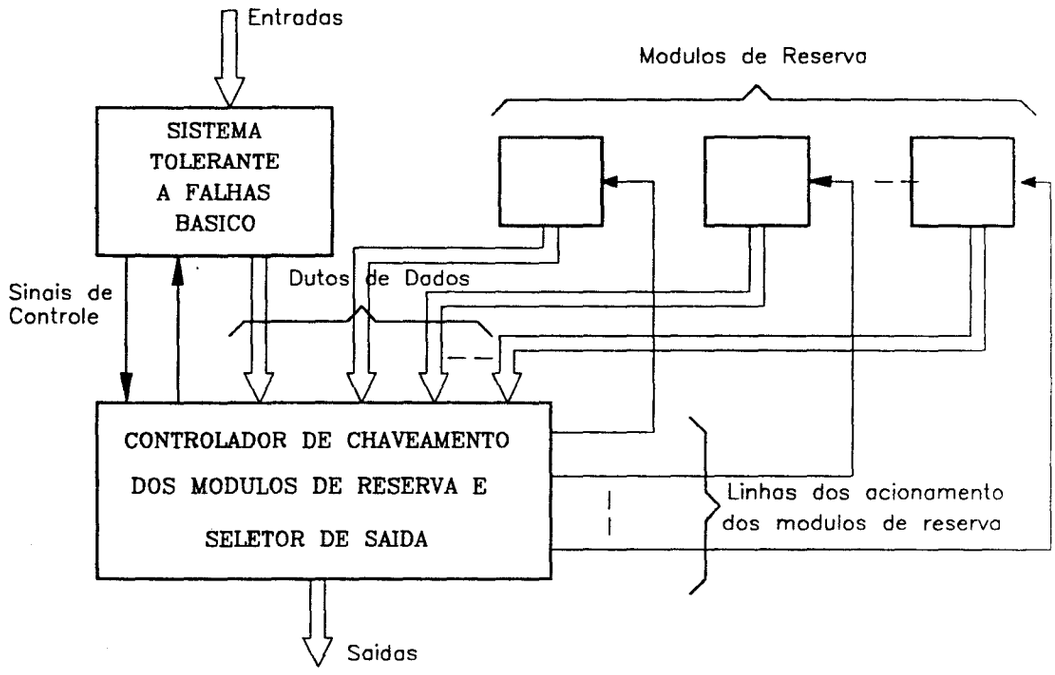


Figura 3.14: Configuração de sistemas tolerantes a falhas híbridos.

é dada por:

$$R(DH(S))(T) = R_M^2 . R_s^S(T) + \int_0^T e^{-(2\lambda+S\mu)t} . R(DH(S-1))(T-t)\,dt$$

onde:

R_m é a confiabilidade dos módulos ativos do sistema DUPLEX

R_s é a confiabilidade dos módulos de reserva

T é o tempo de missão (intervalo de tempo em que se espera que o sistema opere corretamente)

Observe-se que a expressão da confiabilidade do sistema DH(S) tem a forma de uma integral recursiva. As expressões da confiabilidade do sistema DH(S) com 1,2 e 3 módulos de reposição estão apresentadas no apêndice C.

Avaliou-se a confiabilidade de diversas configurações da

arquitetura DUPLEX/HÍBRIDA em função do tempo normalizado λt. Foi considerado que os módulos de reposição tem uma taxa de falha, μ, menor do que a taxa de falha dos módulos ativos, λ, tendo sido consideradas duas situações: $\mu = 0.1\lambda$ e $\mu = 0.5\lambda$. Esta consideração baseia-se no fato de que circuitos eletrônicos desligados tem uma taxa de falha menor do que quando estão operando, devido a efeitos relacionados com a temperatura de operação dos componentes. Os resultados desta avaliação estão mostrados na figura 3.15, onde também estão colocados para

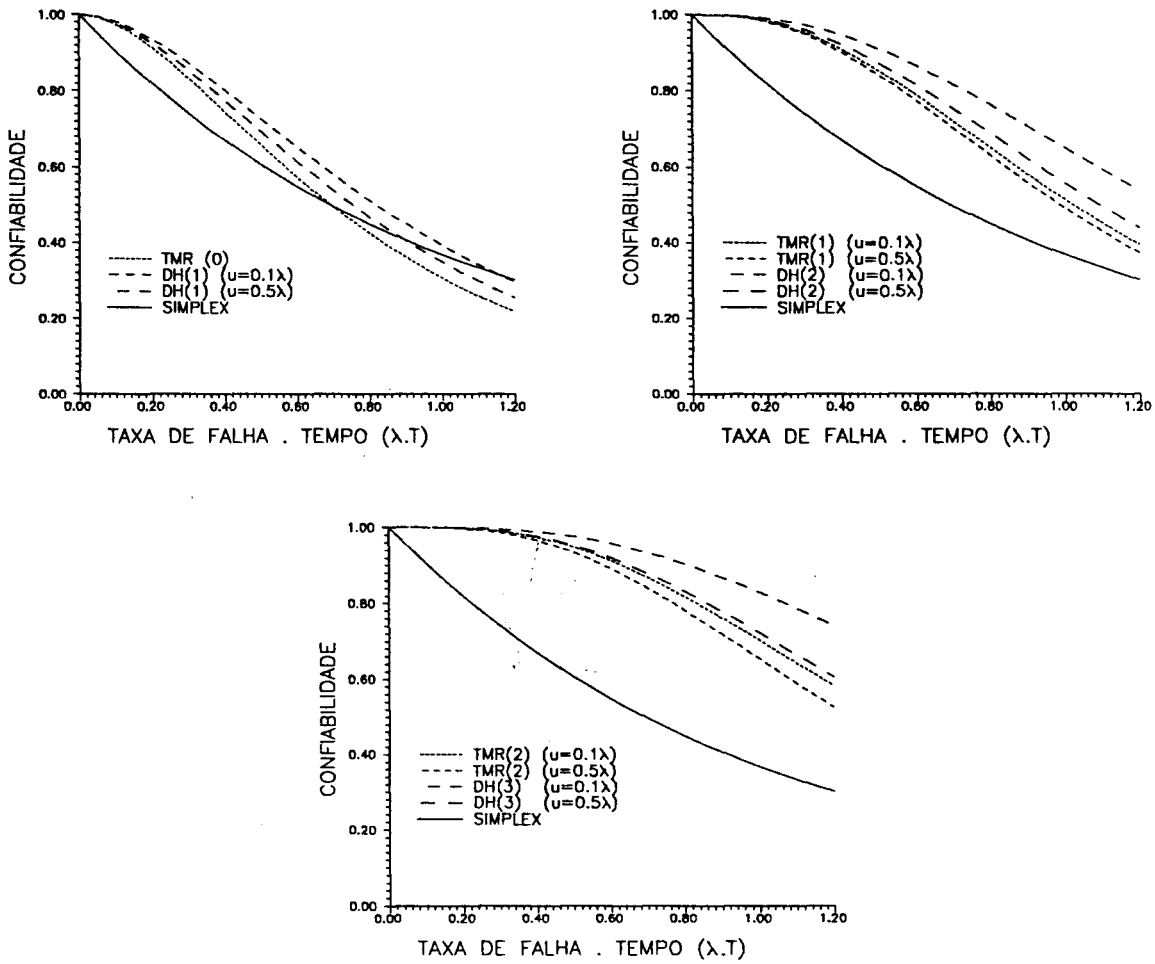


Figura 3.15: Confiabilidade em função do tempo normalizado λt para diversas configurações de sistemas com redundância modular tripla ("TMR") e sistemas DUPLEX/HÍBRIDOS.

comparação a confiabilidade do sistema SIMPLEX e do "TMR".

Pode-se observar claramente que a confiabilidade dos sistemas DUPLEX/HÍBRIDOS é superior à dos "TMR"s e que cresce à medida que maior números de módulos de reposição são utilizados.

3.3.3. Considerações sobre o chaveamento dos módulos de reposição em arquiteturas HÍBRIDAS.

Sistemas utilizando microcomputadores frequentemente operam em ambientes hostís onde estão sujeitos a interferências externas como distúrbios nas linhas de alimentação, pulsos eletromagnéticos, descargas eletrostáticas, etc. Este tipo de distúrbios, como visto no capítulo 2, pode causar defeitos nos circuitos resultando em falhas no sistema. Tres classes de defeitos podem ocorrer **[SOSNOWSKI, 1986a]**: transientes (curta duração), permanentes (longa duração) e fatais (destruição permanente (queima) de dispositivos semicondutores), estes últimos levando a uma disrupção total ou parcial da máquina. Defeitos transientes podem causar um desvio do fluxo de dados do programa que está sendo executado devido à alteração nos registradores internos ou transição incorreta de estados do microprocessador. Na prática estes defeitos podem também ocorrer por problemas de temporização e estado incorreto dos objetos controlados.

Os microcomputadores são máquinas sequenciais cuja operação pode ser descrita por diagramas de transição de estados. Por exemplo, o diagrama de transição de estados para o microprocessador 8085 **[INTEL, 1979]** é mostrado na figura 3.16.

Um determinado estado no diagrama depende do estado anterior e das entradas. Cada instrução executada passa por determinado número de estados que é característico desta instrução. Pode ser visto no diagrama como as linhas "READY",

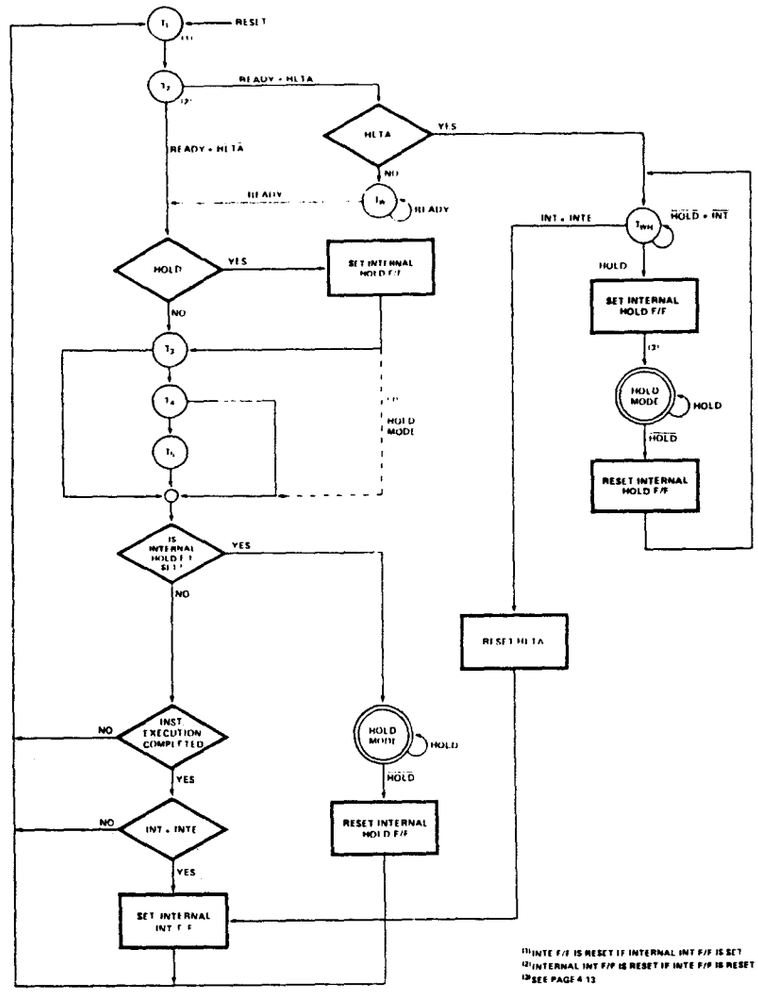


Figura 3.16: Diagrama de estados do microprocessador 8085 segundo **INTEL [1979]**.

"HOLD" e "INTERRUPT" são amostradas durante o ciclo de máquina e em que condições estas linhas modificam a sequência básica de transição. Transições indesejadas dentro do diagrama ,causadas por defeitos transientes podem resultar em falhas do sistema. Existe dentro do diagrama de estados do 8085 um estado que pode levar a uma situação permanente de erro. É o estado T_{WH} que só ocorre normalmente pela execução da instrução "HALT". Em geral, estados da máquina que não podem ser facilmente

abandonados são chamados estados armadilha. Quando o microprocessador entra no estado T_{WH} somente poderá sair por um comando de "RESET" ou por acionamento da interrupção. Em sistemas onde as interrupções não são acionadas, uma transição falsa para o estado T_{WH} devido a um defeito transiente leva ao travamento da máquina.

Máquinas sequenciais podem incorporar estados não utilizados (resultantes da codificação binária de estados) e defeitos transientes podem levar a esses estados errôneos. Caso a máquina não tenha sido bem projetada, esses estados podem ser estados armadilha que também levam ao travamento da máquina.

Sob outro ponto de vista, os microprocessadores são máquinas programáveis que podem realizar um número muito grande de funções. Essas funções são realizadas por um conjunto de máquinas sequenciais (sequenciador de programa, controlador de interrupção, etc) que executam um programa específico. Um erro nestas máquinas resulta em falhas na execução do programa. Por exemplo defeitos transientes no sequenciador de programa podem causar a interpretação errada de uma instrução ou a procura incorreta da próxima instrução. No caso de falhas do contador de programa e circuitos de endereçamento pode ocorrer a transferência da execução do programa para endereços incorretos, por exemplo na área de dados ou fora do espaço válido de memória.

Erros na execução de programas devido a defeitos transientes geralmente ocorrem por transições indesejadas de um ponto de execução do programa para um outro ponto qualquer dentro do espaço de endereços. De acordo com **GLASER e MASON** [1981], programas executados em pequenos sistemas podem ser classificados em dois tipos: aqueles que executam um cálculo ou função e terminam e aqueles que ficam continuamente em laço

("loop"). Estes últimos são bastante comuns em sistemas de tempo real, por exemplo sistemas de aquisição de dados e controladores.

No primeiro caso uma falha transiente pode levar a um erro de cálculo, execução errônea de uma função (por exemplo um comando errôneo de entrada ou saída de dados). No segundo caso a falha transiente pode fazer com que o fluxo do programa seja desviado para fora do laço principal levando a um travamento da máquina ou executando um laço falso dentro do espaço previsto do programa.

Consideremos o diagrama de fluxo da Figura 3.17. Uma transição falsa para o endereço **A1** inicia a execução de uma sequência de instruções que não são de desvio. No ponto **D** uma instrução de desvio é executada, que faz com que o programa retorne ao ponto **C1** dentro da sequência préviamente executada. O sistema permanece executando um laço sem fim, sómente podendo sair através de interrupção ou por um comando de "RESET". Como ilustração consideremos uma sequência de instruções executadas pelo microprocessador Z80:

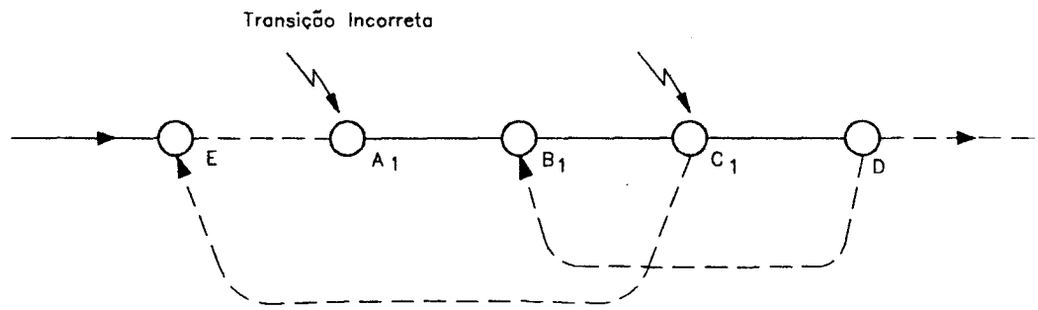


Figura 3.17: Criação de laços falsos dentro de uma sequência de instruções.

End.	Dados	código	progama	Código laço L1
.	.		.	
.	.		.	
.	.		.	
210B	3A	LD	A, (2AC0H)	
210C	C0			

```
210D    2A                          LD    HL,(0DBBEH)   <-┐
210E    BE    CP    (HL)                                  │
210F    DB    IN    A,06H                                 │
2110    06                          LD    B,0C2H          │
2111    C2    JP    NZ,532H                               │
2112    32                          LD    (3E05H),A       │
2113    05                                                  │
2114    3E    LD    A,0C3H                                │
2115    C3                          JP    210DH           └
2116    0D    DEC   C
2117    21    LD    HL,0CFCBH
2118    CB
2119    CF

  .       .           .
  .       .           .
  .       .           .
```

Uma transição falsa para um dos endereços 210D, 2110, 2112 ou 2115 força a execução de um falso laço entre os endereços 210D e 2114 e uma situação como esta levaria a um travamento do sistema.

Em sistemas com redundância híbrida, pode ocorrer a desativação de um módulo de operação correta pela ocorrência de uma falha transiente. Para evitar que isto ocorra, uma vez detectada uma falha o controlador do sistema deve disparar um processo de recuperação que identifica se a falha é transiente ou permanente e no caso desta última, identificar o módulo falho e substituí-lo por um módulo de reposição.

Existem básicamente tres classes de procedimentos para recuperação de falhas transientes [SOSNOWSKI, 1986b]: reinicialização do sistema, repetição da tarefa ("retry") e retorno a um ponto pré-definido do programa ("roll-back"). O procedimento de reinicialização é o mais drástico consistindo em reexecutar todo o processo (programa) desde o início, este procedimento tem sido utilizado em alguns sistemas de uso dedicado [PEARSON e PREECE, 1982]. A repetição de tarefas, embora mais eficiente do que a reinicialização, requer que os pontos de reexecução das

tarefas sejam muito bem identificados. O retorno a um ponto pré-definido é um dos procedimentos mais interessantes, neste procedimento estados intermediários de uma tarefa, aqueles que não implicam em mudanças de dados externos à máquina, são salvos e quando uma falha ocorre os programas são reexecutados a partir desses pontos. Este procedimento foi utilizado no desenvolvimento de uma arquitetura DUPLEX-HÍBRIDA que descrevemos a seguir.

3.3.4. Sistema Duplex Híbrido com repetição parcial de operações.

Em um sistema de computação, a unidade central de processamento coordena todas as atividades, sequenciando a execução das instruções armazenadas na memória de programa, coordenando as entradas de dados, manipulando os dados de maneira adequar a operação do microcomputador à sua aplicação e coordenando as saídas de dados.

A maior parte das instruções executadas manipulam dados internamente à unidade de processamento, por exemplo no microprocessador Z80, 56,3% das instruções são manipulações internas de dados, 25,5% das instruções utilizam dados externos à unidade de processamento e apenas 18,2% das instruções correspondem a operações de envio de dados para fora da unidade de processamento (envio de dados para a memória ou dispositivos de entrada/saída).

Uma manipulação incorreta de dados pelo processador sómente poderá ser detectada durante a saída de dados, por exemplo nas operações de endereçamento e nas operações de escrita nas memórias ou em periféricos de entrada e saída.

O estado do processador em um determinado instante pode

ser descrito pelo conteúdo dos registradores. Denominamos o estado em que o processador envia dados para o exterior de ***estado de saída***. Quando da ocorrência de um erro de processamento, existe o risco do processador enviar dados incorretos para a saída.

O procedimento de recuperação que utilizamos no microcomputador tolerante a falhas proposto, consiste em salvar o conteúdo dos registradores do processador a cada estado de saída. Caso ocorra uma falha de processamento, será possível recuperar o estado dos registradores e retornar ao estado de saída anterior. Este procedimento é importante para a recuperação do sistema no caso de falhas transientes.

Utilizando o princípio acima descrito, propusemos um microcomputador tolerante a falhas, cuja arquitetura está apresentada na figura 3.18. O microcomputador não foi implementado, entretanto vários dos subsistemas apresentados foram projetados e construídos e seu funcionamento verificado. A arquitetura proposta consiste dos seguintes subsistemas:

Memória com detecção e correção de erros utilizando o código de Hamming modificado. Esta memória está descrita na secção 3.1 deste capítulo, tendo sido testada e funcionado adequadamente.

Módulos idênticos de processamento, dos quais dois permanecerão ativos e os demais serão utilizados como módulos de reposição. Os módulos de processamento deverão ter geradores de relógio sincronizados, uma memória para armazenamento do estado de saída do processador e uma lógica de comparação para comparação dos dados com o outro módulo ativo. Na figura 3.18 está apresentado um sistema DUPLEX/HÍBRIDO com apenas 1 módulo de reposição.

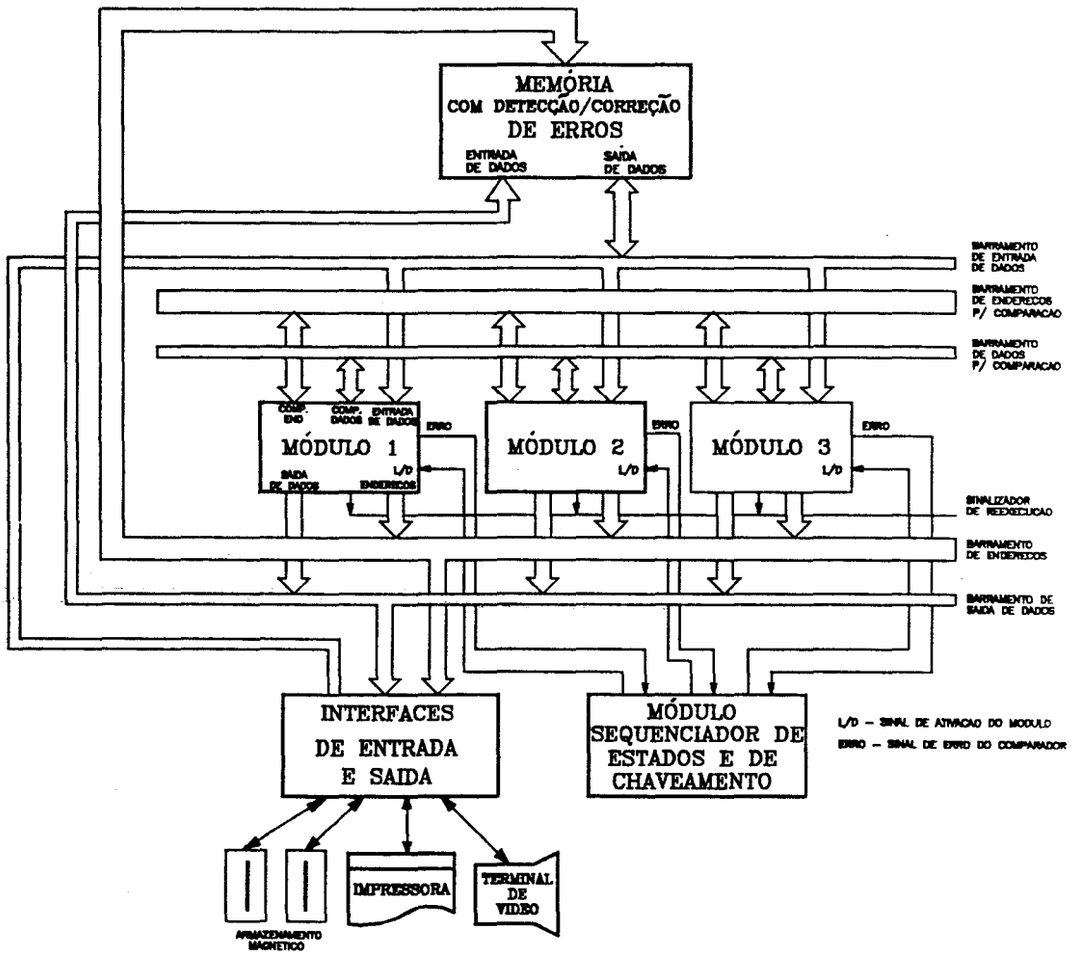


Figura 3.18: Arquitetura DUPLEX/HÍBRIDA com 1 módulo de reposição.

Um ***módulo com as interfaces para os periféricos de entrada e de saída.***

Um ***módulo sequenciador de estados e de chaveamento***. Este módulo determina a sequência de operação dos módulos ativos a partir de uma sequência inicial e dos sinalizadores de erro provenientes dos módulos ativos.

Utilizou-se redundância de módulos sómente na unidade de processamento, já que esta é a parte do microcomputador mais

ativa e mais vulnerável a erros.

Em operação normal, sómente um dos módulos ativos terá acesso aos barramentos de saída de dados e de endereços. Os dados provenientes da memória de dados ou periféricos de entrada alimentarão diretamente os módulos ativos através do barramento de entrada de dados. Foram criados dois barramentos secundários, um de dados e um de endereços. Estes barramentos secundários, chamados de barramentos de comparação, são utilizados por um circuito comparador para detecção de falhas no processamento dos módulos ativos. Os módulos de processamento são acionados pelo sinal L/D proveniente do módulo sequenciador de estados. Em caso de falha permanente é acionado o sinalizador ERRO, para que o módulo sequenciador ative o processo de recuperação e substituição do módulo falho.

A arquitetura do módulo de processamento está apresentada na figura 3.19. O microcomputador opera em configuração DUPLEX, com 2 módulos ativos (acionados pelo módulo sequenciador de estados). Um dos módulos endereça a memória e é responsável pelo envio dos dados de saída para a memória e para os periféricos de saída, este módulo será chamado módulo A. O outro módulo recebe os dados normalmente e envia os seus dados para os barramentos de comparação (de saída de dados e endereços). O módulo A compara síncronamente os dados e endereços por ele gerados com os dados e endereços gerados pelo segundo módulo. Qualquer discordância é sinalizada ao controlador de reexecução de tarefas pelos sinalizadores de erro de endereços (SEE) ou de dados (SED). Isto ocorre durante as operações de escrita e de endereçamento. Os circuitos de comparação de dados e de endereços foram testados em um microcomputador com o microprocessador Z80, tendo apresentado funcionamento satisfatório.

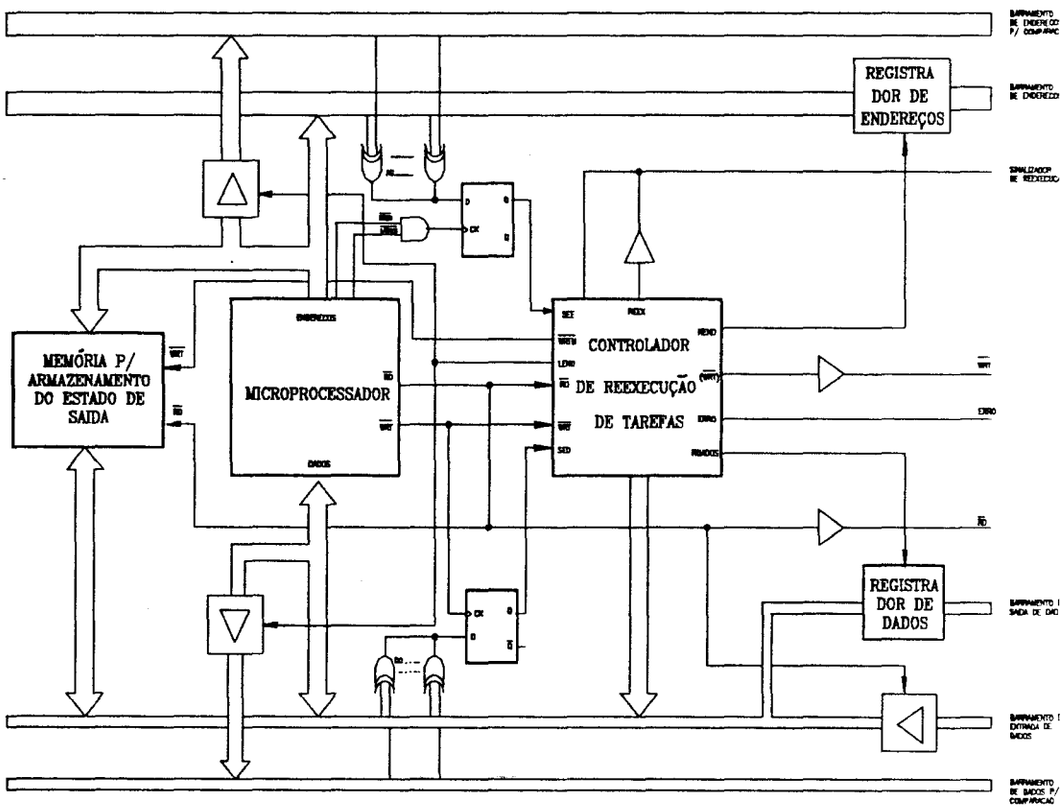


Figura 3.19: Arquitetura do módulo de módulo de processamento do sistema DUPLEX/HÍBRIDO.

O controlador de reexecução de tarefas é responsável pelo armazenamento do estado de saída do microprocessador e em caso de detecção de erro, pela reexecução da tarefa a partir do último estado de saída armazenado. O processo de salvamento do estado de saída é disparado pelo comando de escrita (WRT) do microprocessador. Quando o controlador de reexecução recebe este comando, ele ativa os registradores para que os dados e endereços sejam armazenados. A partir desse ponto ele gera uma sequência de instruções que são executadas pelo microprocessador para o armazenamento dos seus registradores em uma memória disponível para o armazenamento do estado de saída. Uma possível sequência de instruções para o armazenamento do estado

de saída do microprocessador Z80 é a seguinte:

```
SALVA:                  ;*** ROTINA PARA ARMAZENAMENTO DO ESTADO DE
                        ;*** SAIDA DO MICROPROCESSADOR
                        ;
        LD      (01EH),SP   ;ARMAZENA O PONTEIRO DA PILHA NA POSIÇÃO 020
                            ;DA MEMÓRIA
        LD      SP,01EH     ;CARREGA O PONTEIRO DA PILHA COM 020H
        PUSH    AF          ;SALVA ACUMULADOR E SINALIZADORES NA PILHA
        EX      AF,AF'      ;TROCA CONJUNTO DE REGISTRADORES
        PUSH    AF          ;SALVA SEGUNDO CONJUNTO DE ACUMULADOR E
                            ;SINALIZADORES
        EX      AF,AF'      ;TROCA CONJUNTO DE REGISTRADORES
        PUSH    BC          ;SALVA PAR DE REGISTRADORES BC
        PUSH    DE          ;SALVA PAR DE REGISTRADORES DE
        PUSH    HL          ;SALVA PAR DE REGISTRADORES HL
        EXX                 ;TROCA CONJUNTO DE REGISTRADORES
        PUSH    BC          ;SALVA PAR DE REGISTRADORES BC
        PUSH    DE          ;SALVA PAR DE REGISTRADORES DE
        PUSH    HL          ;SALVA PAR DE REGISTRADORES HL
        EXX                 ;TROCA CONJUNTO DE REGISTRADORES
        PUSH    IX          ;SALVA REGISTRADOR INDICE IX
        PUSH    IY          ;SALVA REGISTRADOR INDICE IY
        LD      A,I         ;COLOCA VETOR DE INTERRUPÇÃO NO ACUMULADOR
        PUSH    AF          ;SALVA VETOR DE INTERRUPÇÃO
        LD      SP,(O1EH)   ;RECUPERA O PONTEIRO DA PILHA
```

Esta rotina armazena o conteúdo de todos os registradores do microprocessador, menos o contador de programa, em uma sequência decrescente de endereços de memória a partir do endereço 01FH. Para armazenar o contador de programa a rotina de recuperação podera ser chamada através da instrução CALL, armazenando o conteúdo atual do registrador de programa na memória de armazenamento do estado de saída que funcionará como uma estrutura de pilha.

Terminada a execução da rotina que salva os registradores, o controlador de reexecução gera o comando de escrita (WRT) e libera os dados e endereços de saída. Tudo se passa como se houvesse um retardo no processo de escrita pelo microprocessador.

Caso seja detectado um erro em uma operação de endereçamento ou de saída de dados, indicado pelos sinalizadores de erro, o controlador de reexecução inicia a recuperação dos valores dos registradores do microprocessador. O controlador de reexecução sinaliza este procedimento através do sinalizador REEX para que o outro módulo ativo inicie também a recuperação dos registradores. O procedimento de recuperação dos registradores é similar ao do armazenamento do estado de saída. O microprocessador gera uma sequência de instruções que recuperam o estado de saída do microprocessador. Esta sequência de instruções para o microprocessador Z80 pode ser a seguinte:

```
REC:                    ;*** ROTINA PARA RECUPERAÇÃO DO ESTADO DE
                        ;*** SAIDA DO MICROPROCESSADOR

        LD      SP,08H      ;CARREGA O PONTEIRO DA PILHA COM 020H
        POP     AF          ;RECUPERA O VETOR
        LD      I,A         ;DE INTERRUPÇÃO
        POP     IY          ;RECUPERA O REGISTRADOR INDICE IX
        POP     IY          ;RECUPERA O REGISTRADOR INDICE IY
        EXX                 ;TROCA CONJUNTO DE REGISTRADORES
        POP     HL          ;RECUPERA PAR DE REGISTRADORES HL
        POP     DE          ;RECUPERA PAR DE REGISTRADORES DE
        POP     BC          ;RECUPERA PAR DE REGISTRADORES BC
        EXX                 ;TROCA CONJUNTO DE REGISTRADORES
        POP     HL          ;RECUPERA SEGUNDO PAR DE REGISTRADORES HL
        POP     DE          ;RECUPERA SEGUNDO PAR DE REGISTRADORES DE
        POP     BC          ;RECUPERA SEGUNDO PAR DE REGISTRADORES BC
        EX      AF,AF'      ;TROCA CONJUNTO DE REGISTRADORES
        POP     AF          ;RECUPERA SEGUNDO CONJUNTO DE ACUMULADOR
                            ;E SINALIZADORES
        EX      AF,AF'      ;TROCA CONJUNTO DE REGISTRADORES
        POP     AF          ;RECUPERA ACUMULADOR E SINALIZADORES
        LD      SP,(01EH)   ;RECUPERA O PONTEIRO DA PILHA
```

Terminada a recuperação do estado de saída, os módulos repetem o processamento a partir deste estado. Caso a indicação de erro persista, o controlador de reexecução pode ativar a recuperação do estado de saída por um número determinado de tentativas. No caso de uma falha transiente após desaparecer a

causa da falha, não ocorrerá mais a indicação de erro e os módulos poderão proceder normalmente à execução do programa. No caso de uma falha permanente, não desaparecendo a indicação de erro, o controlador de reexecução sinaliza este fato através do sinalizador ERRO. O módulo sequenciador de estados inicia então o processo de identificação do módulo com defeito e sua substituição por um módulo de reposição. O mecanismo de salvamento e recuperação do estado de saída foi testado com sucesso em um microcomputador baseado no microprocessador Z80. O controlador de reexecução de tarefas não foi implementado. Sendo entretanto, um circuito sequencial com um número não muito grande de estados, este controlador poderá ser implementado com redes lógicas programáveis tipo "PALs" ("Programmable Array Logic") ou circuitos similares.

Da maneira como foi proposto o mecanismo de recuperação de falhas transientes, com o armazenamento do estado de saída do microprocessador a cada saída de dados, a velocidade de operação do sistema deverá ser consideravelmente reduzida. Alguns dos mecanismos propostos, entretanto, poderão ser incorporados ao circuito do microprocessador. Por exemplo a memória de armazenamento de dados poderia ser um conjunto alternativo de registradores, de maneira que o armazenamento e a recuperação do estado de saída pudessem ser uma operação interna do microprocessador. A eficiência seria consideravelmente aumentada, e haveria uma diminuição do número de componentes externos ao sistema, que fazem parte do controlador de reexecução de tarefas. Também os comparadores de dados e de endereços poderiam ser implementados internamente ao microprocessador. Estes são alguns dos trabalhos que poderiam ser realizados na área de projeto e implementação de microprocessadores integrados em VLSI com mecanismos internos de tolerância a falhas.

3.4. CONCLUSÕES.

Neste capítulo são apresentadas algumas contribuições que foram feitas para o desenvolvimento de sistemas mais confiáveis com microprocessadores.

Inicialmente apresentamos alguns conceitos básicos sobre a confiabilidade de memórias a semicondutor e mostramos como a confiabilidade dessas memórias pode ser melhorada através de redundância de informação. Foi também apresentado e descrito o funcionamento dos códigos detetores e corretores de erros através dos quais é possível detectar e corrigir erros nas memórias. Utilizando esses conceitos, descrevemos a implementação de dois subsistemas de memória com detecção e correção de erros que podem ser utilizados em microcomputadores ou controladores de pequeno porte. Embora existam circuitos integrados dedicados para a correção e detecção de erros em memórias, como o 8206 da **INTEL, [1990]** e os 'ALS632 a 'ALS635 da Texas Instruments **[TEXAS, 1989]**, devido ao custo alto destes dispositivos a implementação desta técnica utilizando lógica discreta é ainda interessante. Por conveniência de implementação utilizou-se na memória construida cicuitos lógicos com tecnologia TTL. Sua implementação, entretanto, com redes lógicas programáveis tipo "PAL"s ("Programmable Array Logic") reduzirá a quantidade de componentes utilizados, contribuindo para o aumento da confiabilidade.

O aumento da densidade das memórias e a utilização de memórias com maior capacidade de armazenamento tem reavivado a importância da utilização de técnicas para o aumento da confiabilidade nesses subsistemas. Isto pode ser observado nos cuidados que atualmente alguns fabricantes de microcomputadores tem utilizado. Como por exemplo, a IBM utiliza nos seus

microcomputadores das linhas XT e AT o recurso do ***bit de paridade*** para detectar a ocorrência de erros nas memórias, sem entretanto corrigí-los. Recentemente, os fabricantes de microcomputadores de maior porte (super-micros), estão se preocupando não só com a detecção de erros, mas também com a correção destes. Estes microcomputadores utilizam memórias de grande capacidade, da ordem de dezenas de mega-bytes, que estão mais sujeitas a apresentarem falhas.

Como exemplo de arquitetura tolerante a falhas, apresentamos um microcomputador tolerante a falhas com arquitetura em anel. Este microcomputador utiliza redundância modular tripla, podendo tolerar falhas em um dos módulos de processamento. A detecção da falha é realizada pela comparação de dados de módulos adjacentes, processo este que denominamos votação parcial de dados. Procurou-se na implementação eliminar possíveis pontos comuns vulneráveis a falhas pela redução dos circuitos lógicos externos aos módulos. Dessa forma, o circuito de comunicação de dados é parte integrante dos módulos e os mecanismos de comparação de dados e de seleção do módulo de saída foram implementados em programação em cada módulo. A arquitetura desenvolvida é potencialmente interessante para aplicações em tempo real, pela relativa simplicidade de implementação e pela flexibilidade de implementação dos algoritmos de comparação de dados por programação, que são necessários para comparar entradas analógicas.

Finalmente introduziu-se o conceito de arquiteturas híbridas, desenvolvidas a partir de um sistema tolerante a falhas básico adicionando um conjunto de módulos de reposição que substituem os módulos que venham a falhar. Um dos problemas destas arquiteturas é o chaveamento de um módulo de funcionamento correto no caso de ocorrência de uma falha transiente. Alguns dos

principais efeitos de falhas transientes na operação de sistemas com microprocessadores foram apresentados. Com base nesses efeitos, foi proposta uma arquitetura para um sistema tolerante a falhas DUPLEX/HÍBRIDA, que incorpora um mecanismo de recuperação a falhas transientes.

CAPÍTULO 4

CONCLUSÕES FINAIS E PERSPECTIVAS FUTURAS

Nos capítulos anteriores foram colocadas algumas conclusões parciais com o intuíto de resumir os aspectos abordados e ressaltar os pontos relevantes deste trabalho. Para prover uma visão unificada dos problemas abordados, associados com a confiabilidade em sistemas com microprocessadores, apresentamos neste capítulo algumas conclusões gerais, conclusões estas baseadas nos capítulos prévios.

As perspectivas de utilização de técnicas para o aumento da confiabilidade em sistemas com microcomputadores são extremamente promissoras, haja visto a crescente utilização desses sistemas em grande número de aplicações críticas, onde a falha do equipamento pode trazer graves consequências econômicas ou de risco à vida humana. Em algumas destas aplicações, a operação do computador não pode ser interrompida nem para uma simples manutenção.

Dois aspectos importantes foram analisados no presente trabalho, como contribuição para o estudo e desenvolvimento de sistemas confiáveis com microcomputadores: o estudo da interferência externa, particularmente dos distúrbios da rede elétrica, na operação de microcomputadores e a utilização de técnicas de redundância como meio de aumentar a confiabilidade desses sistemas.

O suprimento de energia elétrica fornecida a sistemas de computação não é, na maioria dos casos, de qualidade compatível com a exigida por esses sistemas. Distúrbios elétricos provenientes da própria rede elétrica, ou induzidos (como interferência eletromagnética ou causados por perturbações atmosféricas) podem

exceder, em muito, os limites de tolerância do computador. Como tentativa de monitorar esses distúrbios foi construido um monitor de transientes de rede dedicado, com o qual foram feitos alguns levantamentos sobre a qualidade do suprimento de energia elétrica em instalações do Departamento de Física e Ciência dos Materiais/USP. Verificou-se a existência de uma correlação entre os dados levantados em dois ambientes diferentes e observou-se a ocorrência de distúrbios com níveis de tensão algumas vezes superiores ao valor nominal da tensão da rede. Procurou-se a concessionária local de eletricidade para obter informações sobre as normas existentes para o fornecimento de energia e sobre estatísticas de ocorrência dos distúrbios. Verificou-se serem escassas essas informações. As normas existentes, assim como alguns dados obtidos, referem-se a interrupções de longa duração. Esse tipo de interrupções paralisa a operação dos sistemas de computação a menos que medidas efetivas sejam tomadas para continuar o suprimento de energia durante a interrupção, como por exemplo com o uso de fontes uninterruptiveis de tensão. A concessionária local não dispunha, ou tinha conhecimento de dados sobre distúrbios de curta duração, que eram de interesse para o presente estudo.

Pela inexistência de dados, o levantamento e a caracterização da qualidade do suprimento de energia elétrica em ambientes onde irão operar sistemas de computação, é uma proposta interessante. Com esse intuito, foi proposto pelo presente autor e está sendo desenvolvido sob a forma de trabalho orientado de mestrado um monitor microprocessado de distúrbios da rede. Este equipamento deverá permitir uma coleta de dados mais completa, para caracterização do suprimento de energia elétrica.

O conhecimento do efeito interferidor de distúrbios externos, sejam eles de natureza elétrica ou eletromagnética, sobre o

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC
FISICA

comportamento de microcomputadores é de grande importância para o desenvolvimento de esquemas de proteção contra as falhas causadas por esses distúrbios.

Com o objetivo de estudar o efeito dos distúrbios elétricos em sistemas com microcomputadores, foi desenvolvido um sistema de testes dedicado. A observação do efeito dos distúrbios é feita comparando-se o comportamento de um equipamento sujeito aos distúrbios, com uma réplica idêntica considerada como padrão, ambos executando simultâneamente a mesma função. Utilizando o equipamento de testes desenvolvido, ensaiou-de um microcomputador dedicado submetendo-o a diversas sequências de distúrbios elétricos, com os parametros de amplitude e duração controlados. Foram obtidos resultados bastante interessantes que podem ser resumidos da seguinte forma: distúrbios elétricos com amplitude positiva (níveis de tensão maiores do que o valor nominal de alimentação do microcomputador) podem ter, dependendo da amplitude, efeito destrutivo sobre o sistema levando à destruição de componentes semicondutores; distúrbios negativos (níveis de tensão menores do que o valor nominal de tensão do microcomputador) tem efeito disruptivo, causando alterações no fluxo de processamento que podem causar erros no processamento ou o travamento da máquina.

O estudo desenvolvido sobre o comportamento de microcomputadores sujeitos aos distúrbios da rede abre, portanto, perspectivas interessantes de novos trabalhos que poderão ser desenvolvidos como continuação. Entre esses trabalhos podemos citar os seguintes:

O ensaio de computadores comerciais para determinar seu nível de susceptibilidade aos distúrbios da rede.

O estudo sobre a utilização de medidas e equipamentos custo-efetivos para adequar o suprimento de energia elétrica aos sistemas de computação.

O estudo e implementação de técnicas de circuitos e de programação para diminuir a susceptibilidade dos microcomputadores aos distúrbios elétricos.

Como complementação a este trabalho, está sendo iniciado um estudo sobre o efeito da interferência eletromagnética no comportamento de microcomputadores. O sistema de testes que está sendo desenvolvido segue a mesma filosofia do sistema de testes utilizado para verificar o comportamento de microcomputadores sob distúrbios elétricos. Um esquema geral é apresentado na figura 4.1. Para os testes serão utilizados dois módulos microcomputadores idênticos, operando sincronamente e rodando o mesmo programa. Um dos módulos sob teste será mantido no interior de uma câmara com blindagem eletromagnética, onde os distúrbios serão gerados pelo faiscamento obtido a partir de uma bobina de alta tensão. A metodologia de obtenção dos dados será a mesma utilizada no estudo da interferência dos distúrbios elétricos.

Na segunda parte do trabalho foi analisada a utilização de redundância para o aumento da confiabilidade de sistemas com microprocessadores. Duas formas de redundância foram analisadas: a redundância de informação para o aumento da confiabilide de memórias e a replicação de módulos de processamento como meio de mascarar falhas de circuitos ou programas.

O aumento da confiabilidade de memórias sob a forma de utilização de códigos detetores/corretores de erros, conhecidos como

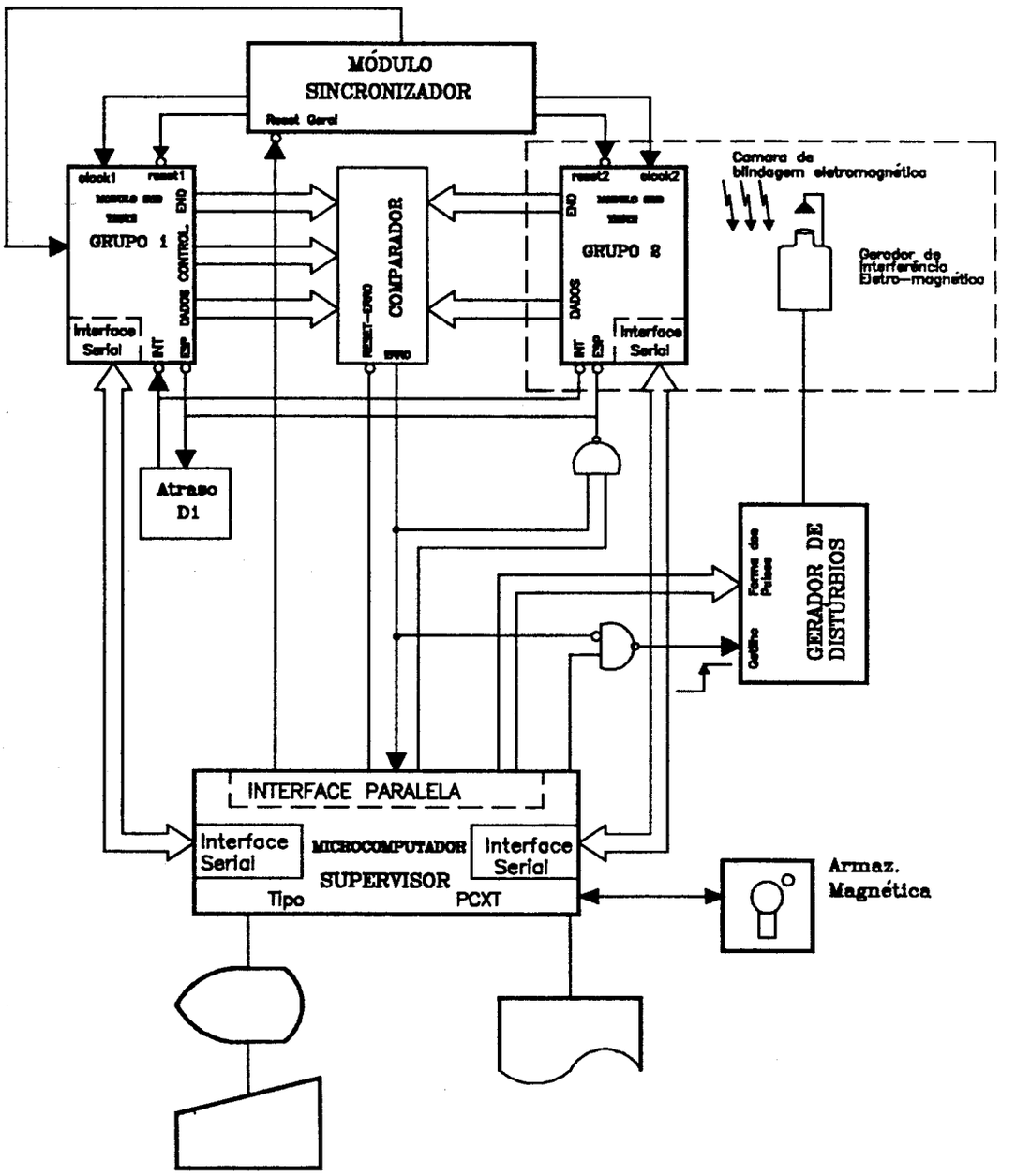


Figura 4.1: Arquitetura do sistema para testes do efeito de interferência eletromagnética sobre a operação de microcomputadores.

códigos de Hamming, foi apresentada sob a forma de implementação de memórias de pequeno porte utilizando essas técnicas.

O aumento de confiabilidade de sistemas de computação pode ser conseguido pela replicação de recursos a nível de circuitos ou programação. Os recursos replicados são utilizados de maneira que quando uma parte do sistema falhe, ela seja automáticamente substituida por uma outra com características similares. Esta técnica é conhecida como tolerância a falhas, tendo sido utilizada no desenvolvimento de um micro computador tolerante a falhas com módulos triplicados (arquitetura em anel) e na proposta de uma arquitetura tolerante a falhas baseada em duplicação de módulos que são substituidos, em caso de falha, por módulos idênticos de reposição (arquitetura DUPLEX/HÍBRIDA).

Diversos aspectos de implementação e da confiabilidade das arquiteturas tolerante a falhas, em anel e DUPLEX/HÍBRIDA foram apresentados, mostrando sua viabilidade e o ganho de confiabilidade desses sistemas em relação a outros que não utilizam redundância.

Um outro aspecto relevante apresentado foi o de recuperação de falhas transientes em microprocessadores.

A confiabilidade de sistemas utilizando microprocessadores está diretamente ligada à confiabilidade dos componentes utilizados (indicada pela taxa de falha destes componentes). A confiabilidade dos componentes por sua vez é determinada por parâmetros construtivos e pelo número de elementos que são integrados em um mesmo invólucro no caso de circuitos integrados. Uma regra geral é que quanto menor o número de componentes mais confiável é o sistema. O aumento da densidade dos novos circuitos integrados através de técnicas de integração de larga escala é um dos fatores determinantes para o aumento da confiabilidade de sistemas. Entretanto, pela crescente utilização de sistemas com microprocessadores nas mais diversas áreas, surgem dia a dia requisitos para que estas máquinas tenham maior

capacidade de processamento e de armazenamento. Isto requer circuitos cada vez mais densos e complexos e programas maiores e mais bem elaborados. Esses fatores contribuem negativamente para a confiabilidade desses equipamentos. As técnicas para o aumento da confiabilidade tem sido normalmente utilizadas em aplicações críticas que exigem a operação correta e segura do sistema de computação. É previsivel entretanto, que o aumento crescente da complexidade dos sistemas implique no aumento da utilização destas técnicas em sistemas de uso mais geral. Isto provavelmente deverá ocorrer a nível de circuitos integrados, com memórias incorporando técnicas de redundância de informação e processadores e periféricos utilizando esquemas de proteção e correção de erros. Isto abre perspectivas interessantes de trabalho para a área de confiabilidade de sistemas.

Desejamos que as contribuições apresentadas nesta tese possam estimular o desenvolvimento de novos trabalhos e concientizar pessoas ligadas à area de informática sobre a importância da confiabilidade em sistemas de computação.

REFERÊNCIAS BIBLIOGRÁFICAS

- **[ALLEN e SEGALL, 1974]** Allen, G.W. e Segall, D.: "Monitoring of computer installations for power line disturbances". Proc. IEEE Power Engineering Society Winter Meeting, pp. 199-206, 1974.

- **[ALTNETHER, 1979]** Altnether, J.: "Error Detecting and Correcting Codes - Part 1", INTEL, AP-46, 1979.

- **[ANDERSON e LEE, 1982]** Anderson, T. e Lee, P.A.: "Fault Tolerance Terminology Proposals", Dig. 1982, Int. Symp. Fault Tolerant Computing, Santa Monica (CA), pp. 29-33, 1982.

- **[AVIZIENIS, 1975]** Avizienis, A.L.: "Architecture of Fault Tolerant Computing Systems", Dig. 1975, Int. Symp. Fault Tolerant Computing, Paris, pp. 3-16, Jun. 1975.

- **[AVIZIENIS, 1978]** Avizienis, A.L.: "Fault-tolerance: The Survival Attribute of Digital Systems", Proc. of the IEEE, Vol 66, no. 10, pp. 1109-1125, Oct. 1978.

- **[AVIZIENIS, 1982]** Avizienis, A.L.: "The Four-Universe Information System Model for the Study of Fault-Tolerance", Dig. 1982, Int. Symp. Fault Tolerant Computing, Santa Monica (CA), pp. 6-13, 1982.

- **[AVIZIENIS, 1986]** Avizienis, A.L: "Approaches to Computer Reliability - Then and now". Proc. of the AFIPS, vol. 45, pp. 401-411, 1976.

- **[BABAOGLU e DRUMMOND, 1987]** Babaoglu , O and Drummond, R.: "(Almost) No Cost Clock Synchronization", 17th. Int. Symp. on Fault-Tolerant Computing; Pittsburgh, pp. 42-47, 1987.

- **[CHANDE et al., 1989]** Chande, P.K.; Ramani, A.K. and Sharma, P.C.: "Modular TMR Multiprocessor System", IEEE Trans. on Industrial Electronics; Vol. 36, no. 1, pp. 34-41, 1989.

- **[CLARK, 1985]** Clark, J.: "Cost-Effective Computer Backup Systems Solve Power Supply Problems", Computer Technology Review, Vol. V, no. 2, pp. 201-205, 1985.

- **[CORTES e Mc CLUSKEY, 1986]** Cortes, M.L. e Mc Cluskey, E.J.: " An experiment on intermittent failure mechanisms", IEEE Proc. of International Test Conference, pp. 435-442, 1986.

- **[CROSTHWAIT, 1985]** Crosthwait Jr., D.L.: "UPS Products Provide Improved Performance and Greater Flexibility", Computer Technology Review, Vol V, no. 3, pp. 208-212, 1985.

- **[DAVIES e WAKERLY, 1978]** Davies, D. and Wakerly, J.F.: "Synchronization and Matching in Redundant Systems", IEEE Trans. on Computers, Vol. C-27, no. 6, pp. 531-539, 1978.

- **[DELLACORNA et al., 1980]** Dellacorna, L.; Morgantin, M. e Novielli, G.: "A microprocessor based control unit for high availability applications". Dig. of the 1980 Int. Symp. Fault Tolerant Computing, Kyoto, pp. 357-362, 1980.

- **[D.O.U., 1978a]** Diário Oficial da União - Serviço Público

Federal: "Portaria no. 046 de 17 de abril de 1978". D.O.U. 26.04.78, pp. 180-190.

- **[D.O.U., 1978b]** Diário Oficial da União, Serviço Público Federal: "Portaria no. 047 de 17 de abril de 1978". D.O.U. 26.04.78, pp. 191-199.

- **[FANCHER, 1985]** Fancher, D.: "Power Line Monitors Provide Data Useful in Error Correction", Computer Technology Review, vol. V, no. 2, pp. 207-213, 1985.

- **[FISCHER, 1990]** Fischer, D.P.M.: "Desenvolvimento de um Sistema Microcomputador Tolerante a Falhas com Arquitetura em Anel". Tese de Doutoramento, Instituto de Física e Química de São Carlos/USP, 171p, 1990.

- **[FRAGA e LEMOS, 1987]** Fraga, J. e Lemos, R.: "Conceituação e Terminologia em Sistemas Informáticos Tolerantes a Falhas e Instrusões", Anais do 2o. Simpósio em Sistemas de Computadores Tolerantes a Falhas, pp. 177-188, 1987.

- **[FRISON e WENSLEY, 1982]** Frison, S.G. and Wensley, J.A.: "Interactive Consistency and its Impact on the Design of TMR Systems", Dig. 12th Int. Symp. on Fault Tolerant Computing, Santa Monica, pp. 228-233, 1982.

- **[GERKE e KIMMEL, 1986]** Gerke, D. and Kimmel, W.: "Designing Noise Tolerance into Microprocessor Systems", EMC Technology, Vol. 5, no. 2, pp. 45-52, 1986.

- **[GLASER e MASON, 1981]** Glaser, R.E. e Mason, G.M.: "The

containment set approach to upsets in digital systems". IEEE Trans. on Comp., Vol C-31, no. 7.

- **[GORDON e NADIG, 1977]** Gordon, G. and Nadig, H.: "Hexadecimal Signature Identify Troublespots in Microprocessor Systems", Electronics, Vol. 50, no. 5, pp. 89-96, 1977.

- **[HOGAN, 1983]** Hogan, T.: "CP/M Guia do Usuário", Mc Graw Hill, 317pp., 1983.

- **[HOROWITZ e HILL, 1989]** Horowitz, P. and Hill, W.: "The Art of Electronics", 2nd ed. Cambridge University Press, Cambridge, 1125pp., 1989.

- **[INTEL, 1979]** Intel Corporation: "MCS - 80/85™ Family User's Manual", Santa Clara, U.S.A., 1979.

- **[INTEL, 1990]** Intel Corporation: "Peripheral Data book", Santa Clara, U.S.A., 1990.

- **[KAJIHARA, 1972]** Kajihara, H.H.: "Power Transient Monitor", IEEE Transactions on Instrumentation and Measurement, Vol. IM-21, no. 1, pp. 74-77, 1972.

- **[KEY, 1979]** Key T.S.: "Diagnosing Power Quality - Related Computer Problems", IEEE Transactions on Industry Applications, Vol. IA-15, no. 4, pp. 381-393, 1979.

- **[KRAFT e TOY, 1981]** Kraft, G.d. and Toy, W.N.: "Microprogrammed Control and Reliable Design of Small Computers", Prentice Hall, N.J., 1981.

SERVIÇO DE BIBLIOTECA E INFORMAÇÃO - IFQSC

- **[LAMPORT e MELLIAR-SMITH, 1985]** Lamport, L. and Melliar-Smith, P.M.: "Synchronising Clocks in the Presence of Faults", Journal of ACM, Vol. 32, no. 1, pp. 52-78, 1985.

- **[LAPRIE, 1985]** Laprie, J.C.: "Dependable Computing and Fault Tolerance: Concepts and Terminology". 15th Symp. on Fault - Tolerant Computing, pp. 2-11, 1985.

- **[LOSQ, 1976]** LOSQ, J.: "A Highly Efficient Redundant Scheme: Self-Purging Redundancy", IEEE Trans. on Comp., Vol. C-25, no. 6, pp. 569-578, 1976.

- **[LU, 1982]** LU, D.J.: "Watchdog Processors and Structural Integrity Checking", IEEE Trans. on Computers, Vol. C-31, no. 7, 1982.

- **[MACHADO-TRAINA, 1987]** Machado Traina, A.: "Desenvolvimento de Software Básico para Gerenciamento de um Sistema Multiusuário Distribuído", Dissertação de Mestrado, Instituto de Ciências Matemáticas de São Carlos, USP, 217pp., 1987.

- **[MARSTON, 1980]** Marston, D.: "Memory System Reliability with ECC", Application Note AP-73, Intel Corporation, 31 pp., 1980.

- **[MARTZLOFF e HAHN, 1970]** Martzloff, F.D. e Hahn, G.J.: "Surge Voltages in Residential and Industrial Power Circuits", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-89, no. 6, pp. 1049-1056, 1970.

- **[MARTZLOFF, 1987]** Martzloff, F.D.: "Protecting Computer Systems Against Power Transients", IEEE Spectrum, Vol. 27, no. 4, pp. 37-40, 1987.

- **[MARTZLOFF e GRUZS, 1988]** Martzloff, F.D. and Gruzs, T.M.: "Power Quality Site Surveys: Facts, Fiction and Fallacies", IEE Transactions on Industry Applications, Vol. 24, no. 6, pp. 1005-1018, 1988.

- **[MATHUR e AVIZIENIS, 1970]** Mathur, F.P. and Avizienis, A.: "Reliability Analysis and Architecture of a Hybrid-Redundant Digital System: Generalized Triple Modular Redundancy with Self Repair", 1970 Spring Joint Comp. Conf., Vol 36, pp. 375-383, 1970.

- **[MATHUR e de SOUZA, 1975]** Mathur, F.P. and de Souza, P.T.: "Reliability Modelling and Analysis of General Modular Redundant Systems", IEEE Trans. of Reliability, Vol. R-24, no. 5, pp. 296-299, 1975.

- **[NATIONAL, 1987]** National Semiconductor Corporation: "Reliability Handbook, 281 pp., National Semiconductor Corporation, USA, 1987.

- **[OBAC-RODA, 1983]** Obac-Roda, V.: "Techniques for High Integrity Intelligent Instrumentation", Ph.D. Thesis University of London, 293 pp., 1983.

- **[OLIVEIRA JR., 1988]** Oliveira Jr., E.: "Informática - Evoluindo a Cultura". Anais do XXI Congresso Nacional de Informática, pp. 21-25, 1988.

- **[OSSFELDT e JONSSON, 1980]** Ossfeldt, B.E. and Jonsson, I.: "Recovery and Diagnostic Control of the AXE Switching System", IEEE Trans. on Comp., Vol C-29, no. 6, pp. 482-491, 1980.

- **[PAULA Jr et al., 1989]** Paula Jr., A.R.; de Souza, J.M.; Leite, J.; Martini, M.R.B.; Losques Filho, O.G.: "Fundamentos da Computação Tolerante a Falhas"; Mini Curso; III Simpósio de Computadores Tolerantes a Falhas, Cap. 1, pp. 2-24, 1989.

- **[PAWLOSKI et al., 1985]** Pawloski, M.B.; Moroyan, R. and Altnether, J.: "Inside CMOS Technology", Application Report AR-302, INTEL Corporation, 1985.

- **[PEARSON e PREECE, 1982]** Pearson, J.C. and Preece, C.: "Hardware and Software Aspects of Fault Coverage in Small Digital Controllers", Proc. of 5th European Conference on Electrotechnics - EUROCON'82, pp. 663-667, Copenhagen, 1982.

- **[PEATMAN, 1981]** Peatman, J.B.: "Digital Hardware Design", Mc Graw Hill, 1981.

- **[RECHSTEINER, 1985]** Rechsteiner, E.B.: "Clean, Stable Power for Computers", EMC Technology, Vol. 4, no. 3, pp. 67-85, 1985.

- **[SIEWIOREK e SWARZ, 1982]** Siewiorek, D.P. and Swarz, R.S.: "The Theory and Practice of Reliable System Design", Digital Press, USA, 772 pp., 1982.

- **[SILVA, 1991]** Silva, A.M.: "Monitor para Caracterização de Ruídos em Redes A.C.", Documento Interno no. 1, PD, v1, 7pp., 1991.

- **[SOSNOWSKI, 1986a]** Sosnowski, J.: "Evaluation of Transient Hazards in Microprocessor Controllers", Dig. 16th Int. Symp. on Fault-Tolerant Computing, Viena, pp. 364,369, 1986.

- **[SOSNOWSKI, 1986b]** Sosnowski, J.: "Risky Transients in M Microprocessor Controllers", Proc. of the 5th Int. Conf. on Reliability and Maintainability, Biarritz-France, pp. 641-646, 1986.

- **[TEXAS, 1989]** Texas Instruments Incorporated: "MOS Memory Data Book", Dallas, U.S.A., 1989.

- **[THOMPSON e NEWSTEAD, 1982]** Thompson, L.R. and Newstead, D.: "Safety Equipment for Mine Cages", IEE Colloquium on Microprocessor Applications Requiring High Integrity and Fault Tolerance, IEE Digest 1982/67, London, 1982.

- **[TRAINA Jr, 1986]** Traina Jr. C.: "Máquina e Modelo de Dados Dedicados para Aplicações de Engenharia", Tese de Doutoramento, Instituto de Física e Química de São Carlos - USP, Departamento de Física e Ciência dos Materiais, 379pp. 1986.

- **[TRINDADE Jr., 1987]** Trindade Jr., O.: "Uma contribuição ao Estudo do Efeito dos Distúrbios da Rede Elétrica sobre a Operação Confiável de Microcomputadores". Dissertação de Mestrado, Instituto de Física e Química de São Carlos/USP, 201p, 1987.

- **[U.S. NAVY, 1977]** U.S. Navy, Naval Facilities Engineering Command: "Reports on Power line disturbance monitoring and transient susceptibility testing for sensitive electronic equipment at Naval shore stations", Civil Engineering Lab., Port Hueneme (CA), 1968-1977.

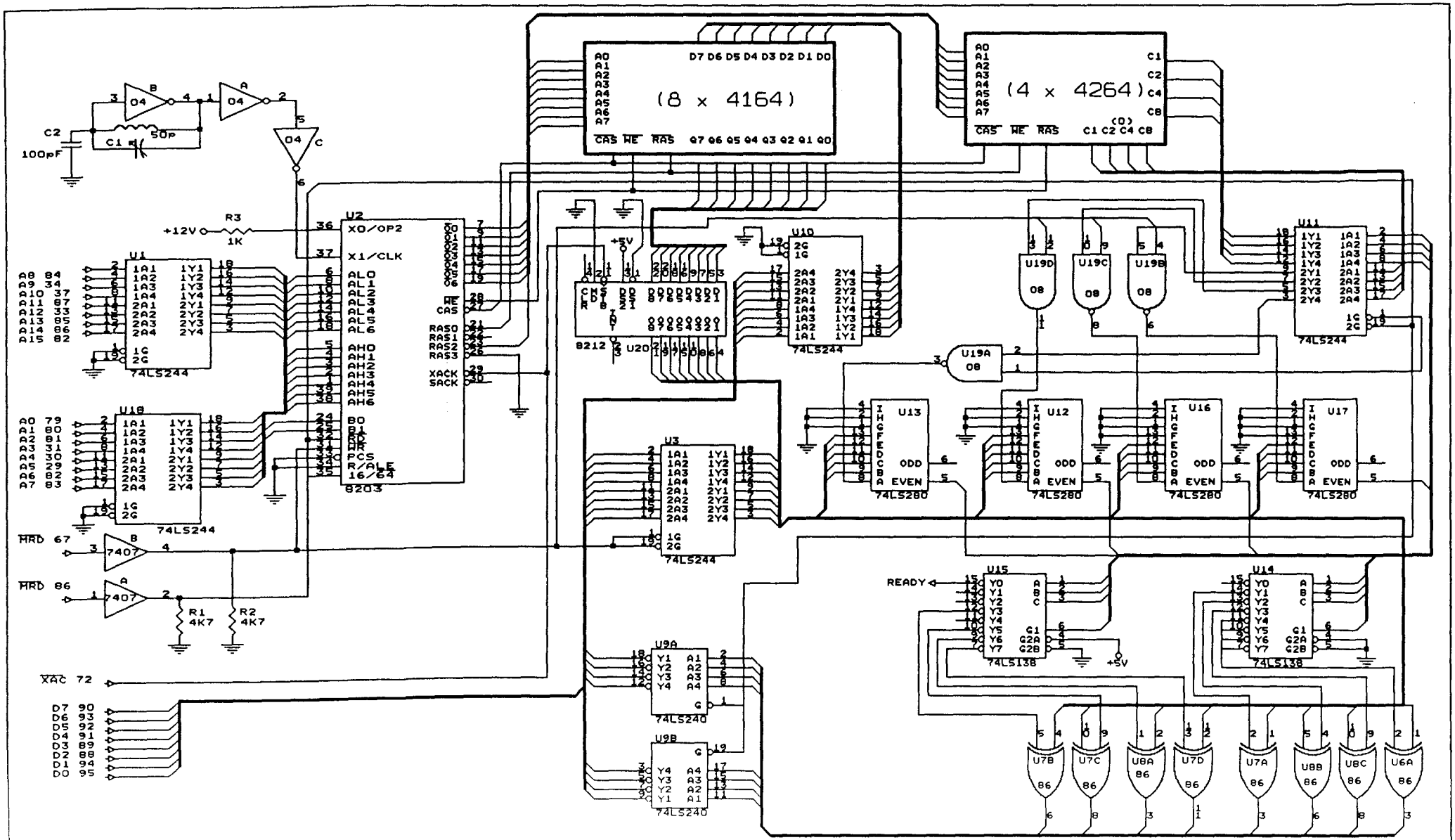
- **[von NEUMAN, 1956]** von Neuman, J.: "Probabilistic Logic and the Synthesis of Reliable Organisms from Unreliable Components", Automata Studies, Princeton University Press, p. 43-98, 1956.

- **[WENSLEY, 1983]** Wensley, J.H.: "An Operating System for a TMR Fault-Tolerant System", Dig. 13th Int. Symp. on Fault-Tolerant Computing, Milan, pp. 452-455, 1983.

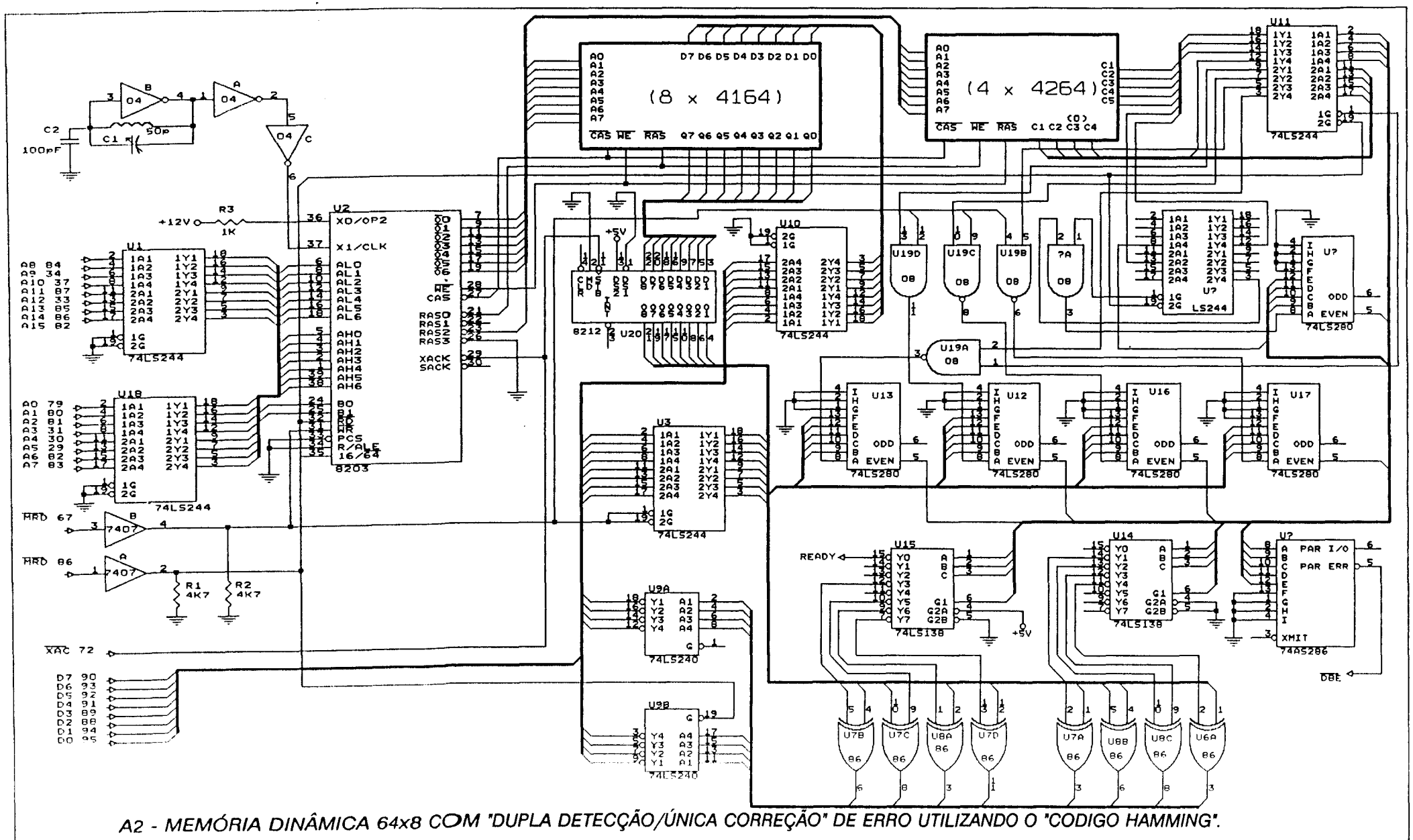
- **[WILLIAMSON, 1982]** Williamson, T.: "Designing Microcontroller Systems in Electrically Noisy Environments", Application Note AP-125, INTEL Corporation, 1982.

APÊNDICE A

DIAGRAMAS ESQUEMÁTIVOS DAS MEMÓRIAS COM DETECÇÃO/CORREÇÃO DE ERROS.



A1 - MEMÓRIA DINÂMICA 64Kx8 COM "ÚNICA DETECÇÃO/ÚNICA CORREÇÃO" DE ERRO UTILIZANDO O "CÓDIGO HAMMING".



A2 - MEMÓRIA DINÂMICA 64x8 COM "DUPLA DETECÇÃO/ÚNICA CORREÇÃO" DE ERRO UTILIZANDO O "CODIGO HAMMING".

APÊNDICE B

CONSIDERAÇÕES SOBRE A CONFIABILIDADE DO SISTEMA DUPLEX-HÍBRIDO.

Consideremos um sistema DUPLEX cujos módulos ativos tem uma taxa de falhas constante λ, sendo a confiabilidade desses módulos dada por:

$$R_m = e^{-\lambda t}$$

Consideremos tambem que a partir do sistema DUPLEX básico seja construido um sistema DUPLEX-HÍBRIDO (DH(S)) cujos módulos de reposição são considerados inativos (dormentes) e tem uma taxa de falha constante μ ($\mu < \lambda$). A confiabilidade deste módulo de reposição é dada por:

$$R_m = e^{-\mu t}$$

A confiabilidade do sistema DUPLEX ou DH(0) é dada pela expressão da confiabilidade de dois módulos em série:

$$R(DH(0)) = R_m^2$$

Sejam ainda feitas as seguintes considerações:

- O mecanismo comparador e de ativação dos módulos de reserva é considerado livre de falhas, ou seja a confiabilidade deste mecanismo é muito maior do que a dos módulos do sistema. Isto é razoavel se o número de componentes e a

complexidade destes for menor do que os componentes utilizados nos módulos.

- As falhas dos módulos são estatísticamente independentes.

- O sistema deve manter a operação correta durante um tempo de missão T.

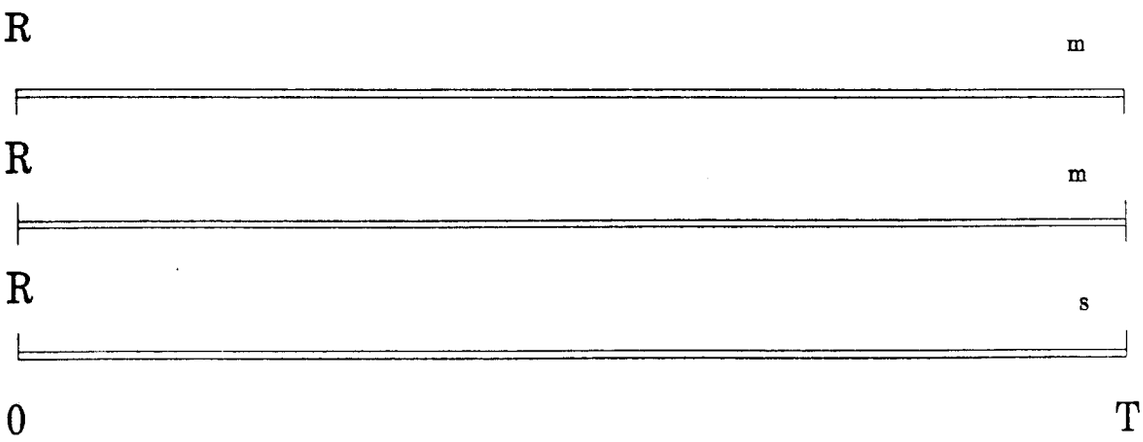
- A confiabilidade dos módulos do sistema DUPLEX e dos módulos de reposição no início da operação do sistema deve ser igual a 1.

- Apenas dois estados operacionais são permitidos para cada módulo: operação correta ou falha. Não foi considerado o caso de recuperação e integração ao sistema do módulo falho.

Para que o sistema DH(1) sobreviva o tempo de missão T deve ocorrer uma das seguintes situações:

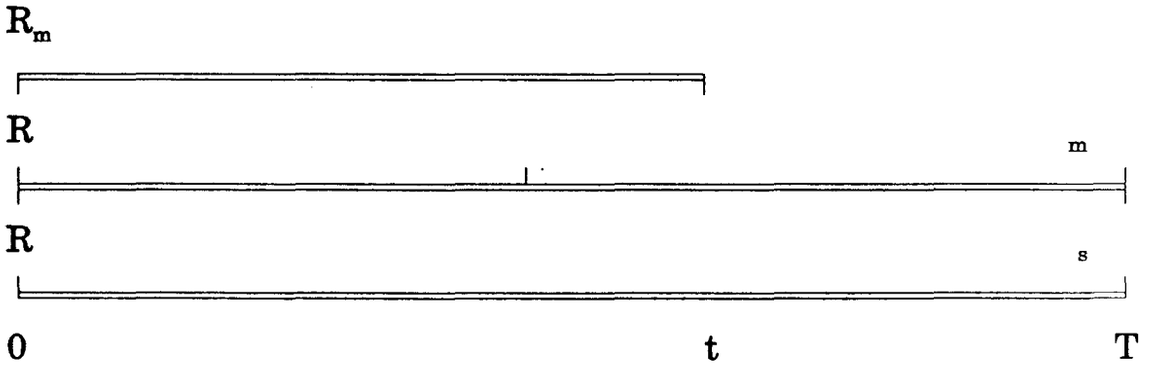
1) Os dois módulos ativos e o de reposição operam corretamente durante todo o tempo de missão. A probabilidade deste evento é:

$$R_m^2 . R_s$$



2) Num certo instante t (t < T) um dos módulos ativos falha, sendo

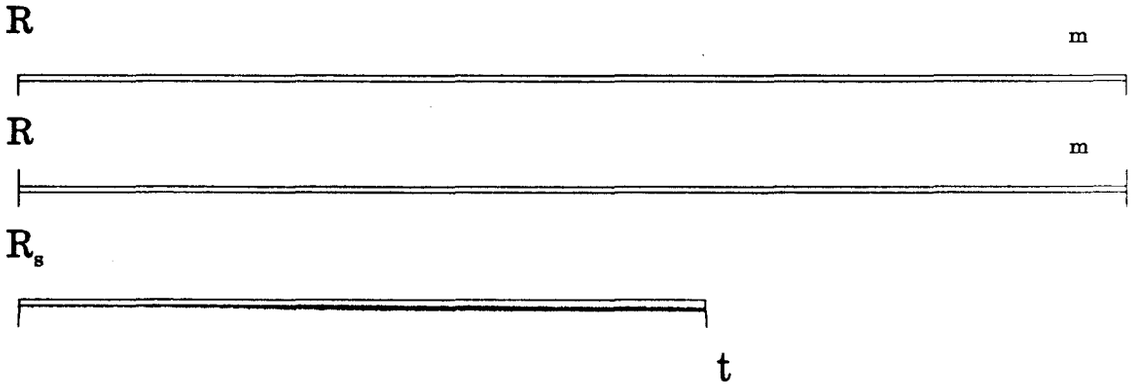
substituido por um módulo de reposição. O sistema opera na configuração DUPLEX até completar o tempo de missão.



A probabilidade deste evento é dada por:

$$2.\int_0^T e^{-\mu t}.e^{-\lambda t}.\lambda e^{-\lambda t}.R(DH(0))\,(T-t)\,dt$$

3) Num certo instante t, o módulo de reserva falha e o sistema continua em operação DUPLEX pelo resto do tempo de missão (T-t).



A probabilidade deste evento é dada por:

$$\int_0^T e^{-2\lambda t}.\mu.e^{-\lambda t}.R(DH(0))\,(T-t)\,dt$$

A confiabilidade total do sistema é determinada pela adição das probabilidades nos tres casos.

$$R(DH(1))(T) = R_m^2 . R_s(T) + (2\lambda+\mu)\int_0^T e^{-(2\lambda+\mu)t} . R(DH(0))(T-t)\,dt$$

$$R(DH(1))(T) = R_m^2 . R_s(T) + \frac{2\lambda+\mu}{\mu} R_m^2 . (1-R_s)$$

Para $\mu = \lambda$, a expressão acima fica:

$$R(DH(1))(T) = R_m^3(T) + 3(1-R_m) . R_m^2$$

que é identica à expressão da confiabilidade do sistema com redundância modular tripla ("TMR").

Para o sistema DUPLEX-HÍBRIDO com dois módulos de reposição, a expressão da confiabilidade pode ser determinada de maneira similar. Os tres casos que permitem a sobrevivência do sistema durante o tempo de missão T, são:

1) Os dois módulos ativos e os dois módulos de reposição sobrevivem o tempo de missão T. A probabilidade deste evento é:

$$R_m^2 . R_s^2$$

2) Um dos módulos ativos falha no instante t, sendo substituido por um dos módulos de reserva, deixando o sistema no modo DH(1) pelo resto do tempo (T-t). A probabilidade deste evento é:

$$2 . \int_0^T e^{-2\mu t} . e^{-\lambda t} . \lambda e^{-\lambda t} . R(DH(1))(T-t)\,dt$$

3) Em determinado instante t, um dos módulos de reposição falha,

deixando o sistema no modo DH(1) pelo resto do tempo (T-t). Este evento tem a probabilidade:

$$2.\int_0^T e^{-2\lambda t}.e^{-\mu t}.\mu.e^{-\mu t}.R(DH(1))(T-t)\,dt$$

Somando as probabilidades para os tres casos:

$$R(DH(2))(T)=R_m^2.R_s^2(T)+(2\lambda+2\mu)\int_0^T e^{-(2\lambda+2\mu)t}.R(DH(1))(T-t)\,dt$$

$$\begin{aligned}R(DH(2))(T)=&R_m^2.R_s^2(T)+\frac{2\lambda+2\mu}{\mu}R_m^2R_s.(1-R_s)\\&-\frac{(2\lambda+\mu)^2}{\mu^2}.R_m^2.R_s(1-R_s)\\&-\frac{(2\lambda+\mu)^2}{2\mu^2}.R_m^2.(1-R_s^2)\end{aligned}$$

APÊNDICE C

EQUAÇÕES DA CONFIABILIDADE DOS SISTEMAS DUPLEX/HÍBRIDOS

1. Sistema DUPLEX/HÍBRIDO com um módulo de reposição.

$$R(DH(1)) = R_M^2 . R_S . [1 - \frac{(2\lambda+\mu)}{u}] + \frac{(2\lambda+\mu)}{\mu} . R_m^2$$

2. Sistema DUPLEX/HÍBRIDO com dois módulos de reposição.

$$R(DH(2)) = R_M^2 . R_S . [1 - \frac{2\lambda+2\mu)}{\mu} + \frac{(2\lambda+\mu)\ (2\lambda+2\mu)}{\mu^2} - \frac{(2\lambda+\mu)\ (2\lambda+2}{2\mu^2}$$

$$+R_m^2 . R^s . [\frac{(2\lambda+2\mu)}{\mu} - \frac{(2\lambda+\mu)\ (2\lambda+2\mu)}{\mu^2}] + R_m^2 [\frac{2\lambda+\mu)\ (}{2\mu}$$

2. Sistema DUPLEX/HÍBRIDO com tres módulos de reposição.

$$R(DH(3)) = R_M^2 . R_s^3 . [1 - \frac{(2\lambda+3\mu)}{\mu} - \frac{(2\lambda+2\mu)\ (2\lambda+3\mu)}{2\mu^2} + \frac{(2\lambda+2\mu)\ (2\lambda+3}{\mu^2}$$

$$- \frac{(2\lambda+\mu)\ (2\lambda+2\mu)\ (2\lambda+3\mu)}{\mu^3} - \frac{(2\lambda+\mu)\ (2\lambda+2\mu)\ (2\lambda+3\mu)}{6\mu^3} + \frac{(2\lambda+}{}$$

$$+R_m^2 . R_s^2 . [\frac{2\lambda+3\mu)}{\mu} - \frac{(2\lambda+2\mu)\ (2\lambda+3\mu)}{\mu^2} + \frac{(2\lambda+\mu)\ (2\lambda+2\mu)\ (2\lambda+3\mu)}{\mu^3} -$$

$$+R_m^2 . R_s . [\frac{(2\lambda+2\mu)\ (2\lambda+3\mu)}{2\mu^2} - \frac{(2\lambda+\mu)\ (2\lambda+2\mu)\ (2}{2\mu^3}$$

$$+R_m^2 . [\frac{(2\lambda+\mu)\ (2\lambda+2\mu)\ (2\lambda+3\mu)}{6\mu^3}]$$