

**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA**

**TESE DE MESTRADO
UTILIZAÇÃO DE EQUIPAMENTOS
AUTOMÁTICOS DE TESTE EM
CIRCUITOS INTEGRADOS DIGITAIS.**

AUTOR : Rogério Lara Leite

ORIENTADOR : Dr. José Antônio Siqueira Dias

Campinas, 1994.

AGRADECIMENTOS

TESE

preparada no

**Departamento de Eletrônica e Microeletrônica - DEMIC
da Faculdade de Engenharia Elétrica da UNICAMP**

visando a obtenção

do Título de Mestre em Engenharia Elétrica - FEE/UNICAMP

Especialidade : Eletrônica

por

Rogério Lara Leite *nº/536*

Engenheiro Eletricista

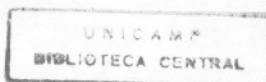
Utilização de Equipamentos Automáticos de Teste em Circuitos Integrados Digitais

Banca Examinadora:

Prof. Dr. José Antonio Siqueira *X* Dias (Presidente) - DEMIC/FEE/UNICAMP.

Prof. Dr. Laércio Caldeira - Escola Federal De Eng. de ITAJUBÁ - EFEI.

Prof. Dr. Alberto Martins Jorge - DEMIC/FEE/Unicamp.



Este exemplar corresponde à redação final da tese defendida por *Rogério Lara Leite*

e aprovada pela Comissão

Julgadora em *27* / *09* / *94*

José Antonio Siqueira
Orientador

9418972

AGRADECIMENTOS

Este trabalho contou com a colaboração de várias pessoas sem as quais não teria sido possível realizá-lo. Em especial ao orientador desta tese, professor Dr. José Antônio Siqueira Dias. Também foi muito importante o apoio da TELEBRÁS - Telecomunicações Brasileiras S. A - que forneceu a infra-estrutura necessária ao desenvolvimento deste trabalho.

Para minha esposa
Maria Isabel
e para minhas duas filhas
Helena
e
Júlia.

Para meus pais pelo incentivo e apoio que recebi.

RESUMO

Este trabalho comenta alguns aspectos importantes do teste automático de um circuito integrado digital. Apresenta os principais tipos de testes elétricos realizados por um equipamento automático de teste, comentando as diferenças dos testes dependendo da tecnologia do componente, nas diversas fases da vida de um circuito integrado digital. São descritos, de forma sucinta, os principais mecanismos de falhas em CI's digitais e são apresentadas as principais medições elétricas necessárias para avaliar o desempenho de um circuito integrado. Descrevemos também o equipamento automático de teste (ATE) e sua linguagem de programação, comentando como esta máquina é importante para testar circuitos integrados digitais. O trabalho termina com dois programas de teste reais, escritos em Pascal, comentando os resultados das medições de cada programa.

ABSTRACT

This work comments some important aspects of the digital integrated circuit automatic test. It presents the most common electrical tests done by an Automatic Test Equipment - ATE. The test differences depending on chip technology in the various steps of the integrated circuit life are commented. The main IC's digital faults and failures mechanisms are commented in a introductory way. The principal electrical measurements necessary to estimate the performance of an digital IC is presented. The architecture and the language of the ATE is presented , discussing how this machine is important to test digital integrated circuits. The work ends with two real test programs, written in Pascal commenting the results of the measurements of each test program.

Sumário

1	INTRODUÇÃO AO TESTE DE CIRCUITOS INTEGRADOS DIGITAIS	3
1.1	OBJETIVO DA ENGENHARIA DE TESTE	4
1.2	CUSTO / EFICIÊNCIA	4
1.3	CONSIDERAÇÕES	4
1.4	APLICAÇÃO DO TESTE	5
1.4.1	Depuração do protótipo	6
1.4.2	Avaliação funcional	6
1.4.3	Caracterização	6
1.4.4	Produção	7
1.4.5	Ensaio de confiabilidade	7
1.4.6	Inspeção de recebimento	8
1.4.7	Análise de falhas	8
1.5	FATORES QUE INFLUENCIAM NO TESTE	8
1.5.1	Propósito do teste	9
1.5.2	Tipos de testes	9
1.5.3	Tecnologia do dispositivo	10
1.5.4	Estágio de fabricação do dispositivo	10
1.5.5	Tipos de dispositivos	11
1.6	EQUIPAMENTOS AUTOMÁTICOS DE TESTE - ATE	12
1.7	TESTE DE CIRCUITOS DIGITAIS	13
1.8	REFERÊNCIAS BIBLIOGRÁFICAS	14

2	FALHAS E MECANISMOS DE FALHAS	16
2.1	INTRODUÇÃO	17
2.2	FALHAS TIPO "SINGLE STUCK-AT"	17
2.3	MÚLTIPLAS FALHAS "STUCK-AT"	18
2.3.1	Múltiplas falhas interativas	19
2.3.2	Falhas "Bridging"	19
2.4	FALHAS INTERMITENTES	20
2.4.1	Vibração	20
2.4.2	Estresse térmico	21
2.5	DESCARGA ELETROSTÁTICA - ESD	22
2.6	SOBRE ESTRESSE ELÉTRICO - EOS	23
2.7	CORROSÃO	24
2.8	FALHAS EM MEMÓRIAS	24
2.8.1	Principais falhas nas memórias	26
2.8.2	Descrição das falhas	26
2.8.3	Padrões de teste	27
2.9	REFERÊNCIAS BIBLIOGRÁFICAS	28
3	TESTES REALIZADOS POR UM ATE	29
3.1	TIPOS DE TESTES	30
3.2	FUNCIONAL	30
3.3	PARAMÉTRICO DC	31
3.3.1	Testes paramétricos realizados	32
3.3.2	Teste de continuidade	32
3.3.3	Medição de IIL	34
3.3.4	Medição de IIH	34
3.3.5	Medição de IOZ	35
3.3.6	Medição de VIH E VIL	37

3.3.7	Medição de VOH E VOL	40
3.3.8	Medição de IDD ou ICC	41
3.3.9	Medição de IOS	43
3.3.10	Medição de IOL	44
3.3.11	Medição de IOH	46
3.4	PARAMÉTRICO AC	46
3.4.1	Evento	47
3.4.2	Atraso de propagação	48
3.4.3	tPHL	48
3.4.4	tPLH	49
3.4.5	tPD	49
3.4.6	tPHZ	49
3.4.7	tPLZ	50
3.4.8	tPZH	50
3.4.9	tPZL	50
3.4.10	Tempos de transição	52
3.4.11	tTLH	53
3.4.12	tTHL	53
3.4.13	Tempo de "setup" e de "hold"	53
3.4.14	Tempo de "setup"	54
3.4.15	Procedimento para medição do tempo de "setup"	54
3.4.16	Tempo de "hold"	55
3.4.17	Procedimento para medição do tempo de "hold"	56
3.4.18	tW	57
3.4.19	fmax	58
3.5	REFERÊNCIAS BIBLIOGRÁFICAS	58
4	ATE E LINGUAGENS DE PROGRAMAÇÃO	60

4.1	DEFINIÇÃO	61
4.2	HISTÓRICO	61
4.3	APLICAÇÃO DOS ATE'S	61
4.4	CUSTO DO TESTADOR	62
4.5	ARQUITETURA BÁSICA DE UM ATE	62
4.6	J325 - SISTEMA AUTOMÁTICO DE TESTES DE CIRCUITOS INTEGRADOS DIGITAIS	64
4.6.1	Introdução	64
4.6.2	Terminal de vídeo	64
4.6.3	Unidade de fitas	65
4.6.4	Impressora	65
4.6.5	Computador	66
4.6.6	Memória	66
4.6.7	Unidade de medidas analógicas (AMU)	66
4.6.8	Conversores analógicos digitais e digitais analógicos	68
4.6.9	Fontes de tensão para o dispositivo	68
4.6.10	Sistema de medidas diferenciais	68
4.6.11	Estações de teste	68
4.6.12	Características da estação HV	69
4.6.13	A estação HV do J325	69
4.6.14	"Drivers" e detectores	70
4.6.15	"Load board"	70
4.6.16	DIB - "Device interface board"	72
4.6.17	Circuitos de tempo	72
4.6.18	Modos de operação	74
4.6.19	Cargas externas	75
4.7	Linguagem de Programação dos ATE	75
4.7.1	A linguagem de programação Pascal-T	76

4.7.2	Estrutura do Programa	76
4.7.3	Definições de canais	78
4.7.4	Teste de continuidade	79
4.7.5	Níveis de tensão das fontes	80
4.7.6	Níveis de tensão dos "drivers"	80
4.7.7	Medidas de tensões e de correntes	81
4.7.8	Programando o teste funcional	82
4.7.9	Temporização	83
4.8	MACROS	84
4.9	REFERÊNCIAS BIBLIOGRÁFICAS	85
5	PROGRAMA DE TESTE - APLICAÇÕES	86
5.1	INTRODUÇÃO	87
5.2	PROGRAMA DE TESTE	87
5.3	RESULTADOS	98
5.3.1	"Datalog" do 74HCT32	98
5.3.2	"Datalog" do HM62256	100
5.4	CONCLUSÃO	101
5.5	REFERÊNCIAS BIBLIOGRÁFICAS	103

INTRODUÇÃO

A evolução atual da tecnologia de semicondutores possibilita a produção de dispositivos com mais de 10.000.000 de transistores difundidos em uma única pastilha de silício, com tecnologia de $0.4\mu\text{m}$. O aumento da complexidade dos circuitos integrados eleva o grau de dificuldade para os projetistas de circuitos integrados, responsáveis pelo desenvolvimento do *chip*, e para os engenheiros de teste, responsáveis por verificar que o circuito fabricado satisfaça plenamente as especificações elétricas e funcionais.

Do ponto de vista de teste, um circuito integrado encapsulado deve ser considerado como uma caixa preta, com um número finito de entradas e saídas. Diferentemente de circuitos eletrônicos montados em placas de circuitos impressos, não se tem acesso a pontos intermediários do circuito integrado, a menos que este ponto seja levado a um terminal (pino) do CI. Daí a preocupação, hoje em dia, dos projetistas de CI com a testabilidade do componente em desenvolvimento.

O teste requer que certas condições de entrada sejam aplicadas ao dispositivo e as saídas sejam comparadas com respostas esperadas, conhecidas, que foram geradas na fase de projeto por um simulador.

Circuitos puramente digitais podem ser funcionalmente verificados pela aplicação de padrões ou seqüências de padrões de “0”s e “1”s (níveis baixos e altos de tensão), com duração de tempo pré-definidas, nas entradas do CI, para produzir o padrão ou a seqüência de padrões desejada na saída do componente.

Circuitos puramente analógicos, entretanto, necessitam do uso de geradores de sinais mais especializados para estimular o componente e equipamentos mais sofisticados para observar e medir a resposta do componente. Isto requer um equipamento de teste mais complexo, contendo vários equipamentos de estímulo e de medida de resposta, possibilitando ao engenheiro de teste, total controle das funções de cada instrumento de medida através de um programa.

Circuitos digitais de grande complexidade, com escala de integração VLSI e circuitos mistos, onde funções analógicas e digitais estão reunidas em um único circuito integrado, o que está se tornando cada vez mais freqüente nos dias de hoje, requerem um equipamento deste tipo para serem testados.

Constata-se, hoje em dia, que as ferramentas de projeto de CI e de desenvolvimento de programas para testes de circuitos integrados estão acompanhando e viabilizando a complexa evolução dos Circuitos Integrados. Tais ferramentas necessitam de pessoal especializado para operação, programação e manutenção.

Este trabalho descreve o Teste de Circuitos Integrados Digitais, em seus aspectos gerais, utilizando Equipamentos Automáticos de Testes - ATE . No Capítulo I temos uma visão geral do que venha a ser teste de circuitos digitais. O Capítulo II discute as Falhas e os Mecanismos em Circuitos Integrados Digitais. A descrição do ATE e os Tipos de Teste que realiza são apresentados no Capítulo III. A Arquitetura e a Linguagem de Programação dos ATE's está no Capítulo IV e no Capítulo V estão as aplicações práticas.

Capítulo 1

INTRODUÇÃO AO TESTE DE CIRCUITOS INTEGRADOS DIGITAIS

1.1 OBJETIVO DA ENGENHARIA DE TESTE

O objetivo de se testar um componente é assegurar que o dispositivo desempenhe corretamente todas as funções para que foi projetado, considerando-se o pior caso das especificações para o ambiente de teste especificado. Equipamentos Automáticos de Teste são controlados por computador, o que requer do engenheiro de teste uma experiência em ciência da computação tanto em programação quanto operação de sistemas. O engenheiro de teste deve saber também, como interfacear corretamente o dispositivo a ser testado ao equipamento de teste.

1.2 CUSTO / EFICIÊNCIA

O teste representa cerca de 40% do custo do dispositivo, podendo, em alguns casos, atingir até 60% do custo de desenvolvimento do componente. O custo de um teste deve ser mantido o mais baixo possível. Este custo varia dependendo do tipo de teste a ser realizado e do tipo de componente a ser testado. A figura 1.0 mostra a variação relativa do custo para corrigir uma falha em diversas etapas do desenvolvimento de um sistema.

1.3 CONSIDERAÇÕES

O engenheiro de teste, além de se preocupar com o desenvolvimento de programa de teste (para equipamentos automáticos de teste - ATE), deve também se preocupar com a “fiação” que conecta o componente em teste no equipamento de teste, que se não for bem feita, pode acarretar distorções nas medidas realizadas.

É importante ajustar a diferença entre o ambiente de teste e o ambiente final de campo no qual o componente será utilizado.

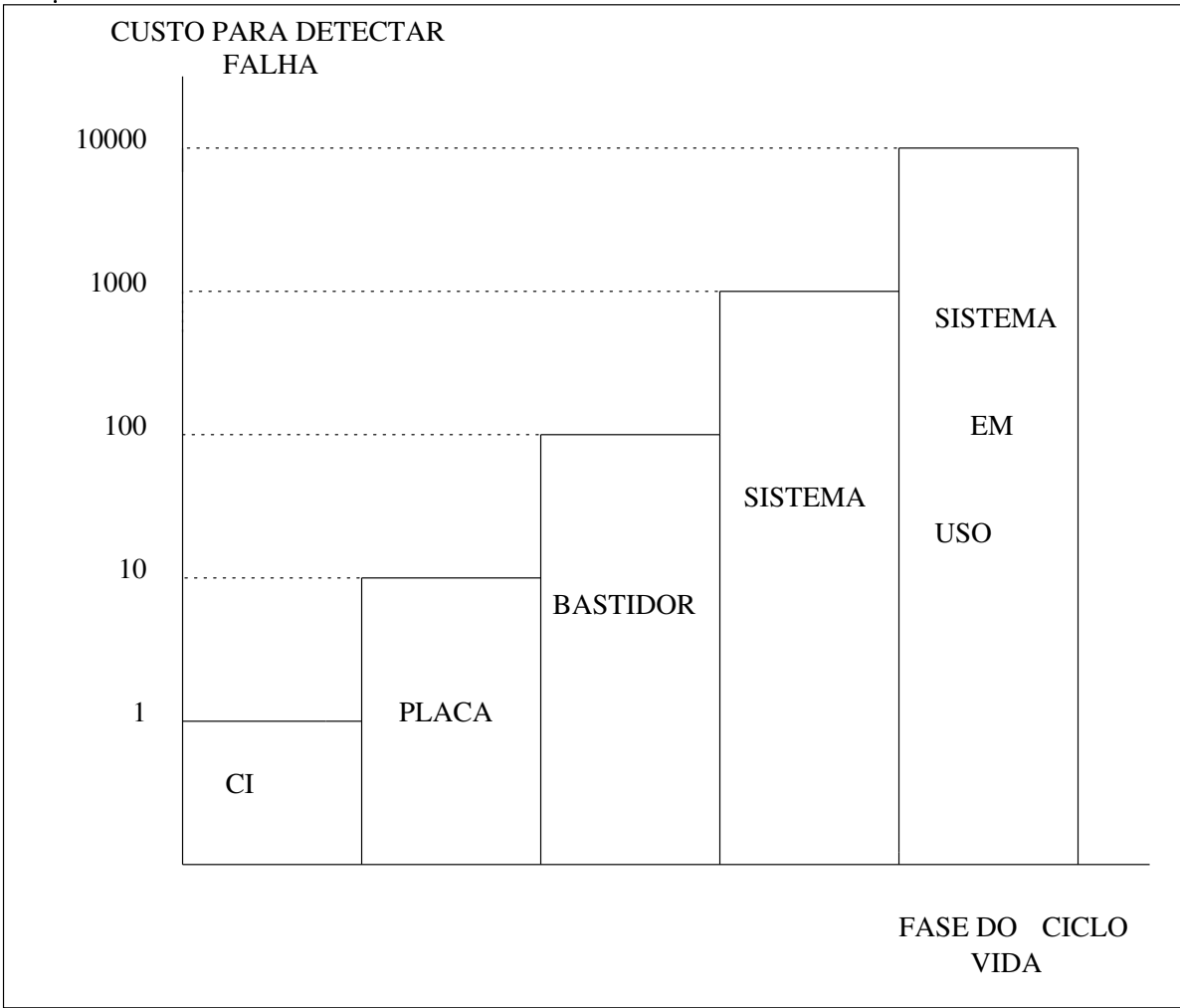


Figura 1.0 : Custo relativo para reparar uma falha em diversas etapas do desenvolvimento de um sistema.

1.4 APLICAÇÃO DO TESTE

O teste de um componente pode ser necessário para vários propósitos diferentes a saber:

- Depuração do Protótipo
- Avaliação Funcional

- Caracterização
- Produção
- Ensaio de confiabilidade
- Inspeção de Recebimento
- Análise de Falhas

1.4.1 Depuração do protótipo

É um teste realizado durante a fase de desenvolvimento do *Chip*. Após o projeto e após exaustivas simulações do funcionamento do circuito, é construído um protótipo do integrado. Algumas das amostras construídas vão para o teste, onde será verificado se o protótipo satisfaz o projeto lógico.

1.4.2 Avaliação funcional

Este teste também é realizado durante a fase de desenvolvimento do *Chip*. Tem a finalidade de verificar se o projeto lógico cumpre as especificações.

1.4.3 Caracterização

A função da caracterização é determinar se o projeto está correto e levantar as características elétricas do dispositivo. Isto significa realmente medir as características AC e DC ao invés de verificar se o dispositivo atende as especificações-padrões da tecnologia utilizada.

A caracterização é o tipo de teste utilizado para levantar as especificações finais do componente (Manual do Componente). Neste caso, o tempo de teste não é crítico, a meta é obter medidas precisas e exatas.

Requer-se, além de um bom testador, um equipamento térmico especial para aquecer e resfriar o componente na cabeça de teste, para se obter as medidas dos parâmetros em temperaturas extremas.

Durante esta fase é necessário também utilizar ferramentas (*softwares*) estatísticas

para verificar e estudar as variações e desvios das medidas realizadas. É nesta fase também que são gerados os limites de passagem e falha (*guardbands*) que serão utilizados no teste de produção. É a etapa mais cara do teste.

1.4.4 Produção

O objetivo do teste de produção é assegurar que o dispositivo encapsulado esteja dentro das especificações, geradas na fase de caracterização, e que funcione corretamente. Este tipo de teste deve ser rápido para uma melhor relação custo/eficiência, isto é, deve ter uma máxima cobertura em um mínimo tempo. Este é o tipo de teste utilizado nas fábricas de circuitos integrados para teste de seus produtos.

Um outro tipo de teste de produção é o chamado *Wafer Sort*, que tem a finalidade de selecionar os componentes antes do encapsulamento.

1.4.5 Ensaios de confiabilidade

O objetivo destes ensaios é verificar a conformidade dos componentes após submetidos a condições estressantes para avaliar se são ou não confiáveis. Os ensaios que necessitam de teste elétrico no decorrer de sua execução são: variação rápida de temperatura, seqüência climática, calor úmido prolongado, calor úmido polarizado, vida operacional, armazenagem a alta temperatura e sensibilidade a descarga eletrostática.

O ensaio de vida operacional consiste em assegurar que o dispositivo funcione corretamente durante um determinado número mínimo de anos. O processo de envelhecimento do componente é baseado em estressar o componente em uma temperatura elevada. Devem ser feitas correlações entre a vida útil do dispositivo em temperatura ambiente e em uma temperatura mais elevada.

A idéia deste tipo de ensaio é colocar o dispositivo em uma estufa com temperatura especificada (maior que a temperatura ambiente), com a polarização adequada, por um período de tempo, e então testá-lo novamente. Com base na temperatura e no tempo em que o componente ficou na estufa, pode-se determinar a expectativa de vida do componente. Este procedimento requer

análises estatísticas e probabilísticas com profundidade.

1.4.6 Inspeção de recebimento

Este tipo de teste consiste em verificar se o componente recebido está funcionando de acordo com as especificações do fabricante. É um teste rápido, por amostragem de lote, mas assegura que o lote de peças recebidas está em perfeitas condições de utilização, garantindo maior confiabilidade ao produto final.

1.4.7 Análise de falhas

Como o próprio nome diz, significa analisar, pesquisar a causa de uma determinada falha em um circuito integrado. Para realizar este trabalho, o analista de falhas necessita, além das técnicas e equipamentos de microscopia, de uma capela e de produtos químicos para a abertura do componente e de um teste funcional e paramétrico do componente, a fim de verificar o tipo de falha para orientar o trabalho de busca da causa da falha.

Algumas vezes, o analista de falhas e o engenheiro de teste vão juntos ao equipamento automático de teste para depurar a falha do CI. Neste caso, às vezes, é necessário alterar o programa de teste para caracterizar bem a falha e quantificar precisamente o valor de um parâmetro medido, que porventura tenha ultrapassado o limite da escala programada para o medidor. Geralmente o teste do *chip* é realizado antes que o analista abra o componente para iniciar o trabalho de pesquisa da falha.

1.5 FATORES QUE INFLUENCIAM NO TESTE

Técnicas diferenciadas de teste são aplicadas dependendo da finalidade do teste. Tais técnicas dependem :

- Do propósito do teste
- Do tipo de teste a ser aplicado
- Da tecnologia utilizada no *chip*

- Do estágio de fabricação do dispositivo
- Do tipo de dispositivo

1.5.1 Propósito do teste

A complexidade de um programa de teste, o pessoal envolvido, o custo e o tipo de equipamento necessário para o teste variam dependendo se o teste é para depuração do protótipo, para avaliação funcional, para caracterização do dispositivo, para teste de vida, para produção, para inspeção de recebimento ou para análise de falhas.

O teste funcional é suficiente para a depuração do protótipo e para a avaliação funcional.

Um teste paramétrico realizado durante a fase de caracterização de um *chip* é bem diferente de um teste paramétrico para a produção ou para a inspeção de recebimento, embora a mesma especificação seja verificada em cada um dos casos.

Na caracterização, os parâmetros AC e DC do componente são realmente medidos e seus limites são levantados, inclusive em temperaturas diferentes.

Para o ensaio de vida é necessário um teste paramétrico e um teste funcional, além de estufas para envelhecimento do componente. Na produção e na inspeção de recebimento estes parâmetros são “checados” contra os limites do manual.

1.5.2 Tipos de testes

Há basicamente três tipos de testes elétricos realizados em um componente:

- Funcional
- Paramétrico DC
- Paramétrico AC

O teste funcional verifica se o dispositivo está operando como o esperado. Neste teste a tabela da verdade do componente é verificada. Este tipo de teste será discutido com mais

detalhes no Capítulo III.

O teste paramétrico DC verifica se o dispositivo opera bem em seu ambiente especificado. Este teste mede a corrente de consumo do dispositivo e sua habilidade de operar corretamente com níveis de tensão apropriados. As correntes de fuga também são medidas neste teste.

O teste paramétrico AC está relacionado com tempos, com atrasos de propagação, tempos de transição, tempos de acessos, etc.

Tanto as medidas paramétricas DC quanto as medidas paramétricas AC estão comentadas no Capítulo III.

1.5.3 Tecnologia do dispositivo

Os parâmetros DC e AC de um componente eletrônico variam bastante, dependendo da tecnologia utilizada. Os dispositivos processados em cada tecnologia possuem características elétricas diferentes. Assim, diferentes especificações devem ser verificadas ou testadas de maneiras diferenciadas. Por exemplo, as exigências de níveis de tensão de entrada e as correntes de fuga para a tecnologia ECL possuem significados diferentes das mesmas exigências de um TTL ou CMOS, com alta impedância de entrada. As tensões de saídas de um CMOS são medidas sem carga, enquanto que as mesmas medidas para um TTL são realizadas com uma corrente de carga.

O resultado de um teste paramétrico AC para dispositivos ECL são dependentes do tempo de subida do sinal de entrada, o que tem pouca influência nos resultados destas medidas para os dispositivos CMOS.

1.5.4 Estágio de fabricação do dispositivo

O modo de interfacear o dispositivo sob teste depende do estágio de fabricação do dispositivo. Isto tem efeito no ambiente elétrico em que o dispositivo que está sendo testado está envolvido.

Quando se realiza um teste, é aconselhável ter o *pin electronics* do testador mais próximo possível do componente sob teste, a fim de se evitar interferências devido as indutâncias e capacitâncias dos cabos envolvidos. O *pin electronics* do testador é a eletrônica ou os circuitos

eletrônicos localizados na cabeça de teste do testador, que estimulam o dispositivo e detectam sua resposta. São os canais de *driver* e de comparadores.

Durante o teste de *Wafer*, isto é, do *chip* antes de ser encapsulado, a *interface* do testador com o *wafer* é difícil de ser realizada, pois envolve fiação coaxial externa, considerando que a cabeça de teste do ATE muitas vezes é grande e difícil de ser movida.

Em alguns casos é possível interfacear o testador com um equipamento micro-manipulador. Este equipamento, comandado pelo ATE, coloca as micro agulhas nos *pads* do circuito integrado para estimulá-lo e medir sua resposta.

O *wafer sort* é um teste complicado de ser realizado, além de ser caro. É necessário um estudo bem detalhado de sua viabilidade econômica, pois, por outro lado, o encapsulamento do CI também é caro. Então, encapsular somente as pastilhas selecionadas pode ser vantajoso economicamente, dependendo do caso.

No teste de *Wafer*, o problema de *interface* do dispositivo com o testador é crítico. O engenheiro de teste deve tomar bastante cuidado com a fiação (os cabos coaxiais) e fazer os ajustes de *software* necessários, tanto de *timing* quanto de níveis para corrigir eventuais distorções ocasionadas pelos cabos.

O problema acima de interfaceamento é menor no teste com o *chip* encapsulado. Neste caso, o soquete que suportará o componente a ser testado é montado em um *Device Interface Board - DIB*, uma placa de *interface* que vai conectada na cabeça de teste bem próxima do *pin electronics*. Mesmo assim, as ligações dos canais do testador com os pinos do dispositivo devem ser realizadas com cabos coaxiais e deve-se tomar os cuidados necessários para se trabalhar com alta frequência.

1.5.5 Tipos de dispositivos

As diferenças do teste também dependem do tipo de dispositivo que será testado.

As memórias e os dispositivos lógicos utilizam aparatos diferentes para gerar seus padrões funcionais de teste.

Os testadores de *chips* lógicos armazenam a tabela da verdade do dispositivo na memória de padrão e aplicam esses padrões seqüencialmente ao dispositivo.

Os testadores de memória utilizam algoritmos geradores de padrões (ALPG). Vários tipos de instruções criam um padrão de teste completo para o teste da memória.

Os *gate arrays* são testados no testador de lógica, mas podem ser testados também em testadores de memória, devido ao tamanho pequeno da tabela da verdade que define a sua lógica.

Há possibilidade de se testar memórias em testadores de lógicas; o problema é que o padrão de teste fica grande e o programa de teste fica um pouco lento, inviável para o teste de produção.

1.6 EQUIPAMENTOS AUTOMÁTICOS DE TESTE - ATE

Os Equipamentos Automáticos de Teste, os ATE (*Automated Test Equipment*), são sistemas programáveis que controlam e monitoram os circuitos integrados em teste para determinar se são ou não operacionais, estimulando as entradas e analisando as saídas do circuito. São equipamentos imprescindíveis e importantíssimos para a verificação do correto funcionamento de circuitos integrados, que cada vez mais estão se tornando mais complexos, contendo mais transistores chaveando em velocidades cada vez maiores e, portanto, mais difícil de serem testados.

Os equipamentos programáveis que possibilitam a automação dos testes dos circuitos integrados estão em constante evolução, acompanhando e auxiliando esta fantástica e rápida evolução dos circuitos integrados.

Os dispositivos desenvolvidos pela indústria de semicondutores de vanguarda são sempre mais evoluídos que os *chips* que vão nos equipamentos automáticos de teste. Embora sendo construídos de componentes de uma geração tecnológica anterior, os ATE possuem a missão de testar os componentes de última geração.

Os ATE contêm um conjunto de equipamentos para estímulo, para medida da resposta do componente, fontes de tensão, fontes de corrente, processadores digitais de sinais, etc. Todos esses equipamentos são totalmente controlados por um computador.

Alguns ATE possuem um computador somente para comandar os equipamentos relacionados com a cabeça de teste, e um outro computador para a *interface* homem/máquina e para

o gerenciamento da rede local.

O engenheiro de teste possui total controle dos equipamentos através de um programa escrito em linguagem de alto nível. Este assunto será comentado com mais detalhes no capítulo IV.

1.7 TESTE DE CIRCUITOS DIGITAIS

O objetivo primário do engenheiro de teste é assegurar que o circuito digital em perfeito funcionamento opere de acordo com sua tabela da verdade. A tabela da verdade (padrão de teste) é armazenada na memória de padrão do testador. Cada linha do padrão, chamada de Vetor de Teste, contém os *bits* de estímulos e a resposta esperada.

Para estimular o dispositivo, os vetores de teste armazenados na memória de padrão do ATE passam por um circuito chamado formatador, que combina os dados digitais com as bordas de tempo, programadas para gerar a forma de onda digital desejada. Esta forma de onda passa depois por um circuito de *driver* que coloca os níveis de tensão, programados para o nível lógico 1 e para o nível lógico 0.

A forma de onda digital completa é aplicada ao dispositivo sob teste em uma dada frequência também programada por *Software*, através de uma *interface* chamada de *Loadboard*, ou em alguns testadores *Device Interface Board - DIB*, que conecta os canais do testador com o soquete que contém o componente a ser testado. O dispositivo responde com uma saída, que passa através do *Loadboard* e vai para o comparador localizado na cabeça de teste. O dado esperado é comparado com as tensões de referência em um dado instante de tempo. Tanto as tensões de referências quanto o instante de tempo em que a comparação será realizada podem ser alterados no programa de teste.

É possível conectar nos pinos de saída do componente, cargas ou correntes de cargas programáveis, necessárias para as medidas paramétricas ou para a execução de um teste funcional mais rigoroso. Naturalmente, existe um *driver* e um comparador para cada pino do CI, exceto, é claro, para os pinos de alimentação, que são conectados a uma fonte de tensão programável para o pino de VDD e ao terra para o pino de GND.

Os vetores de teste armazenados na memória de padrão do testador são utilizados também para colocar o dispositivo em uma determinada condição ou estado que possibilite a realização de medidas paramétricas AC e DC.

O computador toma a decisão se o componente passa ou não, comparando o que foi medido com o que foi programado. O resultado do teste, chamado *datalog*, pode ser armazenado em disco ou fita, apresentado na tela do testador ou impresso em papel.

Às vezes não é necessário visualizar ou gravar os valores das medidas realizadas e sim a classificação do componente. Isto é feito pelo *Bin*. Alguns ATE apresentam o número do *Bin* na tela; outros, mais antigos, acendem um *LED* correspondente ao resultado do teste. O engenheiro de teste programa o *Bin*. Por exemplo, se o componente falhou no teste paramétrico DC o *Bin 2* programado aparece na tela do testador, ou o *LED* do 2 do *Binbox* acende. Se o componente estiver bom, o *Bin 1* aparecerá na tela ou o *LED 1* acenderá.

O capítulo III comenta os testes funcional, paramétrico DC e paramétrico AC com mais detalhes.

1.8 REFERÊNCIAS BIBLIOGRÁFICAS

- STEVENS, Anthony K. *Introduction to component testing*. Addison-Wesley, 1986.
- BENNETTS, R. G. *Design of testable logic circuits*. London, Addison-Wesley, 1984.
- BRAGA, Flávio. *Teste de circuitos integrados*. Documentação interna ao CPqD TELEBRÁS, DME, Seção de Aplicação, 1992.
- FEUGATE JUNIOR, Robert J; MCLNTYRE, Steven M. *Introduction to VLSI testing*. Englewood Cliffs, Prentice-Hall, 1988.
- NOVO, Marco Antônio Duarte; RIBEIRO, Álvaro José Lopes; PEDRINI, Hélio; *Uma metodologia para caracterização de CI's no CPqD*. IN: Anais do Congresso da Sociedade Brasileira de Microeletrônica 8., Campinas, 1993.
- CORTNER, J. Max. *Digital test engineering*. "Wiley-Interscience", 1987.

- FANTINI,F; GIOVANETTI, A; GAVIRAGHI; BRAMBILLA, P. *Asic qualification and testing: some meaningful examples*. Teletra S. P. A. 20059 Vimercate , Italy.
- FRANK, Edward H;SPROUL, Robert F. *Testing and debugging custom integrated circuits*; Computing Surveys, Vol 13, N. 4, December 1981.

Capítulo 2

FALHAS E MECANISMOS DE FALHAS

2.1 INTRODUÇÃO

Se não existisse a possibilidade de falha em um circuito integrado digital não haveria a necessidade de testá-lo. O tipo de uma falha pode influenciar o projeto do teste. Portanto, é apropriado discutir os tipos de falhas mais comuns que ocorrem em um circuito digital, antes de descrever os métodos de teste. Uma falha pode ser descrita como falta da performance esperada.

Para a eletrônica digital, a performance desejada é uma condição (ou seqüência) esperada de saída em resposta a uma condição (seqüência) de entrada. Uma falha é uma anomalia física, ou uma anomalia do projeto original do circuito integrado. Mais precisamente, em eletrônica, uma falha é um defeito como um circuito aberto, um curto-circuito, um circuito intermitente ou o aparecimento de um ponto de terra indesejado no circuito integrado.

Sempre que uma falha é introduzida erroneamente em um circuito integrado durante sua fabricação ou no campo por estresse ambiental, mecânico ou elétrico, cai em uma das seguintes categorias: um fio quebrado no circuito integrado, trilha de metalização rompida, quebra de um terminal ou um conector com um pino contaminado cria um circuito aberto. Espalhamento de solda faltando isolamento ou excessos de metal causam curto-circuitos. É comum, na prática da eletrônica digital, categorizar falhas pelos efeitos que elas causam no estado lógico de um circuito ao invés de apontá-las diretamente pelas causas. Esta organização será utilizada neste capítulo, onde também serão discutidas as falhas mais simples e mais comuns.

2.2 FALHAS TIPO "SINGLE STUCK-AT"

Uma falha tipo *single stuck at* faz com que um e somente um nó de um circuito digital permaneça em um determinado nível lógico. A falha tipo *stuck-at-1* (s-a-1) inibe o nó de comutar para 0, enquanto a falha tipo *stuck-at-0* (s-a-0) inibe a comutação para 1.

Para a lógica de 3 estados, a falha *stuck-at-off* (s-a-Z) pode ser definida como a impossibilidade de um sinal lógico de comutar em um barramento. A figura 2.0 mostra o diagrama esquemático do efeito de uma falha S-a-1, causada por um corte na trilha do circuito impresso. Podem ocorrer outros tipos de defeitos no CI ocasionando o mesmo erro na lógica do circuito. Uma trilha

aberta entre a metalização do *pad* e o fio que conecta o terminal de entrada ou saída do encapsulamento, por exemplo, poderia resultar em um aberto tanto na saída da porta A como na entrada da porta B. Os dois casos causarão uma falha tipo S-a-1 na porta TTL.

Talvez o mais indeterminado defeito na lógica comum é o defeito no habilitador de barramento. A saída de um *driver tri-state* pode estar amarrada no estado de alta impedância por um curto no caminho do habilitador. Neste caso, a falha é do tipo *stuck-at 1* (s-a-1) se o barramento estiver com um resistor de *pull-up* como carga, ou se o tempo para o circuito comutar para nível lógico 1 não for suficiente, como resultado de correntes de fuga (IIH) de uma porta que está sendo excitada pelo circuito.

O caso de um aberto no caminho do habilitador é mais grave, pois o circuito não irá para o estado de alta impedância, ocasionando uma colisão no barramento.

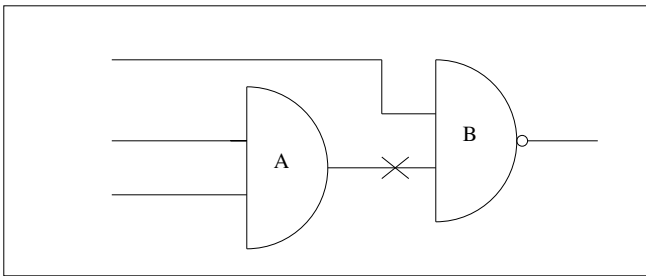


Figura 2.0 : O corte da trilha deste circuito resulta em uma falha tipo *stuck at 1* para a porta B.

2.3 MÚLTIPLAS FALHAS ”STUCK-AT”

Mesmo sendo aplicado com frequência, o modelo de falha tipo *single stuck at* é uma simplificação da realidade. Em circuitos reais, principalmente os que são fabricados pela primeira vez, quase sempre ocorrem múltiplas falhas. Um CI fabricado incorretamente pode gerar falhas em mais de um nó do circuito; uma corrosão ocorrida no *chip* causará provavelmente mais que um circuito aberto. Se as falhas não são interativas ou interdependentes, a situação de análise se torna direta. Dois circuitos abertos em diferentes *bits* de um barramento de comunicação podem não interagir. Neste caso, as falhas podem ser consideradas isoladas e tratadas como tal, uma vez que o teste para detectar um tipo não afetará a detecção do outro.

2.3.1 Múltiplas falhas interativas

Este tipo de falha é um caso bem diferente, podendo levar a imprecisões no grau de cobertura do teste. Se o barramento de comunicação descrito acima fosse verificado por um circuito detector de paridade, cuja saída fosse o único ponto de observação, a condição de duas falhas citadas acima poderia não ser detectada pelo teste de paridade.

Considere o caso em que 32 zeros serão aplicados no barramento. Paridade par será indicada, significando que o circuito está funcionando bem. Se duas das linhas do barramento estiverem com falhas tipo S-a-1, a paridade permanecerá par, sinalizando erroneamente que o circuito está bom. É óbvio que se as entradas do circuito detector de paridade fossem observadas, as falhas seriam identificadas.

2.3.2 Falhas "Bridging"

Estas falhas podem ser causadas pelo deslocamento do terminal de um componente que toca em outro circuito, formando um caminho de passagem para a corrente elétrica. Com o tempo, temperatura e campos elétricos trabalhando nos metais do circuito integrado, a migração de íons pode criar estranhos estalagmites que crescem e contactam outras regiões de metal e/ou silício no circuito.

Este tipo de falha é também conhecido como *Wisker*. O efeito no circuito envolvido depende da tecnologia de implementação da lógica. Para TTL, uma porta *wired-and* é criada. No caso do ECL esta falha cria um circuito *wired-Or*.

Quando uma falha tipo *bridging* conecta a saída de um circuito com sua entrada, uma malha de realimentação é estabelecida. Um exemplo simples é mostrado na figura 2.1. Esta falha pode converter o circuito combinacional em um *latch* sequencial, pela malha de realimentação.

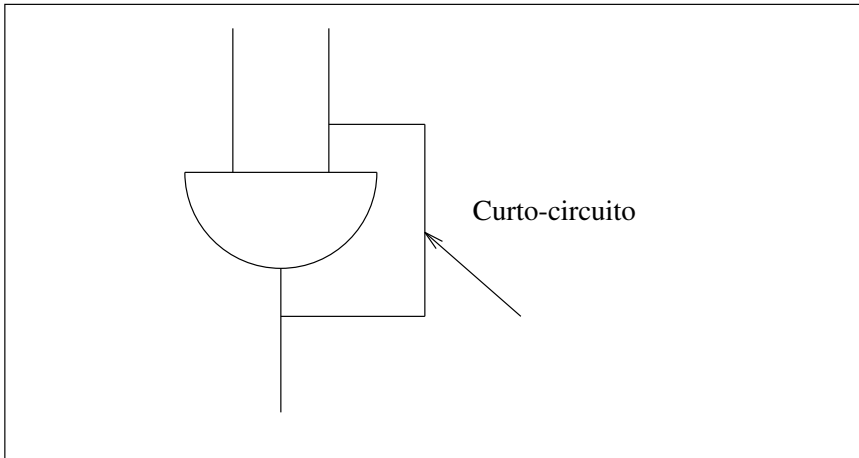


Figura 2.1 : Falha tipo *bridge* em uma porta *and* .

2.4 FALHAS INTERMITENTES

Qualquer circuito eletrônico, independente de sua velocidade, leva um certo tempo para desempenhar sua função. O processo de teste também requer um tempo para sua execução. Quando uma falha aparece e desaparece durante o teste, é dita falha intermitente.

Geralmente uma falha que aparece e desaparece, sem que nenhuma ação corretiva seja tomada, é uma falha intermitente. Mas, do ponto de vista de teste, a intermitência só deverá ocorrer durante a execução do programa de teste; caso contrário, não seria possível detectá-la.

Os mecanismos que causam falhas intermitentes são de natureza ambiental. É difícil de identificar uma falha intermitente apenas analisando a lógica do circuito integrado. É razoável, nestas circunstâncias, considerar a implementação física do circuito. Pode parecer estranho que um curto-circuito no transistor de saída, formando um S-at-0, possa desaparecer sozinho. Se este curto-circuito for resultado de partículas de metal livres, dentro do encapsulamento do CI, uma vibração pode temporariamente remover a falha.

2.4.1 Vibração

A vibração é realmente a principal causa de falhas intermitentes. Os condutores vibram e se tocam, placas se flexionam e se quebram, conexões soldadas fatigam e se soltam. Os

testes para verificar o desempenho de componentes eletrônicos sob condições de alta vibração podem ser realizados utilizando-se vibradores. Equipamentos sofisticados, com possibilidade de gerar padrões de vibração na frequência de áudio ou próximo da faixa de frequência supersônica, são disponíveis no mercado.

A dificuldade de se testar ou identificar falhas intermitentes está na coincidência da ocorrência da falha durante a execução do programa de teste. Modernos equipamentos automáticos de teste (ATE) podem facilmente ser configurados para repetir uma seqüência de teste *ad infinitum*, o que, sem dúvida, auxilia a detecção de falhas intermitentes. A limitação está no custo, já que o tempo de teste é caro.

2.4.2 Estresse térmico

O estresse térmico também causa falhas intermitentes, mas o período de tempo da intermitência é muito maior. A maioria dos problemas térmicos são causados pela diferença dos coeficientes de expansão térmica dos materiais utilizados para manufaturar dispositivos eletrônicos.

Realmente, com o aumento da temperatura, cada material cria um estresse no outro, resultando em uma expansão ou em uma contração. A tensão mecânica pode criar fendas e outros efeitos que resultam em falhas elétricas. Se a temperatura do circuito diminuir, a fenda pode se fechar, corrigindo temporariamente a falha. Neste caso, uma única passagem pela seqüência de teste tem boa probabilidade de detectar a falha se o teste for realizado na temperatura de ocorrência da falha, com auxílio de equipamentos térmicos especiais.

Como extremo distante das falhas intermitentes, estão as interferências eletromagnéticas, os picos (*glitches*) de alimentação, radiações cósmicas, etc, que podem ocasionar falhas espúrias de curta duração (nanossegundos) no circuito integrado. Estas falhas são praticamente impossíveis de se isolar para detecção. Felizmente, estas falhas não ocorrem com frequência. Nestes casos, o esforço maior está em proteger o circuito destes tipos de falha.

2.5 DESCARGA ELETROSTÁTICA - ESD

Sempre que dois materiais entram em contacto , com um certo movimento entre eles e então são separados rapidamente, uma carga de igual intensidade e polaridade oposta é desenvolvida nas superfícies.

Se um dos materiais é um condutor, ele mantém a carga até que seja aterrado, e esta carga é considerada uma carga móvel. Se o material for um isolante, a carga é imóvel e ela permanece no material até ser neutralizada de alguma maneira .

A ESD (descarga eletrostática) é uma das causas importantes das falhas nos vários estágios de um dispositivo semicondutor: na produção, na montagem, no teste, na instalação e no campo. Daí a importância do correto manuseio dos componentes eletrônicos. A utilização de pulseiras, calcanheiras e mantas antiestáticas é muito importante quando se está manuseando componentes semicondutores, especialmente em se tratando de dispositivos fabricados com tecnologia CMOS. A utilização de embalagem apropriada para o transporte e armazenagem de circuitos integrados, como canaletas e sacos antiestáticos, podem evitar falhas causadas por ESD.

Existem três fontes principais de carga que podem levar a eventos de descarga eletrostática, podendo danificar ou alterar significativamente as características do dispositivo semicondutor:

- HBM (Human body model) Modelo do corpo humano.

Uma pessoa carregada toca um dispositivo e descarrega a sua carga armazenada para ou através do terra do dispositivo.

- CDM (Charged Device Model) Modelo do dispositivo carregado.

O próprio dispositivo atuando como uma placa de um capacitor pode armazenar cargas. No contacto com um terra efetivo, o pulso de descarga pode danificar o dispositivo.

- FIM (Field induced model) Modelo de campo induzido.

Um campo eletrostático está sempre associado com objetos carregados. Sob circunstâncias particulares, um dispositivo inserido neste campo pode ter um potencial induzido através de um

óxido, que ocasiona um *breakdown* .

Os danos provocados por ESD podem tanto provocar a falha completa do dispositivo por desvios paramétricos, curtos ou abertos, como levar ao enfraquecimento do dispositivo, pelo aquecimento, derretimento, por danos nos óxidos, nas junções ou em componentes do dispositivo. Os tipos de falhas provocadas por ESD podem ser classificados como :

- falhas de corrente ou de potência, comumente reconhecidas por pontos quentes ou zonas de derretimento;
- falhas de potencial que estão caracterizadas onde tenha ocorrido o rompimento do dielétrico ou a quebra da superfície ou um arco voltáico.

Geralmente o teste de curto e aberto já detecta a falha, quando um CI sofreu uma descarga eletrostática de alta intensidade. Se o equipamento de teste estiver operando no modo *continue on fail*, os demais testes também apresentarão falhas, principalmente o teste de *leakage* e o teste funcional.

2.6 SOBRE ESTRESSE ELÉTRICO - EOS

A causa fundamental deste tipo de falha é o sobreaquecimento do material por onde uma corrente excessiva está circulando, provocando uma situação de corrida térmica . Este processo pode levar ao derretimento do silício no ponto de maior temperatura, destruindo sua estrutura cristalina e ocasionando um curto na junção.

Pode ser observado que quando a resistência de junção cai dramaticamente como resultado do derretimento do silício, pode ser esperado um acréscimo na dissipação de potência das interconexões do metal. Se o transiente elétrico que originou estresse continuar, o metal pode derreter. Se o transiente for uma alta tensão, um arco voltáico pode ocorrer, ocasionando danos severos ao dispositivo.

Não é fácil para o analista de falhas definir se a causa da falha foi por ESD ou por EOS, devido a semelhança nos danos causados no dispositivos pelos dois mecanismos.

2.7 CORROSÃO

A corrosão de estruturas metálicas de filme fino nos componentes de estado sólido, como nos circuitos integrados, constituem a principal causa deste tipo de falha. Estas estruturas metálicas, usualmente constituídas de alumínio ou de ouro, podem abranger contactos elétricos do semicondutor, linhas de interconexão e *pads*. Com exceção dos *pads*, estas estruturas são quase sempre cobertas por uma camada de material vítreo inorgânico e/ou polímero orgânico (passivação) que protege o dispositivo quimicamente e mecanicamente. Estes filmes de passivação, quando não apresentam defeitos, previnem com muita eficiência os dispositivos que estão sob esta camada. Entretanto, defeitos estruturais na camada de passivação ou a presença de áreas de metal livres desta camada, como nos *pads*, tornam o metal vulnerável ao ataque da corrosão, pelos agentes químicos do ambiente, do encapsulamento plástico, ou da presença de atmosfera nos encapsulamentos defeituosos.

A corrosão do metal de alumínio nos semicondutores é freqüentemente acelerada na presença de contaminantes como umidade, íons clorídricos e íons de sódio. O principal efeito da corrosão é um aumento da resistência ôhmica, dificultando a passagem da corrente elétrica. Em muitos casos a corrosão leva a um aberto da ligação, acarretando em falha no dispositivo.

2.8 FALHAS EM MEMÓRIAS

Provavelmente um dos mais complexos grupos de falhas consiste das falhas de memórias. A extrema miniaturização para um constante aumento da capacidade de armazenamento dos *chips* de memória, torna-os sujeito a vários efeitos de interdependência, acarretando o que é comumente chamado de “falhas de vizinhança” (*neighborhood faults*).

A tendência para uma particular falha depende da disposição física dos elementos internos de armazenamento e dos caminhos de acesso a estes elementos. A figura 2.2 mostra esta diferença. Além das variações térmicas e das flutuações da alimentação, o tempo é um fator a mais nas memórias dinâmicas.

Como exemplo, uma seqüência típica de teste de uma memória RAM dinâmica será explorada. Colocar 1M, 4M, 16M células em um único *die* requer um projeto de célula que

deixe um espaço mínimo entre todos os elementos. Com este espaçamento tão reduzido, efeitos como migração de cargas e efeitos de campo se tornam críticos para a operação. O mecanismo de armazenamento é um capacitor.

Infelizmente, a implementação interna do *chip* de memória não é apresentada no *datasheet* do fabricante. Deste modo, um teste pode ter grau de cobertura diferente, para um mesmo componente, de fabricante diferente, com disposição diferente das células internas de memória. Testes exaustivos que verificam a interferência entre dois bits quaisquer da memória eliminam a dependência do *layout* do circuito, mas o tempo de teste aumenta, aumentando o custo.

Seqüências de padrões de teste devem ser cuidadosamente elaboradas para verificar as fallhas fundamentais em primeiro lugar. Verificar curtos ou abertos nas linhas de endereçamento, escrevendo dados em cada linha de endereçamento e então verificando os dados escritos por uma seqüência de leituras é um teste popular para memórias com organização interna tipo 8K x 8, 16K x 8, 32K x 8, etc.

Para memórias verticais, como as de 1M x 1, 4M x 1, etc, uma técnica diferente deve ser aplicada. Uma das seqüências de testes bastante utilizadas é o padrão *checkerboard*, para memórias com mais de uma linha de dados. Este teste coloca cada linha de dado em “0” e todas as linhas adjacentes em “1”, e vice-versa. Este padrão detecta curtos, abertos e falhas tipo *bridging* entre as linhas de dados.

O teste de memória deve verificar a operação de cada célula e a interdependência entre as células. A essência deste procedimento é selecionar cuidadosamente cada padrão de teste para cobrir um grupo de falhas que pode ocorrer na memória sob teste.

Existem equipamentos automáticos de teste especialmente projetados para testar memórias, com facilidades para a geração dos diversos tipos de padrões de teste.

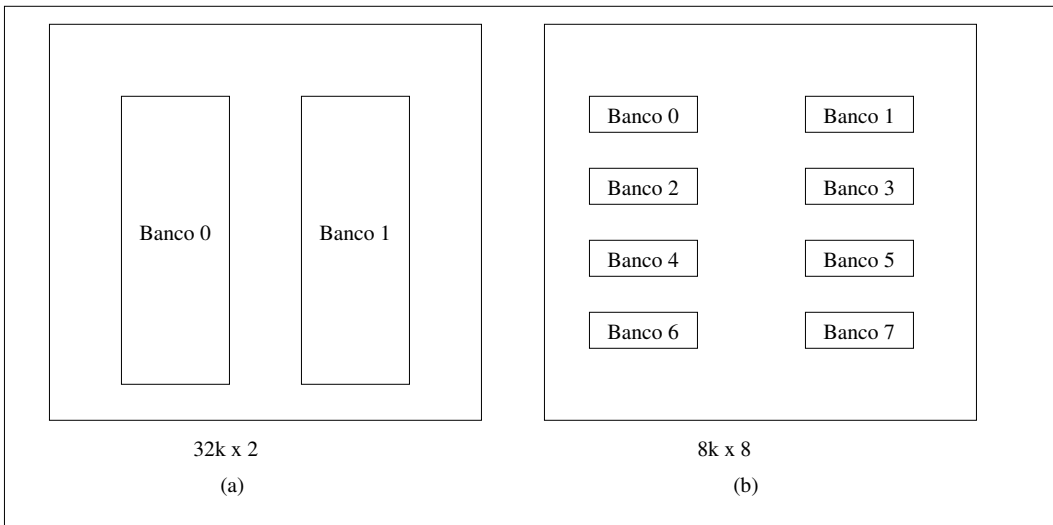


Figura 2.2 : Estruturas internas alternativas para uma RAM de 64k.

2.8.1 Principais falhas nas memórias

As falhas mais comuns que devem ser detectadas na memórias são:

- Curtos e abertos nas células
- Curtos e abertos nos endereços
- Distúrbios envolvendo a célula
- Cobertura de escrita
- Tempo de acesso

2.8.2 Descrição das falhas

- Célula em curto - Célula curtocircuitada para zero ou para um (VCC ou GND).
- Célula aberta - Célula não muda de estado quando for realizada uma escrita.
- Célula adjacente em curto - Dependendo da natureza do curto, escrever em uma determinada célula, faz com que também seja escrito o mesmo valor (ou seu complementar) na célula adjacente.

- Endereços em curto ou em aberto - Algumas células possuem mais que um endereço e alguns endereços acessam mais que uma célula. Este tipo de defeito não é fácil de ser detectado, uma vez que o engenheiro de teste não vê realmente a célula, mas sim apenas a resposta de saída a um dado endereço, e não sabe fisicamente qual célula foi realmente lida.
- Distúrbios nas células - Todas as células devem manter seus estados após uma escrita e durante uma leitura. Algumas vezes, ler ou escrever em uma célula pode ocasionar uma mudança de estado na célula vizinha.
- Cobertura de escrita - É o intervalo de tempo requerido antes que a próxima célula possa ser lida, após uma escrita.
- Tempo de acesso - Possui vários significados, mas é geralmente o tempo entre a habilitação de leitura, *chip select* ou variação do endereço e o aparecimento do dado na saída. O tempo de acesso pode depender da seqüência na qual as células da memória são acessadas.

2.8.3 Padrões de teste

Padrões de teste são seqüências de padrões de níveis lógicos altos e baixos aplicados à memória para testá-la. A capacidade de um determinado padrão de teste detectar uma falha depende, principalmente, do projeto, geometria e *layout* das células da memória. Os padrões para teste de memória mais comuns são :

- Padrão de teste de “ Um e Zero ”
- Padrão de teste *checkerboard*
- Padrão de teste Marchante
- Padrão de teste Caminhante
- Padrão de teste Galopante de cobertura de escrita.

2.9 REFERÊNCIAS BIBLIOGRÁFICAS

- HUNGER, B. A. *Electrostatic Discharge* . IRPS 1981, pp. 193-199.
- CORTNER, J. Max. *Digital test engineering* . " Wiley-Interscience ", 1987.
- STEVENS, Anthony K. *Introduction to component testing*
- CORSI, A; MORANDI, C; *A review of ram testing methodologies; Microelectronics Journal, Vol 14, N. 2, 1983.*
- SMIDTTH, J. S. *Electrical Overstress Failure Analysis in Microcircuits* IRPS 1978, pp 41-46.
- SCHNABLE, G. L., Comizzoli, R. B. , Kern,W., White, L. K. *A Survey of Corrosion Failure Mechanisms in Microeletronic Devices* . RCA Review, vol 40, December 1979.

Capítulo 3

TESTES REALIZADOS POR UM ATE

3.1 TIPOS DE TESTES

Há três tipos de testes elétricos realizados em um componente digital:

- Funcional
- Paramétrico DC
- Paramétrico AC

3.2 FUNCIONAL

Consiste em verificar se o componente está funcionando corretamente, ou seja, verificar se o componente está desempenhando corretamente a função, ou as funções para as quais foi projetado. Neste teste, geralmente são aplicados, na entrada do CI, padrões ou seqüência de padrões de “0” e “1” com amplitude e duração pré-definidas, e a saída é comparada com o padrão ou a seqüência de padrões esperados em um dado instante de tempo. Basicamente, em circuitos digitais mais simples, a verificação da tabela da verdade é suficiente para o teste funcional.

O teste funcional é a parte mais complexa de um programa de teste. Sua complexidade é diretamente proporcional à complexidade do CI. Geralmente o padrão de teste para CI's LSI, VLSI são gerados juntamente com o projeto do *chip*, com o auxílio de simulador. Tais padrões de teste são extremamente complexos e grandes.

Normalmente o nível de tensão aplicado à entrada do CI para “rodar” o teste funcional é V_{IH} mínimo para nível lógico “1” e V_{IL} máximo para nível lógico “0”, para a tensão de alimentação nominal. A saída é comparada com os valores de V_{OH} mínimo e V_{OL} máximo, para uma corrente de carga de $I_{OHmáx.}$ e $I_{OLmáx.}$ Estes valores de corrente de carga, de tensão de entrada e de tensão de saída são especificados no manual do componente ou da tecnologia. Neste teste está embutido parte do teste paramétrico DC.

O tempo de duração do nível lógico “1” e do nível lógico “0” aplicado na entrada do *chip* e o tempo exato em que a saída deve ser comparada com os valores de referência é totalmente controlado por *software* nos equipamentos de testes modernos; por isso, muitas vezes, o teste funcional

contém também parte do teste paramétrico AC.

No caso de programas de teste para qualificação de componentes, realiza-se o teste funcional três vezes, com tensão de alimentação nominal, máxima e mínima.

3.3 PARAMÉTRICO DC

Neste tipo de teste os parâmetros DC do componente são medidos de duas maneiras, a saber:

- Forçando-se uma corrente programada no dispositivo e medindo-se a tensão.(Este tipo de procedimento é feito para as medições de tensão como no caso de VOH e VOL).
- Forçando-se uma tensão programada ao dispositivo e medindo a corrente.(Este tipo de procedimento é feito para as medições de corrente como no caso de IIL e IIH).

Os principais parâmetros de tensão e corrente máxima e mínima, especificados para a operação funcional do dispositivo são:

- **IIL** - Corrente de fuga de entrada para nível baixo máximo aplicado.
- **IIH** - Corrente de fuga de entrada para nível alto máximo aplicado.
- **IOZH** - Corrente de fuga de uma saída em alta impedância para VOL máximo aplicado.
- **IOZL** - Corrente de fuga de uma saída em alta impedância para VOH mínimo aplicado.
- **VIL** - Tensão máxima de entrada nível baixo (limite para o nível lógico “0”).
- **VIH** - Tensão mínima de entrada nível alto (limite para o nível lógico “1”).
- **VOL** - Tensão máxima de saída nível baixo (limite do nível lógico “0”).
- **VOH** - Tensão mínima de saída nível alto (limite do nível lógico “1”).
- **IOS** - Corrente de curto-circuito na saída.
- **IDDdin** - Corrente de consumo dinâmico máxima.

- **IDDest** - Corrente de consumo estático máxima.
- **IOL** - Corrente de saída mínima para determinada tensão de saída nível baixo (VOL).
- **IOH** - Corrente de saída mínima para determinada tensão de saída nível alto (VOH).

3.3.1 Testes paramétricos realizados

- Continuidade (curto e aberto)
- Fuga - IIH,IIL,IOZ
- Limiares (*threshold*) - VIL,VIH
- Tensões de saídas VOL p/IOL máx. e VOH p/IOH máx.
- Consumo de potência (ICC,IDD)
- Corrente de curto-circuito - (IOS)

3.3.2 Teste de continuidade

A finalidade do teste de continuidade é assegurar que a *interface* do testador está em contacto com o dispositivo sob teste. No teste de continuidade é verificado também se há curto-circuito ou circuito aberto nos terminais do CI. Isto é feito aplicando-se uma corrente conhecida ($+ou - 100\mu A$) no pino e medindo-se a tensão resultante. Se a tensão do pino estiver próxima de 0V, o diodo de proteção do pino está defeituoso, causando um curto-circuito. Se a tensão do pino atingir um certo valor crítico, isto indica que existe um circuito aberto nos circuitos do pino.

Para um dado pino passar no teste de continuidade, a tensão no pino deve estar entre a tensão de aberto e de curto-circuito. Os limites de tensão que dividem esta região dependem do dispositivo que está sendo testado. Entretanto, como regra geral para falhas de curto-circuito, os limites de tensão estão na faixa de $+ ou - 0.5V$ e para falhas de aberto de $+ ou - 4.0V$. Usualmente este é o primeiro a teste a ser realizado. A ordem dos testes em um programa comum é :

- Teste de Continuidade

- Teste Funcional
- Teste Paramétrico DC
- Teste Paramétrico AC

A figura 3.1 mostra os limites para o teste de continuidade.

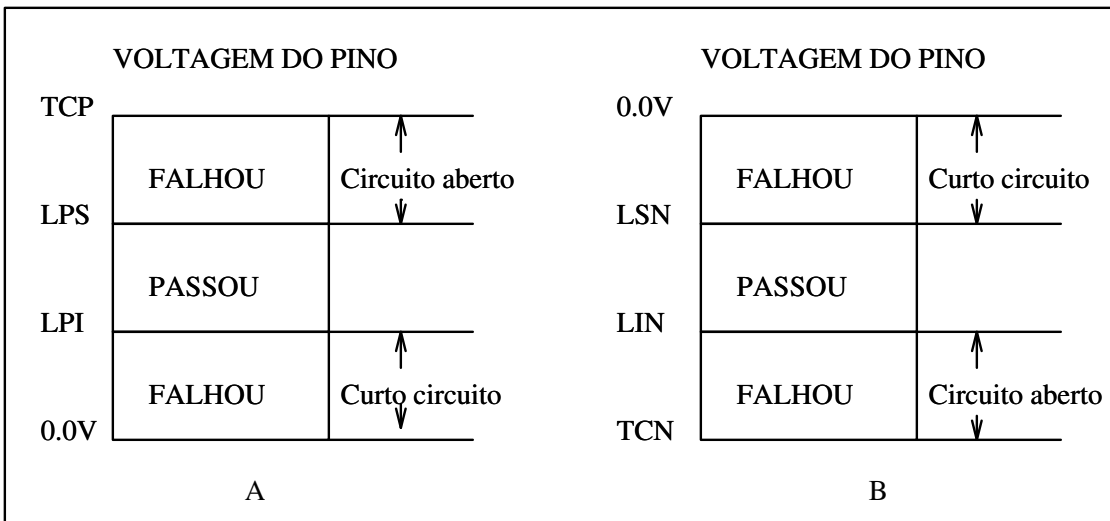


Figura 3.1: Regiões de aceitação/rejeição para o teste de continuidade.

A : Regiões de aceitação/rejeição para correntes de teste positivas

B : Regiões de aceitação/rejeição para correntes de teste negativas

LSN : Limite superior negativo

LPS : Limite positivo superior

LIN : Limite inferior negativo

LPI : Limite positivo inferior

TCN : Tensão de carga negativa

TCP : Tensão de carga positiva

3.3.3 Medição de IIL

A medição de IIL é realizada forçando-se VIL na entrada desejada e medindo-se a corrente desta entrada com as demais entradas em nível lógico “1”, com o *chip* alimentado com VCC máximo e com as saídas em aberto. Quando se utiliza equipamentos automáticos de teste este processo é automaticamente repetido para cada entrada do *chip* que foi programada para ser medida. A figura 3.2 a seguir, ilustra o procedimento para a medição de IIL.

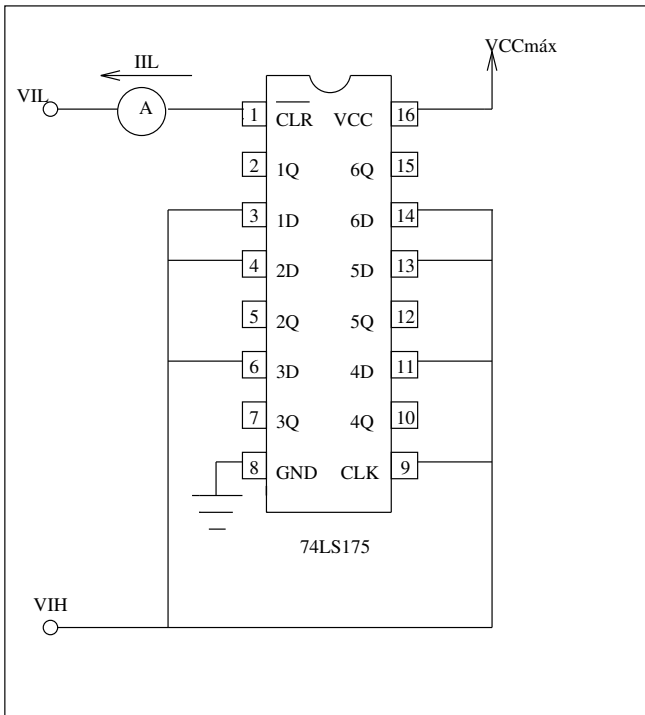


Figura 3.2 : Medição de IIL.

3.3.4 Medição de IIH

Semelhante à medição de IIL, a medição de IIH é realizada forçando-se VIH na entrada desejada e medindo-se a corrente desta entrada com as demais entradas em nível lógico baixo, para o *chip* alimentado com VCC ou VDD máximo, sem carga na saída. Quando se utiliza equipamento automático de teste, este processo é repetido para cada entrada que foi programada para a medição de IIH, ou seja, o micro ou nano amperímetro é colocado em série com a entrada a ser

medida, com um nível de tensão em uma das extremidades de V_{IH} e as demais entradas com nível lógico baixo. O programa vai comutando o micro amperímetro nas entradas onde se deseja medir a corrente e coloca V_{IH} nas demais entradas. A figura 3.3 ilustra o procedimento para a medição de I_{IH} .

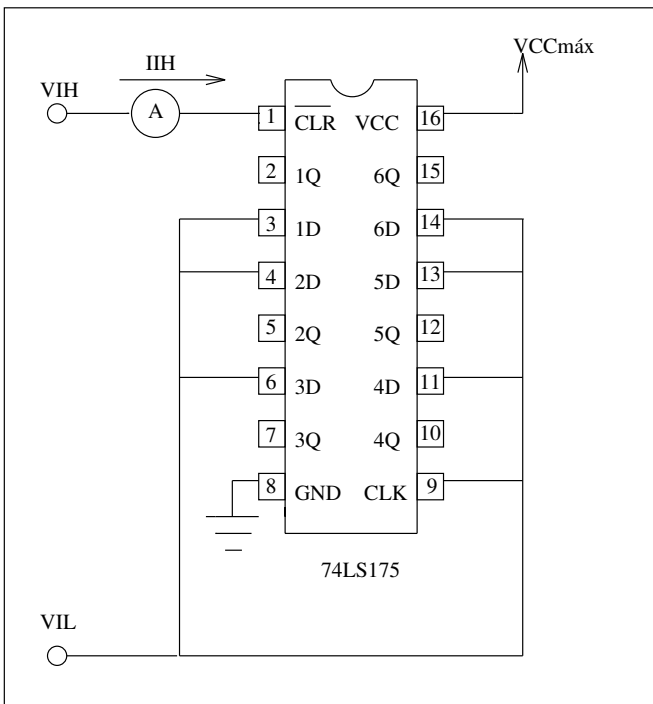


Figura 3.3 : Medição de I_{IH} .

3.3.5 Medição de IOZ

Idealmente, as saídas *Tri-State* são consideradas como circuitos abertos. Na realidade, aparecem como uma alta impedância. A quantidade de corrente que é consumida ou fornecida por cada uma das saídas *Tri-State* é chamada de fuga. Para se medir esta corrente o CI é levado, por um padrão de teste, à condição de *Tri-State*. A corrente de fuga é medida aplicando-se V_{GND} ou V_{CC} e medindo-se a corrente IOZ . São realizadas duas medições desta corrente; $IOZH$ onde é aplicado na saída V_{CC} e $IOZL$ onde é aplicado na saída V_{GND} . Ambas medições são para V_{CC} máximo de alimentação de um *chip* TTL.

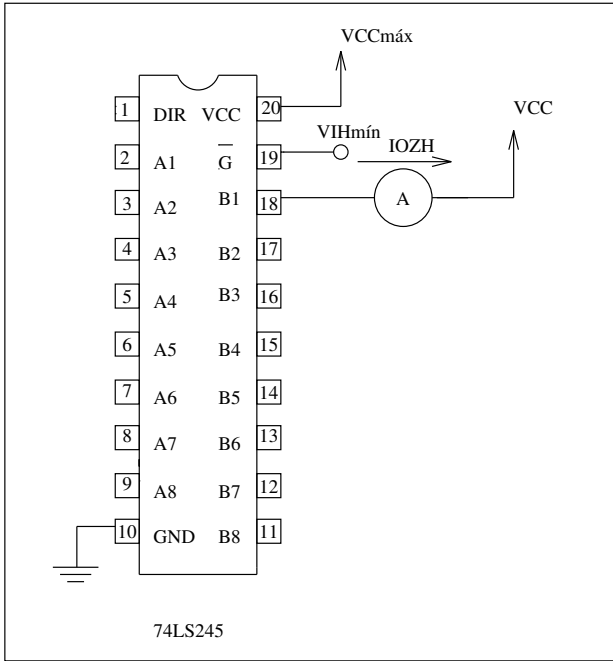


Figura 3.4 : Medição de IOZH.

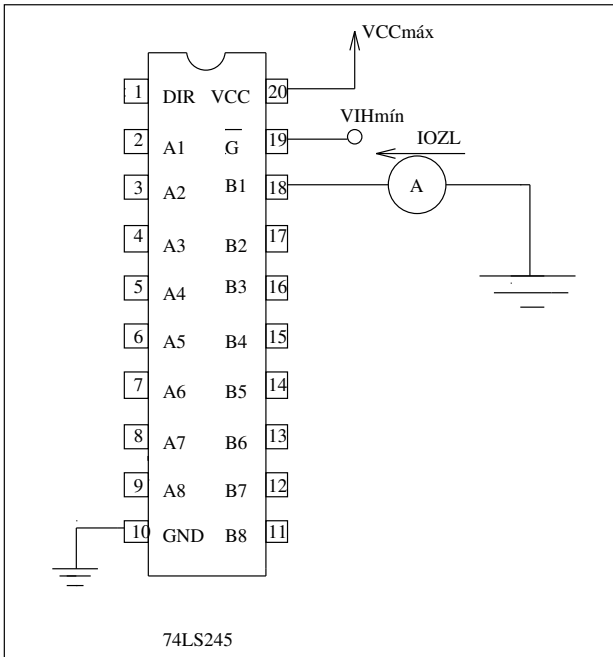


Figura 3.5 : Medição de IOZL.

Tais medições variam um pouco de acordo com a tecnologia do componente. É importante consultar o manual do componente para realizar as medições, colocando-se as mesmas condições de medições especificadas para um dado teste. As figuras 3.4 e 3.5 ilustram as medições de IOZL e IOZH para o CI 74LS245

3.3.6 Medição de VIH E VIL

Estas duas medições determinam o valor real de VIH e VIL que o dispositivo irá operar e são realizadas principalmente na fase de caracterização de um *chip*. São medições lentas feitas por aproximação sucessiva.

Em uma avaliação funcional de um CI por exemplo, o valor de VIH_{mín.} e de VIL_{máx.}, especificado no manual do fabricante do componente, é aplicado às entradas do CI para rodar o padrão funcional. VIL é a tensão máxima que a entrada do dispositivo irá comutar para o estado baixo quando estiver transicionando do estado alto. VIH é mínima tensão na qual a entrada do dispositivo irá comutar para o estado alto quando estiver transicionando do estado baixo. A medição de VIH pode ser feita da maneira indicada no fluxograma da figura 3.6.

Do fluxograma podemos ver que aplica-se VCC na entrada do CI e roda-se um teste funcional com este nível de tensão na entrada do *chip*. Se o componente passar no teste funcional, o valor da tensão de entrada VI é diminuído do valor X e o componente é submetido ao teste funcional novamente. Este processo é repetido até que o componente falhe no teste funcional. Neste ponto o valor de VIH_{mín.} será o valor de VI somado ao valor de X, ou seja, o valor de VI anterior à falha

No valor escolhido para X está a exatidão da medição. Quanto menor for X, mais exata será a medição, mas também será mais demorada. Como na fase de caracterização de um componente o tempo de medição não é crítico, deve-se escolher um valor para X bem pequeno, da ordem de milivolts.

Para se obter resultados mais rápidos não é necessário iniciar a aproximação com o valor VCC aplicado à entrada e sim um valor próximo da tensão de limiar de VIH_{mín.}

Para a medição de VIL_{máx.}, o procedimento é semelhante ao de VIH_{mín.}. O fluxograma da figura 3.7 ilustra este procedimento.

Inicia-se a medição colocando-se zero *volts* em VI (tensão de entrada do componente) e realizando o teste funcional. Se o componente passar, o valor de VI é acrescido de X e o componente é submetido ao teste funcional novamente. Este processo é repetido até que o componente falhe no teste funcional. Assim, o valor de $V_{IHm\acute{a}x.}$ será o valor de VI menos o valor de X, ou seja, o valor de VI anterior à falha.

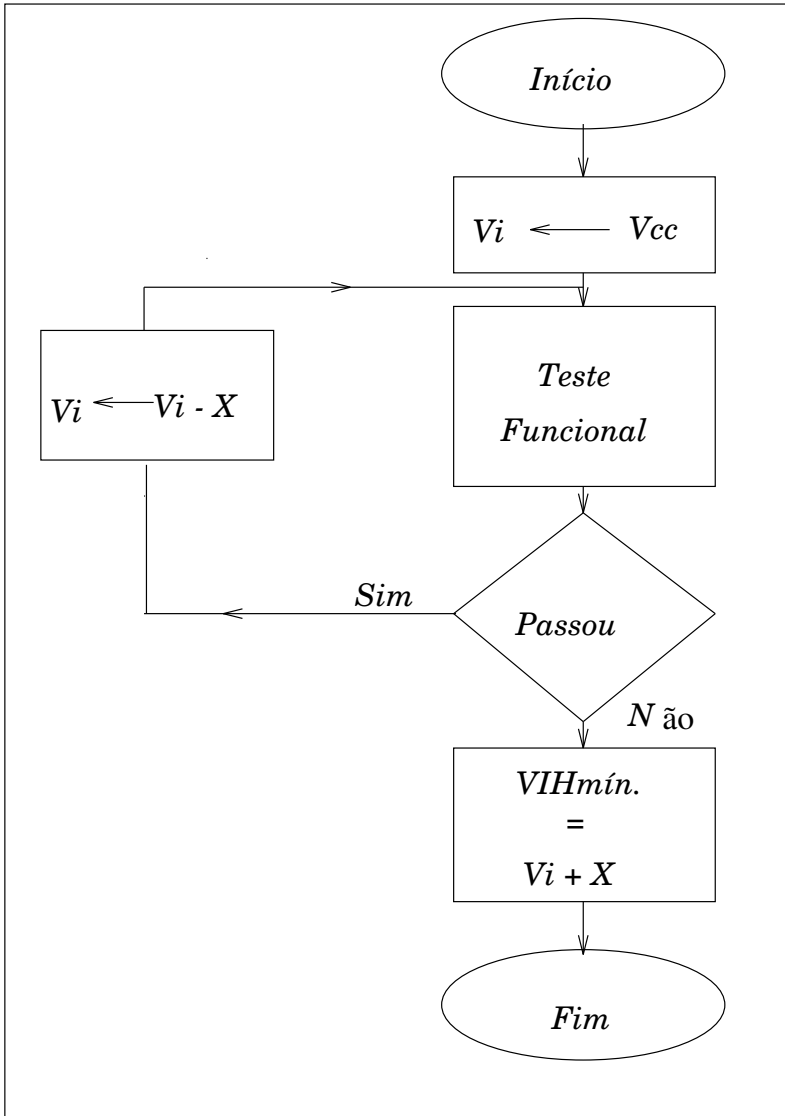


Figura 3.6 : Fluxograma para a medição de VIH.

Igualmente à medição de VIH, a exatidão da medição está ligada ao valor de X . Quanto menor for este valor, maior será a exatidão da medição. A mesma consideração anteriormente citada para o tempo de medição vale aqui também, ou seja, a estimativa para realizar esta medição é lenta, mas o tempo, na fase de caracterização, não é crítico.

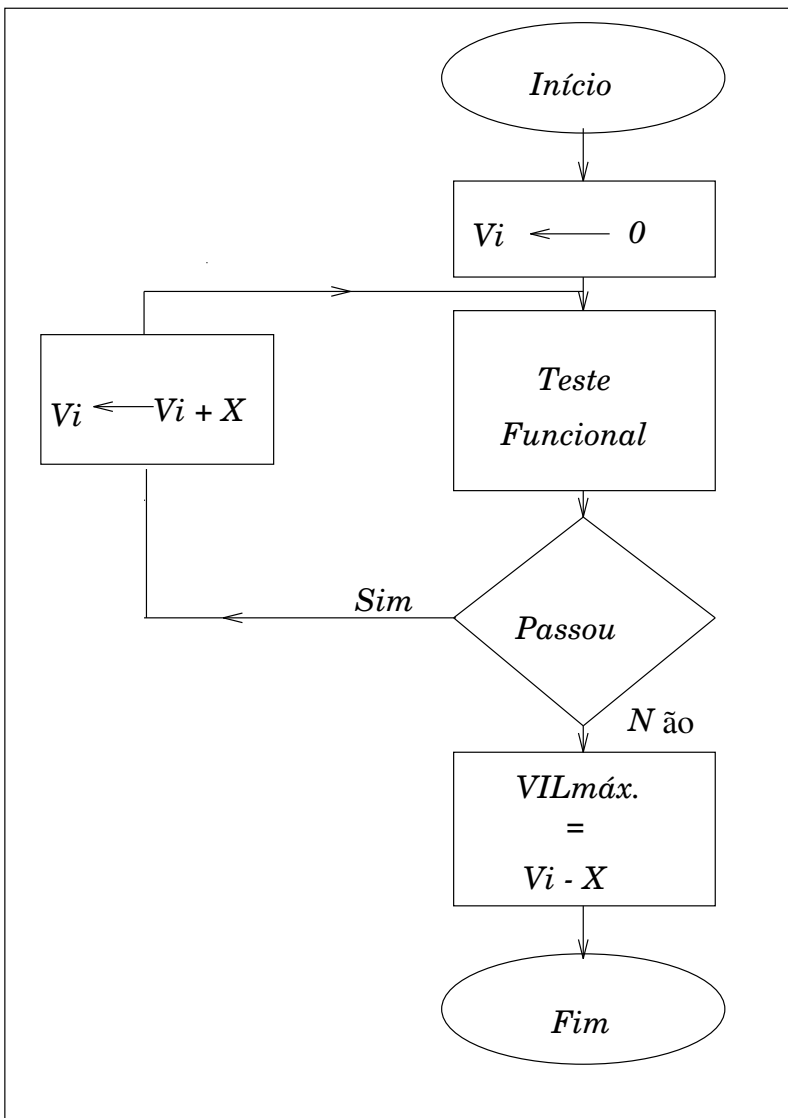


Figura 3.7 : Fluxograma para a medição de VIL.

3.3.7 Medição de VOH E VOL

Nas medições de tensão uma corrente é forçada ao componente, e a tensão é medida. No caso de VOH, por exemplo, $I_{OHm\acute{a}x.}$ é forçado no componente e o nível da tensão no pino de saída do CI é medido e comparado com o valor limite do manual.

Para se medir VOL, é forçado ao componente o valor de $I_{OLm\acute{a}x.}$, mede-se a tensão de saída e compara-se o resultado com a referência do manual.

É importante salientar que, para medir VOH, deve-se ter a condição de saída em nível lógico “1”. Isto é conseguido geralmente executando-se um padrão funcional que coloque as saídas do componente em nível lógico “1”. O mesmo ocorrendo para VOL, deve-se rodar um padrão de teste funcional que coloque as saídas do componente em nível lógico “0”. Somente depois deste passo é que a corrente é forçada e a tensão é medida.

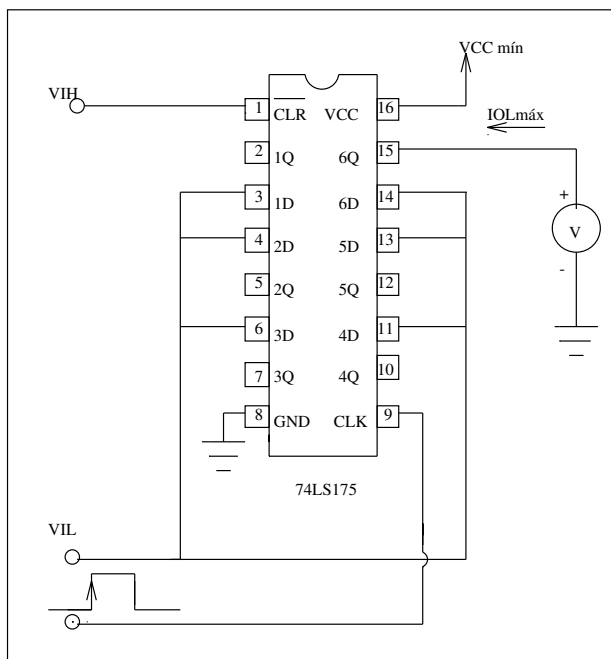


Figura 3.8 : Medição da tensão VOL.

Nos equipamentos automáticos de teste o voltímetro e as correntes de cargas, $I_{OHm\acute{a}x.}$ ou $I_{OLm\acute{a}x.}$ são comutadas automaticamente para cada saída programada para ser medida. Para ambas medições, o *chip* é alimentado com VCC mínimo. As figuras 3.8 e 3.9 mostram as

medições de VOH e VOL.

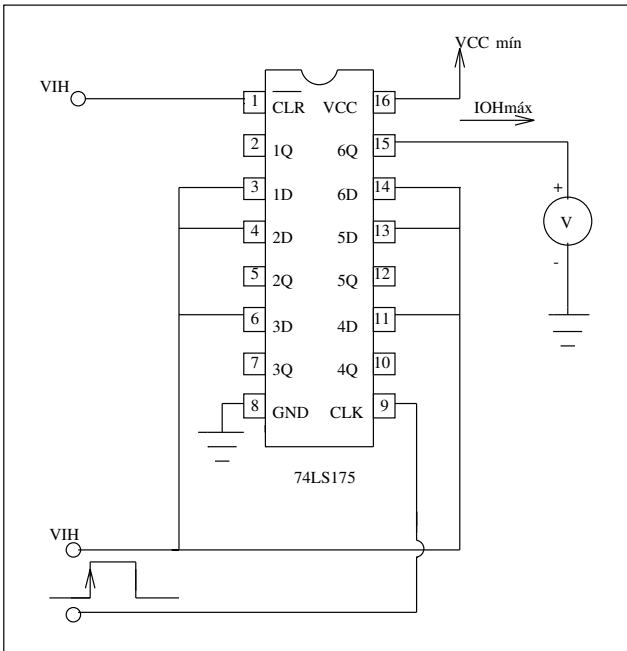


Figura 3.9 : Medição da tensão VOH.

3.3.8 Medição de IDD ou ICC

A quantidade de corrente que um componente consome pode variar, dependendo da frequência do sinal que é aplicado nas entradas do *chip*, principalmente se a tecnologia do dispositivo for CMOS.

O consumo estático de um componente (ICC) é medido com o *chip* alimentado com VCC máximo, com as entradas em um determinado nível fixo de tensão e com as saídas em aberto.

Nos manuais TTL são apresentadas duas correntes de consumo chamadas de ICCH e ICCL. O ICCH é medido em série com o pino de alimentação, com o *chip* alimentado com VCC máximo, com as entradas em níveis de tensão de acordo com a tabela 3.1. A medição de ICCL é semelhante a de ICCH, mas com a condição de entrada diferente (veja a diferença na tabela 3.1).

Para CI's CMOS não existe no manual ICCH e ICCL e sim ICC ou IDD. Esta medição é realizada com as entradas do componente em VCC ou em 0V para o CI alimentado com a

tensão máxima.

FUNÇÃO	CONDIÇÕES DE ENTRADA P/ ICCH	CONDIÇÕES DE ENTRADA P/ ICCL
NAND	Todas as entradas em GND	Todas as entradas com 4,5V
AND	Todas as entradas em 4,5V	Todas as entradas em GND
NOR	Todas as entradas em GND	Uma entrada em 4,5V, demais em GND
OR	Uma entrada em 4,5V, demais em GND	Todas as entradas em GND

Tabela 3.1 :Condições de entrada para a medição de ICCH e ICCL.

Para circuitos integrados tipo *flip flop*, as condições de entrada para a medição de ICC são outras, diferentes destas especificadas na tabela 3.1 e estão apresentadas no manual do componente. A figura 3.10 ilustra a medição de ICCH para o CI 74LS30, uma porta E de oito entradas.

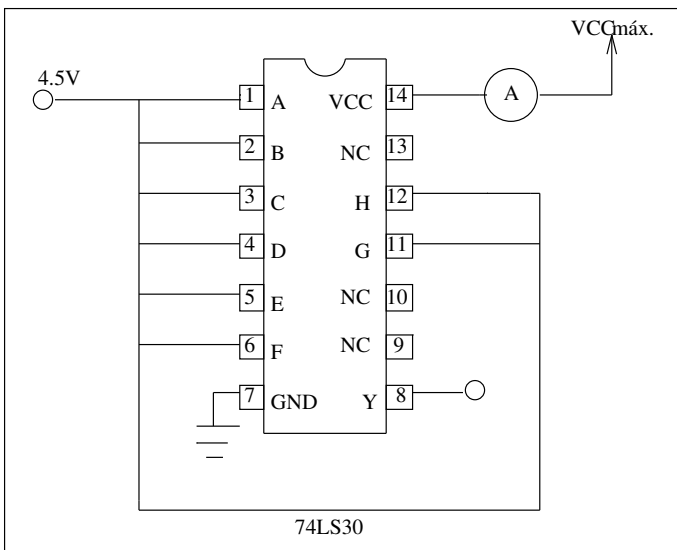


Figura 3.10 : Medição da corrente de consumo estático

A medição de corrente de consumo dinâmica de um componente é realizada com um padrão de teste, ou seja, uma seqüência de “0” e “1” aplicados às entradas do componente; isto é, o nível lógico de cada entrada fica alternando em uma determinada freqüência. As saídas são mantidas em aberto e o *chip* é alimentado com VCC ou VDD máximo. A figura 3.11 ilustra este procedimento.

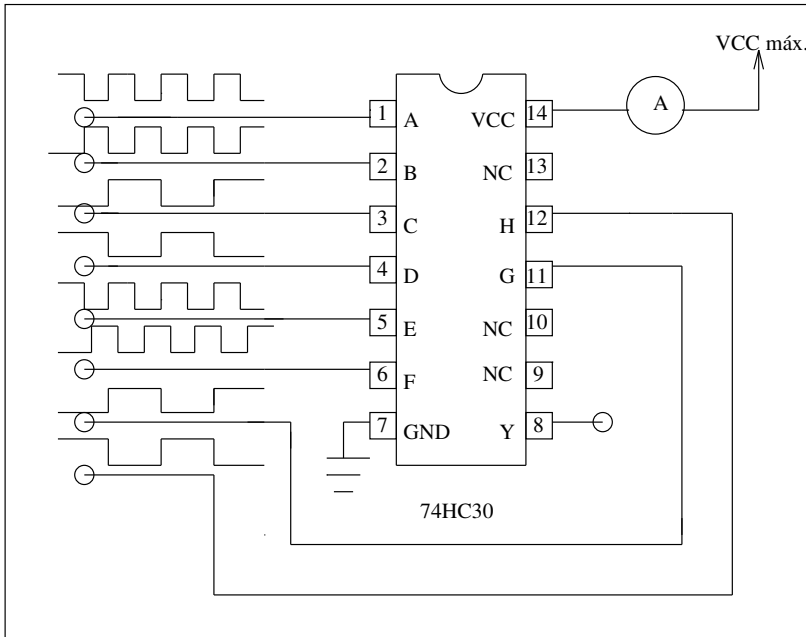


Figura 3.11 : Medição da corrente de consumo dinâmico.

3.3.9 Medição de IOS

Para realizar esta medição coloca-se uma ponta do microamperímetro na saída que se deseja medir e a outra ponta é no terra. Como a resistência ôhmica do microamperímetro é muito pequena, tem-se praticamente um curto-circuito da saída do componente para o terra. Antes de se inserir o microamperímetro é necessário colocar nas entradas do componente uma condição tal que as saídas vão para nível lógico “1”. Somente depois que a saída vai para “1” é que a corrente é medida. Este procedimento deve ser rápido, para não danificar o componente.

Este teste deve ser realizado no final do programa de teste, pois aquece o componente, podendo alterar os valores das outras medições.

O *chip* é alimentado com VCC máximo. A figura 3.12 ilustra a medição de IOS. A tabela 3.2 apresenta as condições de medição da corrente de curto-circuito para as portas AND, NAND, OR, NOR .

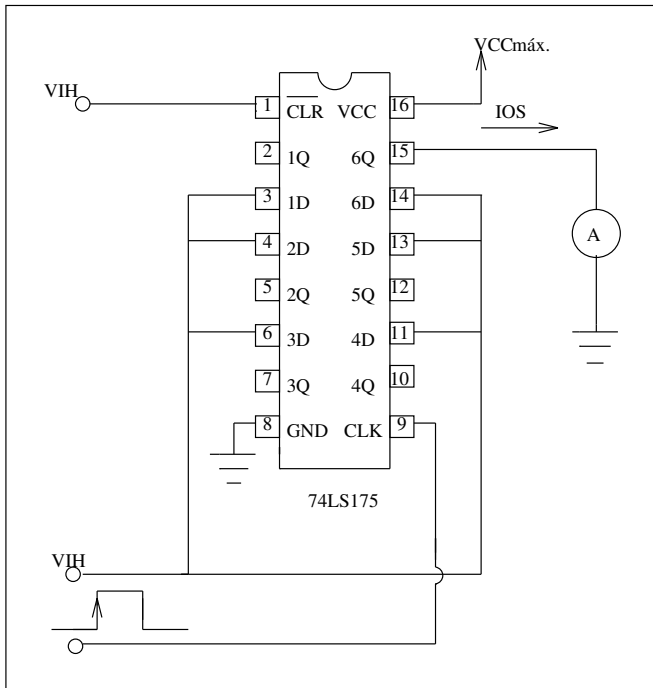


Figura 3.12 : Medição da corrente de Curto-Circuito.

FUNÇÃO	CONDIÇÃO DE ENTRADA
NAND	Todas as entradas em GND
AND	Todas as entradas em 4,5V
NOR	Todas as entradas em GND
OR	Todas as entradas em 4,5V
AND-OR-INVERTER	Todas as entradas em GND
AND-OR	Todas as entradas em 4,5V

Tabela 3.2 :Condições de entrada para a medição de IOS.

3.3.10 Medição de IOL

Esta corrente geralmente não é medida nos testes elétricos mais comuns. Seu valor máximo, fornecido pelo manual do fabricante do componente, é utilizado como corrente de carga para a realização do teste funcional e para a medição de VOL. A corrente IOL deve ser medida na fase de caracterização, forçando-se uma tensão de VOL na saída do componente e medindo-se a corrente resultante. Obviamente, o *chip* deve ser levado à condição de saída de nível lógico baixo. A figura 3.13 ilustra este procedimento.

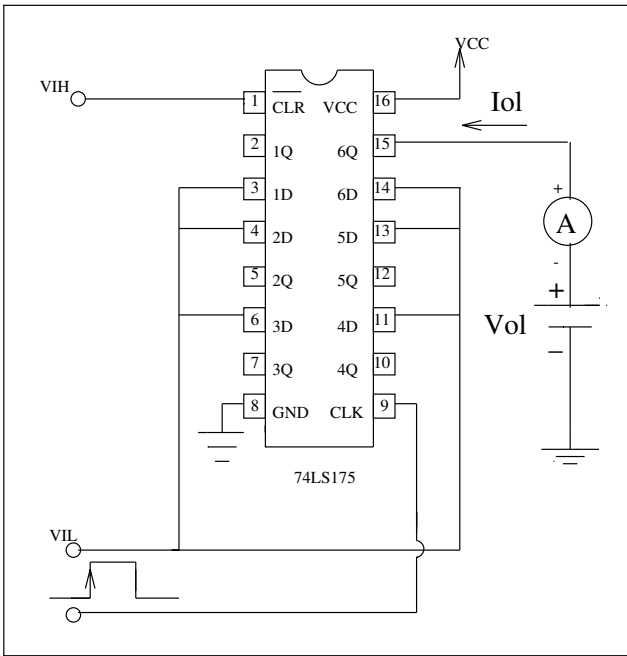


Figura 3.13 : Medição da corrente IOL.

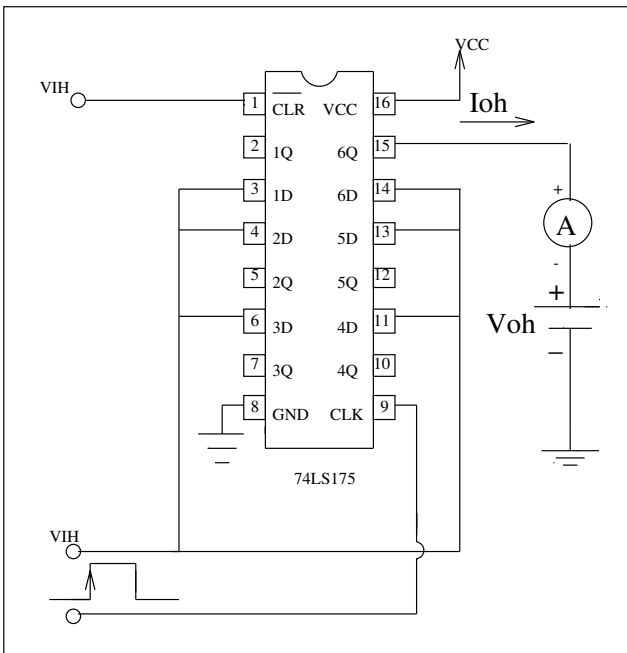


Figura 3.14 : Medição da corrente IOH.

3.3.11 Medição de IOH

Esta corrente geralmente não é medida nos testes elétricos mais comuns. Seu valor máximo, fornecido pelo manual do fabricante do componente, é utilizado como corrente de carga para a realização do teste funcional e para a medição de VOH. A corrente IOH deve ser medida na fase de caracterização, forçando-se uma tensão de VOH na saída do componente e medindo-se a corrente resultante. Obviamente, o *chip* deve ser levado à condição de saída de nível lógico alto, como mostra a figura 3.14.

3.4 PARAMÉTRICO AC

As medições paramétricas AC, ou o teste paramétrico AC, nome geralmente dado para o teste/medições de tempos, são medições mais complexas de serem realizadas que as medições paramétricas DC. Estão relacionadas com os tempos de comutação dos transistores internos ao circuito integrado. Requerem equipamentos mais precisos, necessitam de uma atenção especial na montagem da *interface* que conecta o componente ao testador e gastam um tempo grande de teste.

Geralmente estas medições são realizadas na fase de caracterização do circuito integrado. A finalidade destas medições é assegurar que mudanças de estados lógicos (*edges*) ocorram em um determinado espaço de tempo. A medição paramétrica AC pode precisamente quantificar os valores exatos destas mudanças.

Em um programa de teste para produção, os parâmetros AC não são medidos, pois estas medições são lentas, mas são verificados durante a execução do teste funcional.

Os principais parâmetros AC especificados para a operação funcional do dispositivo são:

- **tPHL** - Tempo de atraso de propagação do nível lógico alto para o nível lógico baixo.
- **tPLH** - Tempo de atraso de propagação do nível lógico baixo para o nível lógico alto.
- **tPHZ** - Tempo para desabilitar uma saída *Tri-state* do nível lógico alto .
- **tPLZ** - Tempo para desabilitar uma saída *Tri-state* do nível lógico baixo.

- **tPZH** - Tempo para habilitar uma saída *Tri-state* para o nível lógico alto.
- **tPZL** - Tempo para habilitar uma saída *Tri-state* para o nível lógico baixo.
- **tTLH** - Tempo de transição do nível lógico baixo para o nível lógico alto.
- **tTHL** - Tempo de transição do nível lógico alto para o nível lógico baixo.
- **tSU** - Tempo de *setup*.
- **tH** - Tempo de *hold*.
- **tW** - Tempo de duração do pulso.
- **fmax** - Frequência máxima

3.4.1 Evento

Um evento é definido como a ocorrência de uma transição de tensão através de um nível especificado, o LIMIAR (*threshold*), em uma dada direção (*slope*), como mostra a figura 3.15. Geralmente as medições de tempo utilizam como referência estes limiares de tensão.

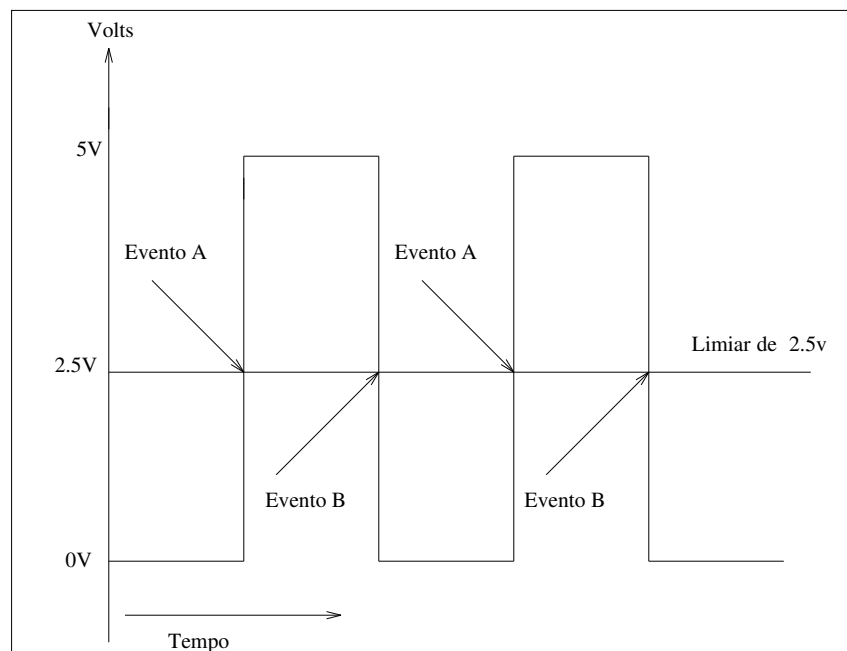


Figura 3.15 : Ocorrência de um evento.

3.4.2 Atraso de propagação

A tensão de saída de uma porta lógica nunca responde instantaneamente às variações de entrada. Há sempre um certo atraso associado à porta lógica. Este atraso é característica de qualquer sistema real e, no caso específico dos circuitos integrados, ele ocorre devido, principalmente, às diversas capacitâncias associadas aos componentes discretos contidos na pastilha. As medidas comuns do tempo de atraso de propagação são:

- Tempo de propagação entre dados de entrada e dados de saída. Para o caso de lógica combinacional e *buffers*.
- Tempo de propagação entre o sinal de habilitar saída (*enable*) e a alteração no sinal de saída. Para o caso dos circuitos com saídas *tri-state*.
- Tempo de propagação entre o sinal de relógio (*clock*) e a alteração no sinal de saída. Para o caso dos *flip-flops* e dos *latches*.

A figura 3.16 mostra as formas de onda de entrada e saída de uma porta inversora TTL, e os tempos de atrasos envolvidos.

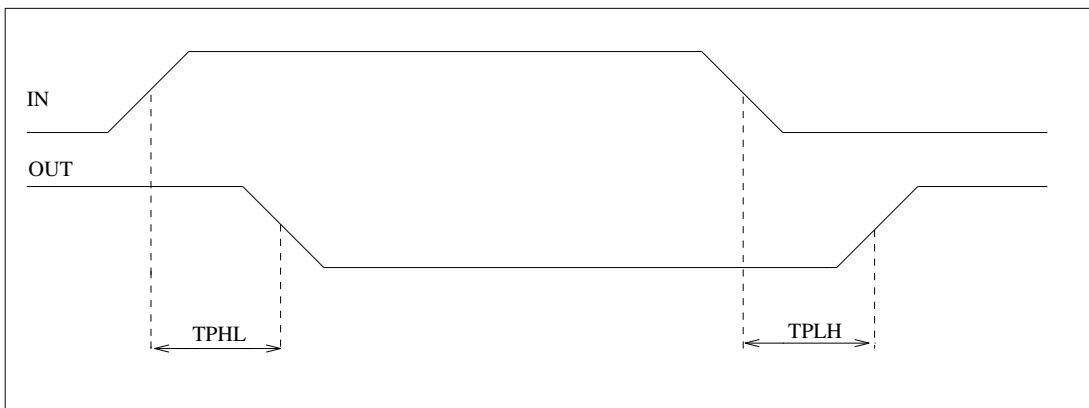


Figura 3.16 : Tempos de propagação t_{PHL} e t_{PLH} .

3.4.3 t_{PHL}

É o tempo de propagação do estado lógico “1” para o estado lógico “0”. Este tempo é medido entre pontos de referência de tensão especificados sobre as formas de ondas de entrada e de

saída, com a saída variando de um nível lógico alto definido para um nível lógico baixo definido.

3.4.4 tPLH

É o tempo de propagação do estado lógico “0” para o estado lógico “1”. Este tempo é medido entre pontos de referência de tensão especificados sobre as formas de ondas de entrada e de saída, com a saída variando de um nível lógico baixo definido para um nível lógico alto definido.

3.4.5 tPD

O tempo de atraso de propagação tPD é medido entre pontos específicos de referência nas formas de onda de entrada e de saída, com a saída variando de um nível lógico definido (alto ou baixo) para outro nível lógico definido (baixo ou alto).

O atraso de propagação é um fator que limita a aplicação de um CI, principalmente nas frequências mais elevadas. Se a frequência do sinal de entrada de um dado CI variar muito rápido, em um tempo menor que o atraso de propagação, a saída do circuito integrado não conseguirá responder às variações da entrada, ocasionando um erro no funcionamento do circuito.

O conhecimento dos tempos de atraso é muito importante no projeto prático do circuito. A fim de se evitar o aparecimento de pulsos espúrios indesejáveis, ou de ter o conhecimento de onde eles ocorrem deve-se fazer, pelo menos em esquemas complexos, o chamado “diagrama de tempo”, que consiste em se desenhar as formas de onda afetadas pelos diversos tempos de propagação.

3.4.6 tPHZ

É o tempo gasto para a saída transicionar do nível lógico “1” para o estado de alta impedância, isto é, o tempo gasto para desabilitar uma saída *tri-state* que estava anteriormente com nível lógico “1”. É medido entre pontos específicos de referência nas formas de onda de entrada (sinal de *enable*) e de saída, com a saída *tri-state* variando de um nível lógico “1” para o estado de alta impedância.

3.4.7 tPLZ

É o tempo gasto para a saída transicionar do nível lógico “0” para o estado de alta impedância, isto é, o tempo gasto para desabilitar uma saída *tri-state*, que estava anteriormente com nível lógico “0”. É medido entre pontos específicos de referência nas formas de onda de entrada (sinal de *enable*) e de saída, com a saída *tri-state* variando de um nível lógico “0” para o estado de alta impedância.

3.4.8 tPZH

É o tempo gasto para a saída transicionar do estado de alta impedância para o nível lógico “1”, isto é, o tempo gasto para habilitar uma saída *tri-state* para o nível lógico “1”. É medido entre pontos específicos de referência nas formas de onda de entrada (sinal de *enable*) e de saída, com a saída *tri-state* variando do estado de alta impedância para o nível lógico “1” .

3.4.9 tPZL

É o tempo gasto para a saída transicionar do estado de alta impedância para o nível lógico “0”, isto é, o tempo gasto para habilitar uma saída *tri-state* para o nível lógico “0”. É medido entre pontos específicos de referência nas formas de onda de entrada (sinal de *enable*) e de saída, com a saída *tri-state* variando do estado de alta impedância para o nível lógico “0” .

As medições dos tempos relacionados com *tri-state* são mais complexas de serem efetuadas que as medições dos tempos de atrasos de propagação, pois o nível de tensão na saída do componente no estado de alta impedância é indefinido. Como as medições de tempo são baseadas em níveis de referência de tensão para início e término da contagem do tempo, é necessário então conectar cargas dinâmicas programáveis de corrente em cada pino, como mostra a figura 3.17 .

Os testadores modernos possuem cargas dinâmicas programáveis. No entanto, para os testadores mais antigos, é necessário conectar um resistor de carga do pino sob teste à uma fonte de tensão programável. O resistor de carga pode ser conectado ou desconectado ao dispositivo sob teste, utilizando-se relés, como mostra a figura 3.18 . Praticamente todos os testadores fornecem sinais para acionamento de relés, que podem ser montados no *loadboard*.

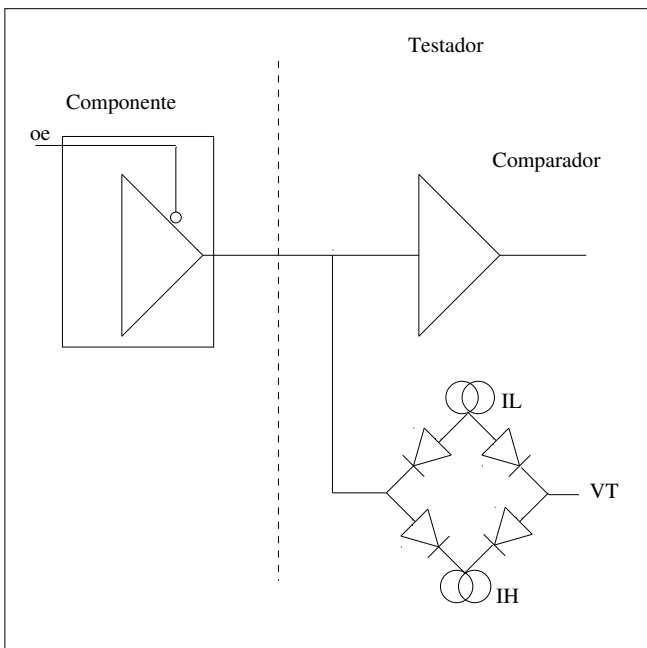


Figura 3.17 : Cargas dinâmicas do ATE.

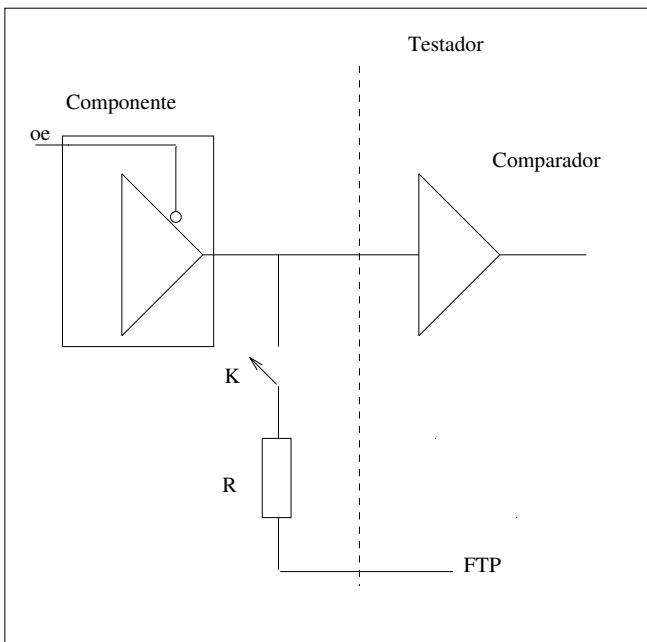


Figura 3.18 : Carga resistiva chaveada por relé.

V_t é uma tensão de limiar, programável. Quando a tensão de saída do com-

ponente for maior que V_t , a carga dinâmica absorve a corrente I_H . Quando o nível de tensão de saída for menor que V_t , a carga dinâmica irá fornecer a corrente I_L .

Quando a saída do componente estiver no estado de alta impedância, a carga levará o nível da tensão de saída para V_t . Este nível de V_t é programado para um nível de tensão intermediário entre as referências de tensão de saída alta e baixa.

A figura 3.19 mostra os tempos de propagação relacionados com a lógica *tri-state*.

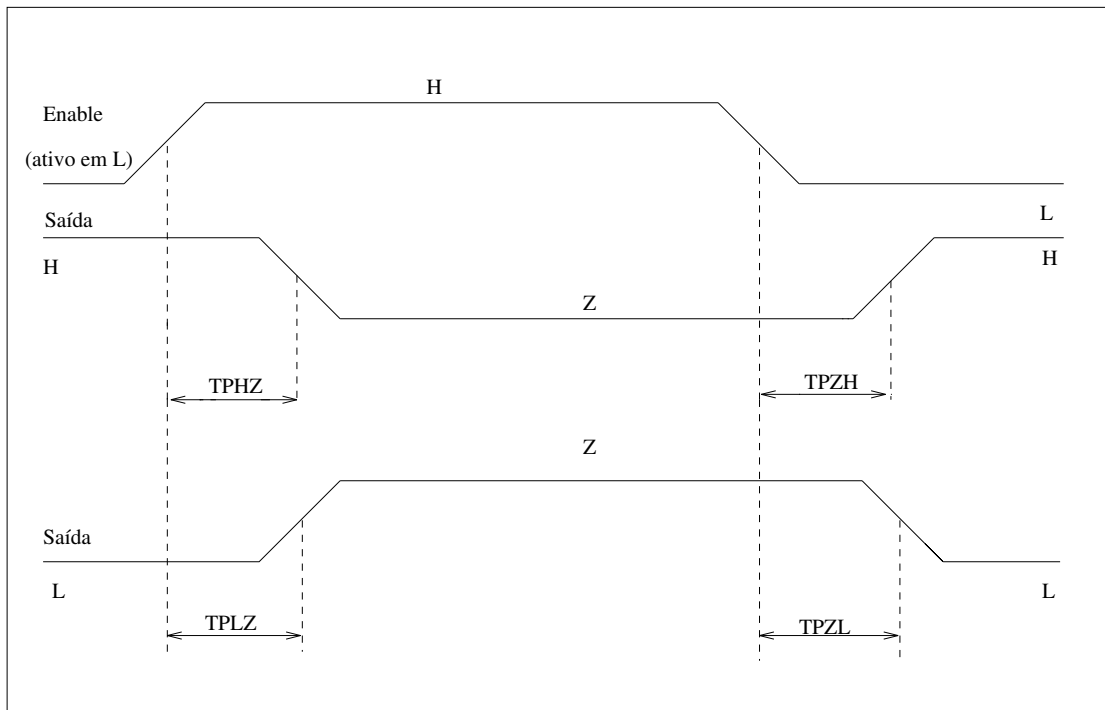


Figura 3.19 : Tempos de propagação $TPZL$, $TPLZ$, $TPZH$ e $TPHZ$.

3.4.10 Tempos de transição

Embora idealmente as transições de um nível lógico para outro sejam abruptas, na realidade estas transições ocorrem em um tempo diferente de zero. Este tempo é um fator muito importante na escolha do CI adequado para uma determinada situação, pois também está relacionado com a velocidade de chaveamento do mesmo. Para apresentação dos tempos de transição observe a figura 3.20 onde é mostrada uma onda típica digital, com a indicação dos tempos de subida e de descida .

3.4.11 t_{TLH}

É o tempo de transição do nível baixo para o nível alto. É medido entre um nível baixo de tensão e um nível alto de tensão especificados pelo fabricante do circuito de uma forma de onda que está transicionando do nível lógico “0” para o nível lógico “1”. Este tempo é também conhecido como tempo de subida (*rise time*). Geralmente é medido entre 10% de VDD e 90% de VDD.

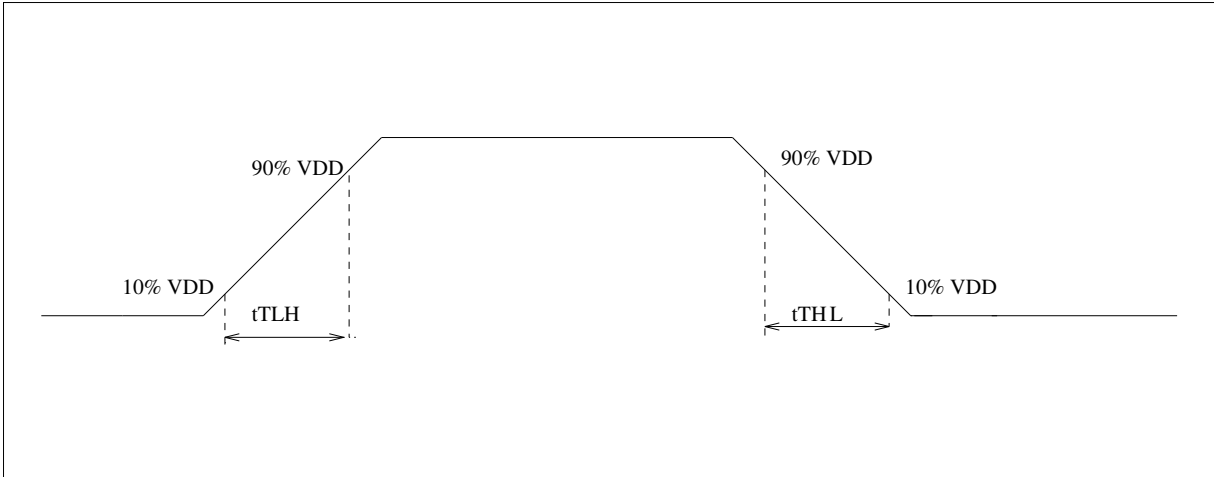


Figura 3.20: Tempos de transição t_{THL} e t_{TLH} .

3.4.12 t_{THL}

É o tempo de transição do nível alto para o nível baixo. É medido entre um nível de tensão alto especificado e um nível de tensão baixo especificado, de uma forma de onda que está transicionando do nível lógico “1” para nível lógico “0”. Este tempo também é conhecido como tempo de descida (*fall time*). Geralmente é medido entre 90% de VDD e 10% de VDD. O tempo de subida e o tempo de descida são determinados pela busca de dois níveis de tensão na borda da forma de onda, subtraindo-se os tempos encontrados.

3.4.13 Tempo de “setup” e de “hold”

As medições dos tempos de *setup* e de *hold* são mais difíceis de serem conceituadas e de serem implementadas. Sua dificuldade aumenta porque engenheiros de teste devem modificar padrões de teste funcional existentes para realizar a medição ou o teste. São medições realizadas por

aproximação sucessiva e/ou busca binária; por isso, são medições lentas e que devem ser efetuadas na fase de caracterização do *chip*. No teste de produção estes tempos são verificados no teste funcional, com base na especificação do manual do fabricante do componente.

3.4.14 Tempo de "setup"

É o intervalo de tempo que um dado nível lógico especificado de entrada do circuito integrado deve preceder ao sinal ativo aplicado em outra entrada, geralmente o sinal de relógio, para garantir o reconhecimento do nível lógico especificado de entrada pelo circuito integrado. A figura 3.21 ilustra os tempos de *setup* e de *hold*.

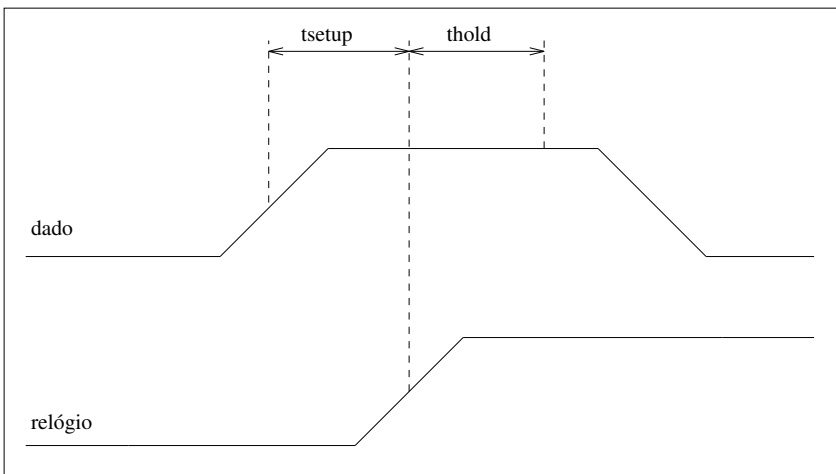


Figura 3.21: Tempos de *setup* e de *hold*.

3.4.15 Procedimento para medição do tempo de "setup"

- Criar um dado de entrada com um pulso centrado em torno do relógio. Assegurar uma grande margem de erro quando ajustar o tempo de *setup* inicialmente.
- Diminuir o tempo programado inicialmente para *setup*. Rodar o teste funcional. Este processo deve ser repetido até o componente falhar no teste funcional.
- Garantir que a falha do componente foi ocasionada pela falta de *setup*.

- O tempo de *setup* será o tempo definido como início da subida do relógio subtraído do tempo definido como início da subida do dado de entrada, quando ocorreu a falha.

A figura 3.22 ilustra o fluxograma para a medição do tempo de *setup*.

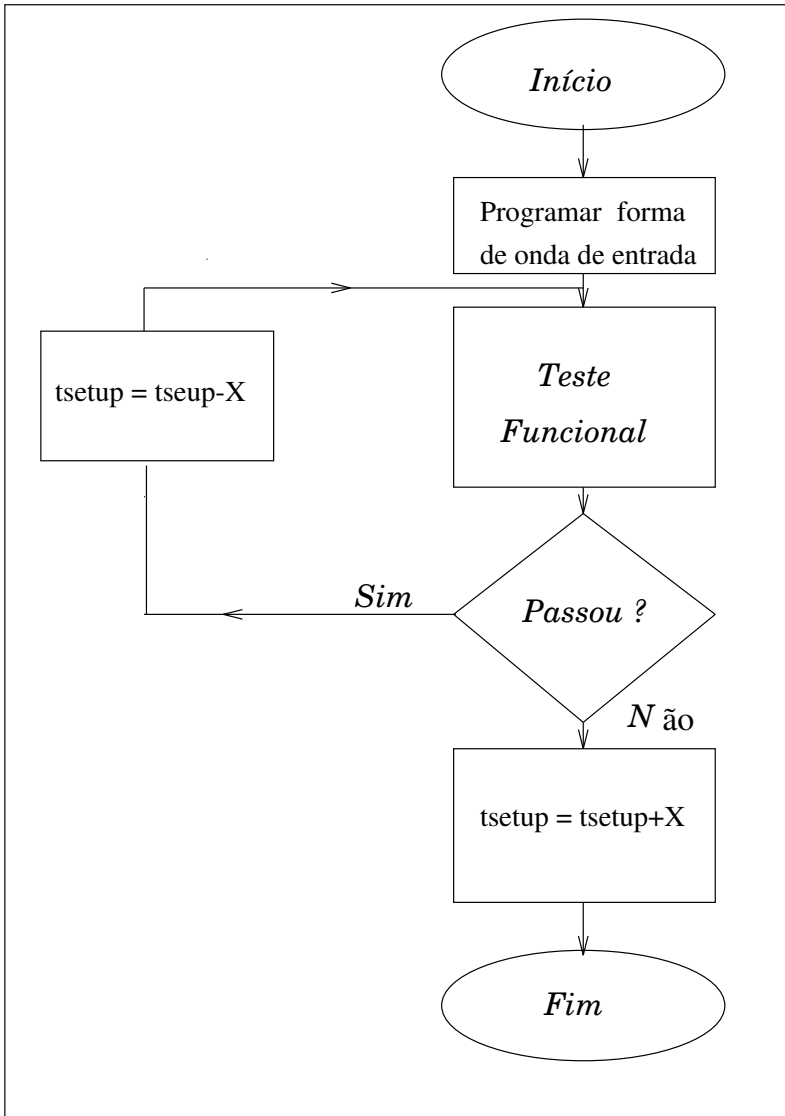


Figura 3.22: Fluxograma para medição do tempo de *setup*.

3.4.16 Tempo de "hold"

É o intervalo de tempo que um dado nível especificado de entrada deve permanecer em uma entrada do circuito integrado após ao sinal ativo ser aplicado em outra entrada, geralmente o

sinal de clock, para garantir o reconhecimento do nível especificado de entrada pelo circuito integrado.

3.4.17 Procedimento para medição do tempo de "hold"

- Criar um dado de entrada com um pulso centrado em torno do relógio. Assegurar uma grande margem de erro quando ajustar o tempo de *hold* inicialmente.
- Diminuir o tempo programado inicialmente para *hold* . Rodar o teste funcional. Este processo deve ser repetido até o componente falhar no teste funcional.
- Garantir que a falha do componente foi ocasionada pela falta de *hold* .
- O tempo de *hold* será o tempo definido como descida do dado de entrada subtraído do tempo de definido como início da subida do relógio, quando ocorreu a falha.

OBSERVAÇÕES :

- É possível, na prática, se encontrar tempos de *setup* e de *hold* negativos, positivos e nulo.
- Quanto menor for o valor em unidade de tempo especificado para X, das figuras 3.22 e 3.23, maior precisão terá a medição tanto do tempo de *setup* como do tempo de *hold*; porém, as medições se tornam mais lentas. Como o tempo de teste não é crítico na fase de caracterização, deve-se escolher um valor pequeno para o X.
- Geralmente os níveis de referência de tensão utilizados para as medições de *hold* e *setup* são especificados no manual do fabricante do componente.

A figura 3.23 ilustra o fluxograma para a medição do tempo de *hold* .

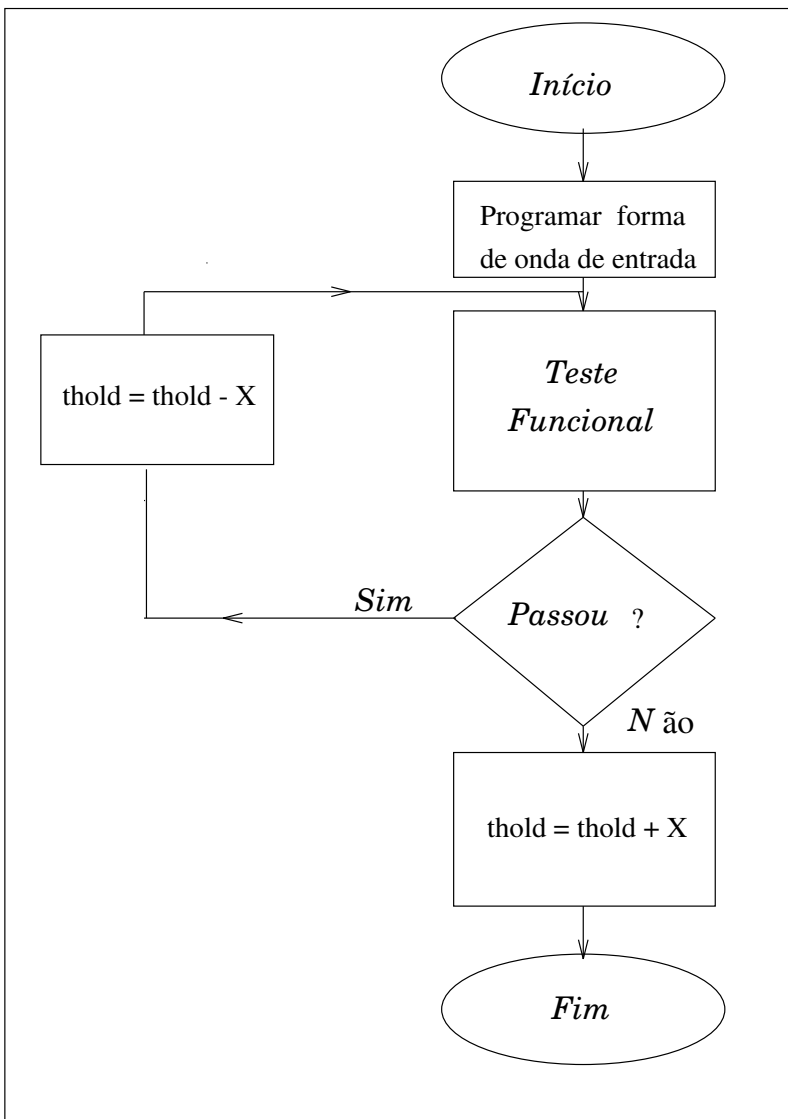


Figura 3.23: Fluxograma para medição do tempo de *hold* .

3.4.18 t_W

É o intervalo de tempo entre níveis de tensão especificado durante a subida e descida ou descida e subida do sinal de entrada aplicado ao circuito integrado ou do sinal de saída do CI. O tempo de duração do nível lógico “1”, t_{WH} , é medido entre níveis de referência de tensão durante a subida e descida do sinal. O tempo de duração do nível lógico “0”, t_{WL} é medido entre níveis de referência de tensão durante a descida e subida do sinal. A figura 3.24 ilustra t_{WH} e t_{WL} .

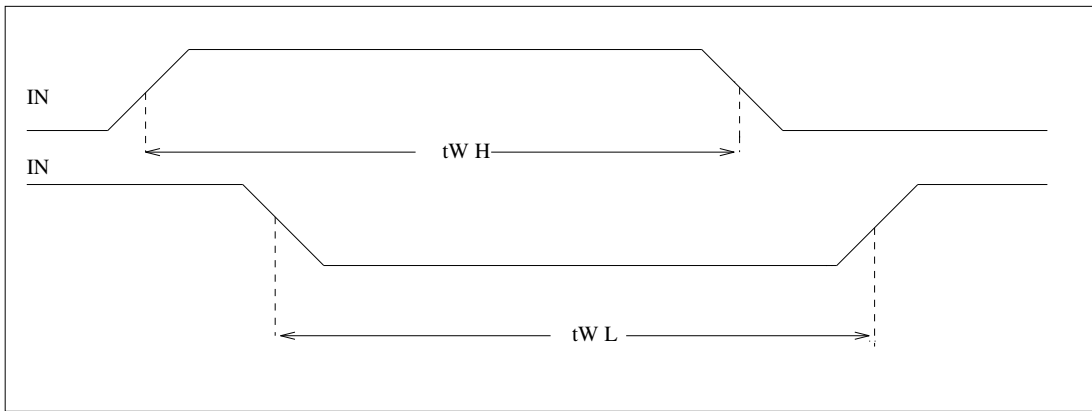


Figura 3.24: Largura do Pulso.

3.4.19 f_{max}

É a máxima frequência que o circuito integrado pode operar corretamente. Pode ser verificada, executando-se um teste funcional no equipamento automático de teste, com $clock = f_{max}$. Isto é, os vetores do padrão de teste excitarão o circuito sob teste nesta máxima frequência.

3.5 REFERÊNCIAS BIBLIOGRÁFICAS

- MILMAN, Jacob; HALKIAS, Christos C. *Eletrônica : Dispositivos e circuitos* ; tradutor Elédio José Robalinho; revisor técnico Paulo Elyot Meirelles Villela. - São Paulo: McGraw-Hill do Brasil, 1981.
- TERADYNE, *A500 Family of test systems, image base language manual*. June 1992.
- TERADYNE, *A500 Family of test systems, mixed-signal instrumentation manual* . January 1990.
- TERADYNE, *A500 Training introduction to the A500*. June 1990.
- TERADYNE, *A500 Training AC & DSP student workbook*. June 1990.
- ANDRADE, Luis Francisco P.J. *Parâmetros de caracterização, versão 1.0*. Documentação interna ao CPqD TELEBRÁS, DME, Seção de Aplicação, Agosto de 1990.

- RIBEIRO, Álvaro José Lopes. *Biblioteca de rotinas para medida de parâmetros de e ac* . Documentação interna ao CPqD TELEBRÁS, DME, Seção de Aplicação, 1993.
- CAMILO, Daniel; TADANOBY, João Baptista; YABU-UTI; YANO, Yuzo. *Circuitos lógicos: teoria e laboratório: engenharia eletrônica* . São Paulo: Liv. Ciência e Tecnologia, 1984.
- AZEVEDO JUNIOR, João Batista. *TTL/CMOS: teoria e aplicação em circuitos digitais* . São Paulo : Érica, 1984.
- PHILIPS Semiconductors-Signetics. Fast TTL logic series data handbook, 1992.
- TEXAS Instruments . TTL logic standard TTL, schottky, low-power schottky data book. 1988.
- NATIONAL Semiconductor. CMOS logic databook. 1988.
- NORMA IEC 748-2, primeira edição, 1985.
- BRYSON, Randy; JARVIS, Dale; PANKRETS, Steve. *Testing the new high-speed logic devices* . Electronics test. March, 1983.

Capítulo 4

ATE E LINGUAGENS DE PROGRAMAÇÃO

4.1 DEFINIÇÃO

Define-se equipamentos automáticos de teste como sistemas eletrônicos programáveis que controlam e monitoram circuitos integrados para determinar sua operacionalidade.

4.2 HISTÓRICO

Os computadores disponíveis no início dos anos 60 eram grandes e de propósito geral, possibilitando a conexão do testador como um de seus periféricos. A eletrônica dos equipamentos de teste era constituída basicamente de uma *interface* com o computador, vários registradores, os circuitos de *drivers* e de detectores. Os circuitos de controle de fontes de alimentação e circuitos para reportar falhas faziam parte dos modelos mais sofisticados.

Em 1965 foi produzido comercialmente pela Fairchild, o primeiro ATE para teste de circuitos integrados: o modelo Fairchild 4000 que era programado por cartão perfurado. Nos modelos posteriores foi incluído o controle direto pelo computador.

Companhias como a Teradyne, com seu modelo J259, e a Texas Instrument, com o modelo 553, iniciaram a comercialização dos ATE's na mesma época.

A importante transição que ocorreu no final da década de 60, incorporou o computador no equipamento automático de teste, permitindo arquiteturas e linguagens especializadas para controle do teste de circuitos integrados. Hoje, alguns tipos de ATE possuem mais de um computador, como o caso do Teradyne A500 que possui um computador para controlar a cabeça de teste e outro para controlar a *interface* homem/máquina e para gerenciar o acesso à rede local, além de possuir redes de processadores para realizar o processamento digital de sinais.

4.3 APLICAÇÃO DOS ATE'S

O equipamento automático de teste se faz necessário em diversas fases da vida do circuito integrado a saber:

- Depuração do protótipo

- Caracterização
- Produção
- Ensaios de confiabilidade
- Inspeção de recebimento
- Análise de falhas

4.4 CUSTO DO TESTADOR

Os ATE's são equipamentos complexos, com custo variando entre centenas de milhares e milhões de dólares, dependendo da aplicação e da complexidade do equipamento. Devido a sua complexidade e ao seu alto custo, necessita de pessoal especializado para:

- Programação
- Operação
- Manutenção preventiva
- Manutenção corretiva
- Manutenção de *software*
- Calibração e aferição
- Suporte para a rede local

4.5 ARQUITETURA BÁSICA DE UM ATE

Para fins didáticos, os equipamentos automáticos de teste estão divididos em cinco partes, como mostra a figura 4.0. A primeira parte é o bloco de entrada e saída, composto de teclado, monitor de vídeo, unidade de disco, unidade de fitas, impressora e *interface* para rede local. O segundo bloco é constituído pelo computador, que comanda o ATE, excitando o dispositivo sob teste e

processando as informações resultantes. Alguns ATE's mais sofisticados possuem mais de um computador. O próximo bloco é a memória do testador, que é composta pela memória de programa, onde o programa de teste executável é armazenado e a memória de padrão, que armazena os vetores de teste. O próximo bloco definido como eletrônica central, engloba as fontes de tensão programáveis, os voltímetros, os amperímetros, os conversores A/D e D/A, os geradores, os processadores digitais de sinais, analisadores lógicos, etc. Todos esses equipamentos estão disponíveis para um pino programado do circuito integrado digital sob teste.

Na cabeça de teste, último bloco da figura 4.0, estão os canais digitais, compostos pelos circuitos de *driver*, para excitar o DUT, e os detectores para verificar a resposta do circuito sob teste. É na cabeça de teste que vai o *Load board*, cuja função é conectar os canais do testador, as fontes de tensão, os medidores com a placa de *interface* do dispositivo - DIB.

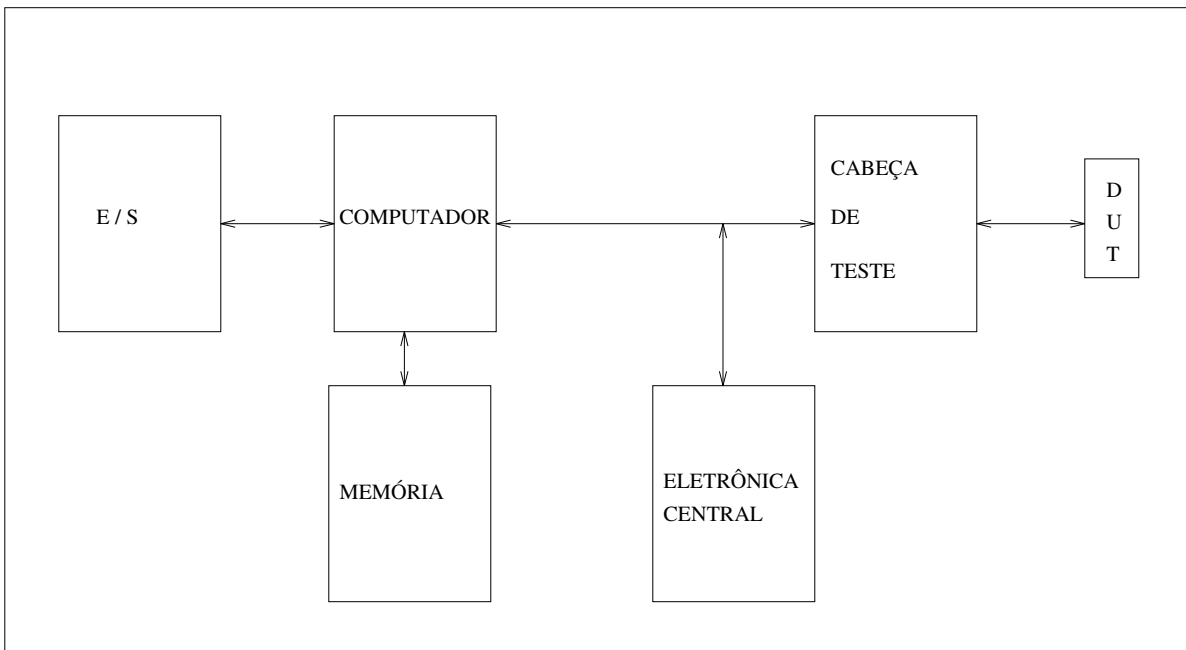


Figura 4.0: Arquitetura básica de um ATE

4.6 J325 - SISTEMA AUTOMÁTICO DE TESTES DE CIRCUITOS INTEGRADOS DIGITAIS

4.6.1 Introdução

O sistema J325 pode testar automaticamente circuitos integrados eletrônicos. Pode ser acoplado a um *Handler*, para teste automatizado de grandes quantidades de componentes; a um *Wafer Probe*, para testes de *Wafer* e para testes de componentes manualmente colocados na cabeça de teste. É um sistema de teste totalmente controlado por computador.

A linguagem usada para a comunicação do homem com o sistema é o Pascal T, uma linguagem de alto nível da Teradyne baseada em Pascal, para programar sistemas de teste.

O J325 é um sistema de teste de circuitos integrados da década de 70, portanto bem antigo, mas bem versátil, apresentando, assim, limitações para teste de circuitos de alta frequência, e para as medidas das correntes de fuga de um *chip* CMOS, que é da ordem de grandeza de picoampère.

Uma das maiores deficiências do J325 é a *interface* homem-máquina bastante pobre, comparada com a dos testadores automáticos dos dias de hoje, principalmente em relação ao seu editor de programas que é bem limitado, devido a idade do sistema.

Outro problema com o J325 é a sua manutenção, que devido à falta de peças e falta de pessoal treinado para a manutenção do equipamento, fica difícil de ser realizada e cara, sendo muitas vezes necessária a importação de placas para a reposição.

O sistema é bom e adequado para o teste de CI's com baixa e média escala de integração. A figura 4.1 mostra o diagrama de blocos do J325.

4.6.2 Terminal de vídeo

É utilizado para realizar a *interface* homem/máquina. É composto por um teclado, onde são digitados comandos e editados programas de teste e por um monitor, onde são apresentadas as respostas do testador, o resultados das medidas, a listagem dos programas, etc.

4.6.3 Unidade de fitas

O J325 utiliza fitas magnéticas para armazenar tanto os *softwares* para a sua operação tais como o Editor de Programas, o Compilador, o Sistema Supervisor, o Update e o MOP, quanto para armazenar os programas desenvolvidos para testes de componentes. A fita utilizada é “3M DC 300A *data cartridge*”. Esta fita possui 4 trilhas para armazenamento de dados e/ou programas.

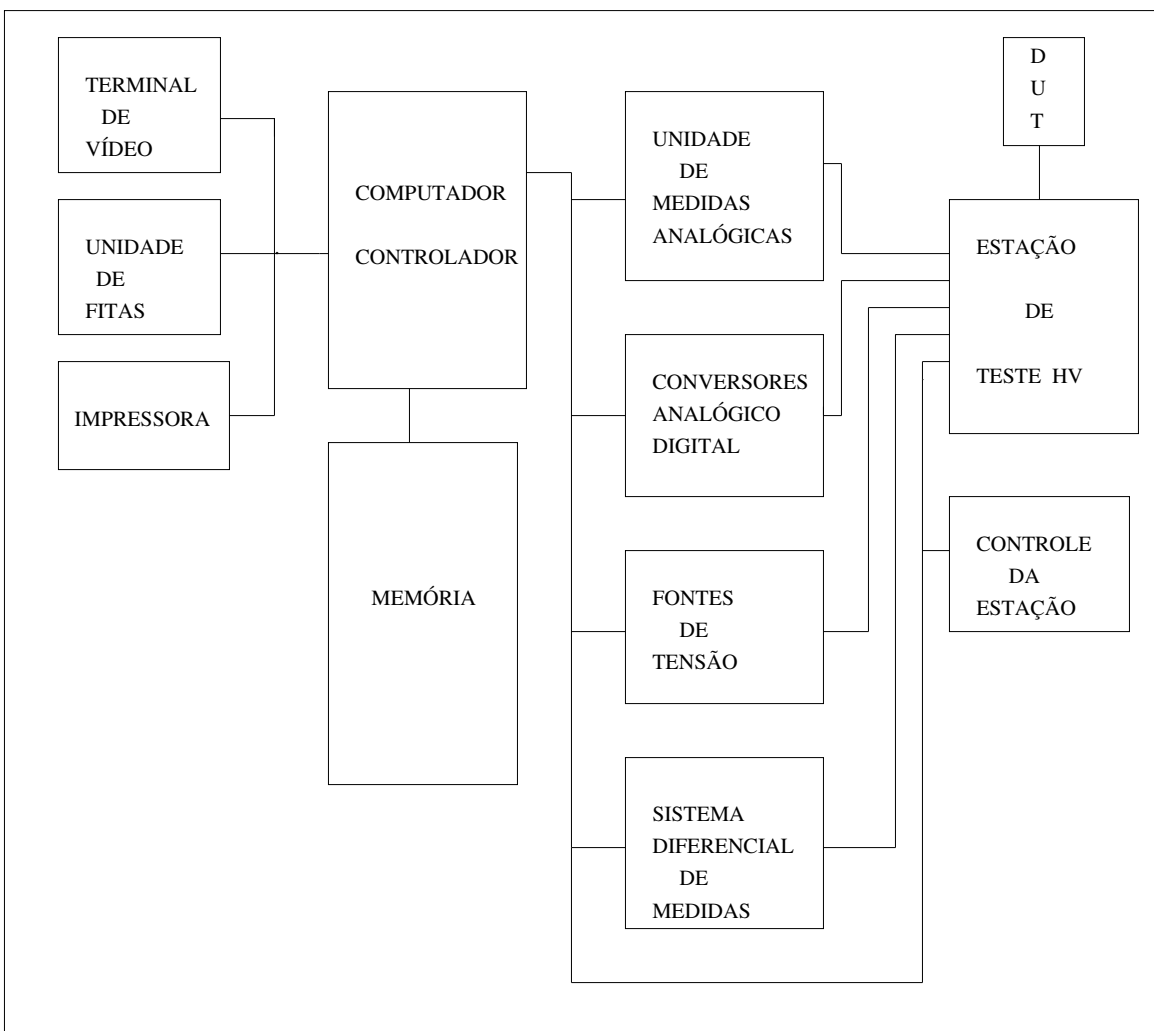


Figura 4.1: Diagrama de blocos do J325.

4.6.4 Impressora

A máquina necessita de uma impressora matricial tipo paralela, para impressão dos resultados (*datalog*) dos testes realizados, do sumário, da listagem dos programas de testes, etc.

4.6.5 Computador

O sistema de teste é totalmente controlado por um computador. Neste ATE mais antigo existe apenas um computador para controlar a estação de teste e a *interface* homem/máquina. Este computador traduz a linguagem objeto do programa de teste e executa os comandos para a realização de medições de corrente e de tensão, para execução do teste funcional , etc. Além disso, realiza , sob controle do Sistema Operacional, o comando de acesso à fita, ao vídeo e à impressora. Traduz para a linguagem de máquina o programa de teste escrito em Pascal T, por intermédio de um compilador.

Este computador, anterior ao microprocessador INTEL 8080, utiliza palavras de 18 *bits* , e pode endereçar até 256 K palavras de memória. A unidade lógica e aritmética, os registros e o acumulador, o contador de programa, etc , hoje blocos de um único *chip* microprocessador, no J325 são circuitos montados em cartões com *chips* TTL.

4.6.6 Memória

O J325 possui 64K *bytes* de memória com palavras de 18 *bits* . A memória é organizada em páginas de 4K *bytes* e pode ser expandida até 256K *bytes* . O computador do J325 utiliza memórias de estado sólido para armazenamento de dados. A memória é uma RAM (memória de acesso aleatório) dinâmica, que utiliza a capacitância de *gate* de um dispositivo MOS como meio de armazenamento. Este CI é o TM116 com capacidade de armazenamento de 16.384 X 1 *bit* . O cartão pode armazenar até 16 k palavras de 19 *bits* , sendo que 18 *bits* são usados para dados e 1 *bit* para a paridade.

4.6.7 Unidade de medidas analógicas (AMU)

Os terminais de entrada e saída do dispositivo sob teste normalmente são conectados aos *drivers* e detectores do sistema para a execução de teste funcional. Estes mesmos terminais podem ser desconectados dos *drivers* e detectores e serem conectados à unidade de medidas analógicas, AMU. A AMU pode ser programada para forçar uma corrente enquanto mede uma tensão ou forçar uma tensão para medir uma corrente.

A estação de teste utiliza relés para chavear entre *drivers*, detectores e a unidade de medida. Tensões e correntes de entrada e saída são medidas desconectando o terminal do dispositivo do *driver* ou detector e conectando-o à AMU.

Para medir uma corrente, a AMU é programada para forçar uma tensão e medir a corrente. A amplitude desta tensão é estabelecida selecionando uma das 3 faixas: $\pm 2V$ com resolução de 1mV, $\pm 20V$, com resolução de 10mV e $\pm 100V$ com resolução de 100mV.

A amplitude da tensão é determinada por uma palavra digital de 12 *bits*, transmitida pelo computador. Este valor é convertido para analógico, por um conversor digital/analógico e enviado para a AMU.

A corrente medida é convertida em um valor digital equivalente por conversores analógico/digital de alta velocidade. A corrente medida está em uma das seis faixas: $2\mu A$, $20\mu A$, $200\mu A$, $2mA$, $20mA$ e $200mA$, com as respectivas resoluções $1nA$, $10nA$, $100nA$, $1\mu A$, $10\mu A$ e $100\mu A$. O valor digital da corrente medida é transmitida ao computador, onde é comparada com os limites programados no teste.

Para medir uma tensão, a AMU é programada para forçar corrente e medir tensão. A corrente que será forçada é estabelecida pela seleção do valor da corrente e da respectiva faixa, que é a mesma utilizado para a medição de corrente. O valor da corrente que será forçada é dado por uma palavra digital de 12 *bits* transmitida pelo computador. Na AMU este valor digital é convertido em uma corrente analógica.

A tensão no terminal do *DUT* é medida por um circuito de medição de tensão, usando uma das mesmas 3 faixas de tensão já citados. A tensão é convertida em um valor digital por um conversor analógico-digital. O valor digital é transmitido para o computador, onde é comparado com os limites do programa de teste. A AMU inclui *clamp* de tensão ou seja, circuitos grampeadores de tensão.

O programador pode especificar a máxima tensão que será aplicada aos terminais do dispositivo. Sem isso, o dispositivo pode ser danificado, particularmente se existir um mau contacto no *handler* ou na provadora.

4.6.8 Conversores analógicos digitais e digitais analógicos

Os conversores D/A são utilizados para converter os sinais de comandos digitais de 12 *bits* do computador em tensões analógicas para os *drivers*, para as fontes de tensão, para as tensões de referência dos detectores, correntes para as medidas de tensão, tensão para as medidas de correntes e tempo.

Os conversores A/D são utilizados no sistema, principalmente para converter os resultados das medições de corrente e de tensão para digital, onde serão processadas pelo computador.

4.6.9 Fontes de tensão para o dispositivo

O J325 possui três fontes de alimentação programáveis de VCC: VCC1, VCC2 e VCC4, utilizados para alimentar os componentes sob teste; ainda tem-se o VCC3 como expansão. As fontes de alimentação do J325 podem ser programadas para fornecer tensões de -10V a +20V. Existem duas faixas de tensão para cada fonte: uma de -10V a +20V com resolução de 10mV e outra de -2.047V a +2.047V com resolução de 1mV. A máxima corrente que VCC pode fornecer é de $\pm 1.6A$. A linha de terra (GND) não é programável. Tanto VCC e GND são conectados ao DUT através do *Load Board*.

4.6.10 Sistema de medidas diferenciais

Para as medições de tensão mais precisas, necessárias para a verificação das especificações dos circuitos integrados CMOS, é mais preciso realizar as medições diferencialmente, comparando-as com a alimentação do dispositivo. O J325 possui uma unidade para este fim. Tensões muito pequenas entre a saída do dispositivo e o terminal de alimentação, podem ser medidas com 100uV de resolução. Para medir uma tensão diferencial, o programador especifica o terminal a ser medido e a fonte de tensão a qual a medição será diferencialmente comparada. O resultado é convertido para digital e transmitido ao computador para ser comparado com os limites do teste.

4.6.11 Estações de teste

São disponíveis duas estações diferentes de testes como parte do sistema J325:

- HV, que é uma estação otimizada para teste de circuitos integrados CMOS e TTL
- LV (Não disponível no testador que estávamos utilizando) para testes de circuitos integrados ECL e TTL. Não será comentado neste trabalho.

4.6.12 Características da estação HV

- Otimizada para dispositivos CMOS
- Usada para teste envolvendo tensões relativamente altas e baixas correntes
- *Drivers* : Com nível de tensão variando de - 5V até + 20V
- Detector : Resolução de medida de 10mV
- Chaveamento de Relé para conexões de baixa impedância
- Resolução de 1nA do Sistema de Medida
- Medidas diferenciais com resolução de $100\mu V$
- Dois relógios programáveis
- *Strobe* no detector para teste de parâmetros de tempo

Cada estação de teste possui a eletrônica que realiza a *interface* do testador com o dispositivo sob teste (*device under test - DUT*). Cada pino do dispositivo é conectado a um soquete que interliga a entrada, a saída e a alimentação do DUT aos circuitos na estação de teste. Algumas conexões vão diretamente para as fontes de alimentação, enquanto que outras vão para circuitos de *driver* e de *detector*, ou para a AMU.

4.6.13 A estação HV do J325

A estação de teste de Alta Tensão HV705, figura 4.2, é otimizada para testar dispositivos CMOS. Para isto a estação possui:

- 12 *drivers* de entrada, 46 opcionais;
- 12 detectores de saída, 46 opcionais;
- 2 relógios programáveis;
- 1 *strobe* programável para o detector;
- 1 circuito de teste de continuidade por canal.

4.6.14 ”Drivers” e detectores

O Sistema possui um par de *drivers* A e um par de *drivers* B por canal, usados para excitar o componente sob teste, denominados AHI e BHI para o nível lógico 1 e ALO e BLO para o nível lógico 0 . Para se testar dispositivos simples, todas as entradas são normalmente conectadas no *driver* A. No entanto, para componentes complexos, pode ser necessário conectar as entradas no *driver* A para alguns testes e no *driver* B para outros.

Cada canal também possui um par de comparadores, COMHI, para comparar o nível lógico 1 e COMLO, para o nível lógico 0. Estes comparadores devem ser conectados aos pinos de saída do DUT. As tensões de saída do componente são comparadas com valores pré-estabelecidos no programa e o sistema decide, de acordo com a sua programação, se o componente está bom ou não. A figura 4.3 mostra os *drivers* e os detectores ou comparadores do J325.

4.6.15 ”Load board”

A função do *Load Board* é conectar todos os canais do testador ao *DIB, Device Interface Board* . Existem para o J325 dois tipos de *Load Board: Partitioned e Non Partitioned* , que serão comentados mais adiante. Os programas de teste deste trabalho foram desenvolvidos para o *Load Board Non Partitioned* (TC 396).

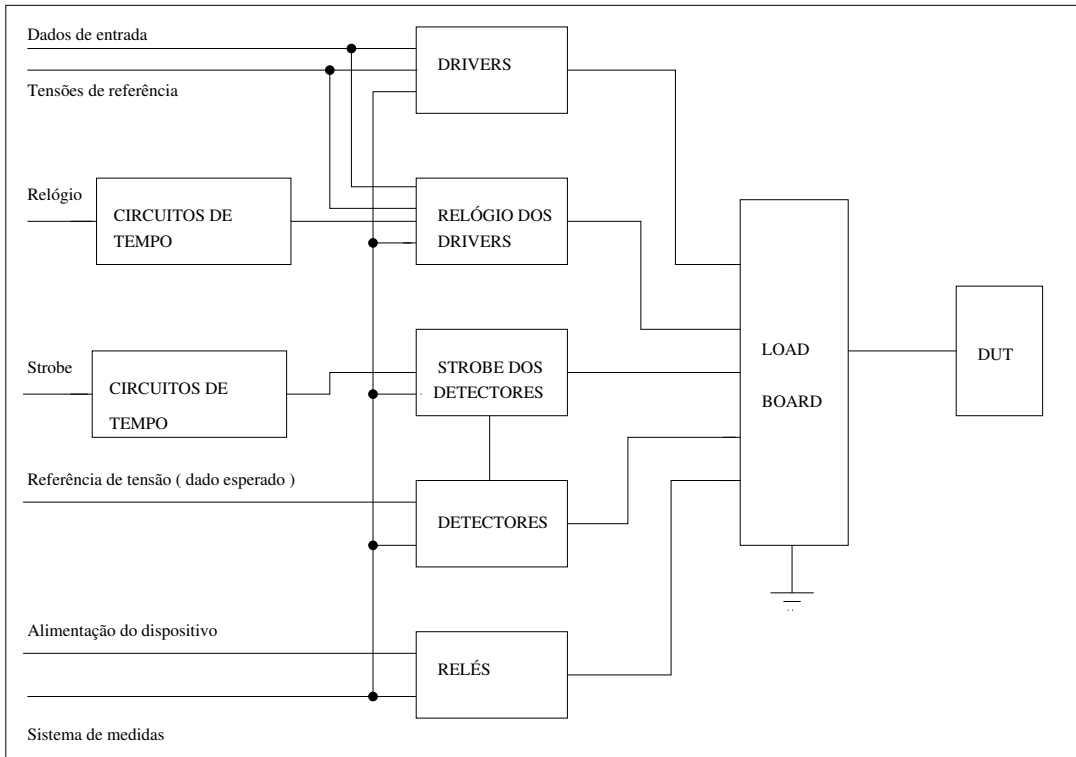


Figura 4.2 : Estação de teste HV.

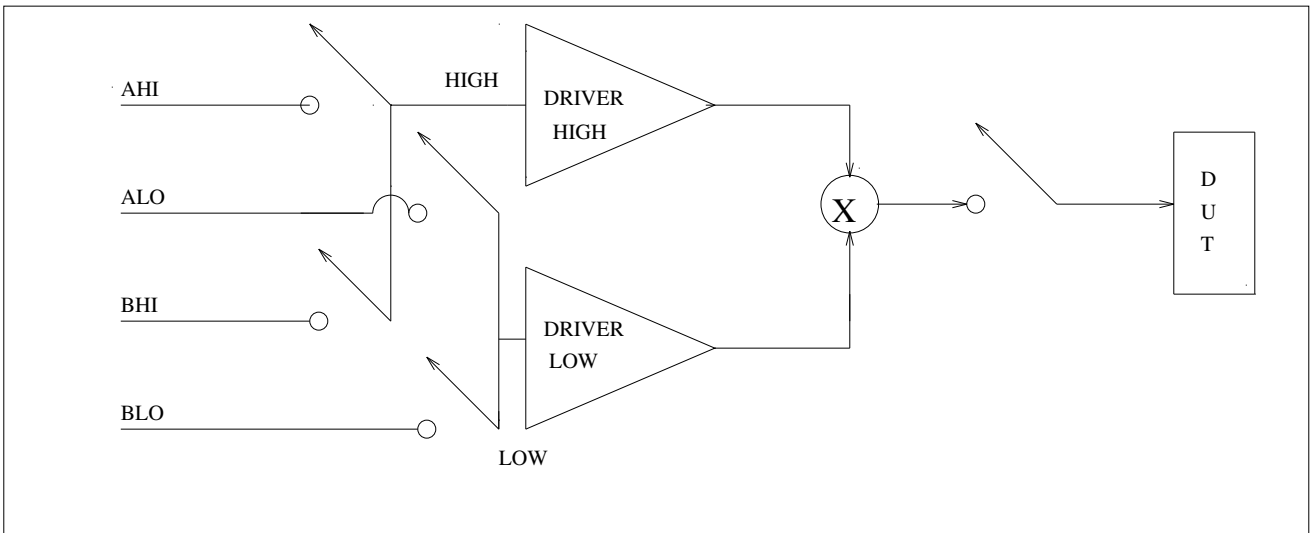


Figura 4.3 : Drivers e detectores do J325.

4.6.16 DIB - "Device interface board"

A função do *DIB* é conectar o *LOAD BOARD* ao *DUT* (Dispositivo sob Teste), isto é, conectar os terminais do dispositivo que está sendo testado aos canais do sistema. São placas de circuito impresso, com pinos no lado de baixo que são conectados nos soquetes do *Load Board*. Pode-se montar *DIB's* com soquetes para diversos tipos de CI's. Foram montados no laboratório da Seção de Ensaios de Confiabilidade da TELEBRÁS vários tipos de *DIB* (plaquinha), uma para cada família de componente, para teste de CI's TTL com 14 pinos, para CI's com 16 pinos, para 20, 24 e 28 pinos. Para teste de componentes CMOS os "*DIB's*" são diferentes, porque o terra não é conectado ao terra do sistema e sim a uma fonte de tensão programada com 0V, para se evitar problemas de flutuação.

4.6.17 Circuitos de tempo

Quando se testa dispositivos semicondutores é extremamente importante que os tempos sejam reproduzidos com precisão. Estes tempos são :

- O atraso e a largura do pulso de relógio.
- O tempo no qual o dado de saída é comparado com a referência (Strobe).

Em muitos casos os dispositivos são separados em categorias, de acordo com os parâmetros de tempo. Sistemas de teste utilizados por fabricantes de circuitos integrados devem ser capazes de reproduzir tempos com uma precisão suficiente para diferenciar uma categoria de outra. Também os sistemas de teste para inspeção de recebimento devem ser capazes de reproduzir tempos precisos. Para isto, o J325 possui dois canais de *drivers* especiais, chamados de relógio. Cada pulso de relógio aplicado aos terminais do dispositivo semicondutor que está sendo testado possui um atraso e uma duração específica. O dado de saída do DUT é comparado com o dado esperado, em um tempo específico. Os dois sinais de relógio e o *strobe* de cada comparador ou detector possuem seus tempos comandados pelo programa de teste e são produzidos por geradores de pulso no sistema. A figura 4.4 mostra esses tempos. A figura 4.5 ilustra os circuitos envolvidos na geração dos pulsos de relógios 1 e 2 e o *strobe* dos detectores.

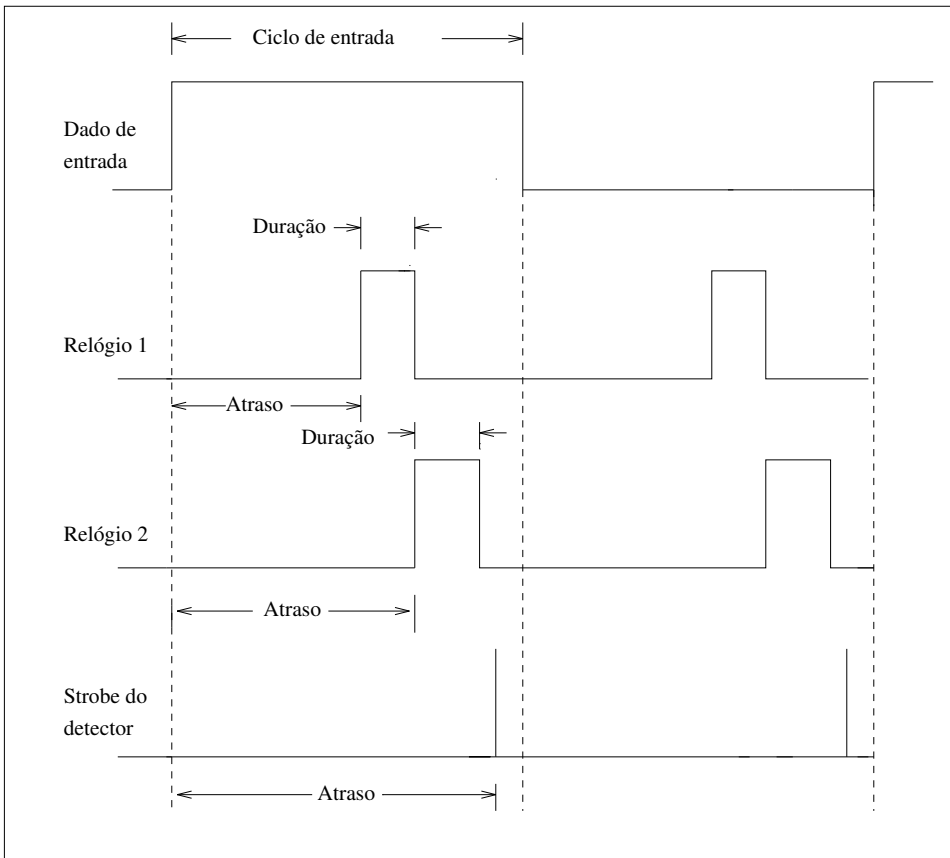


Figura 4.4: Os sinais de relógios e o strobe.

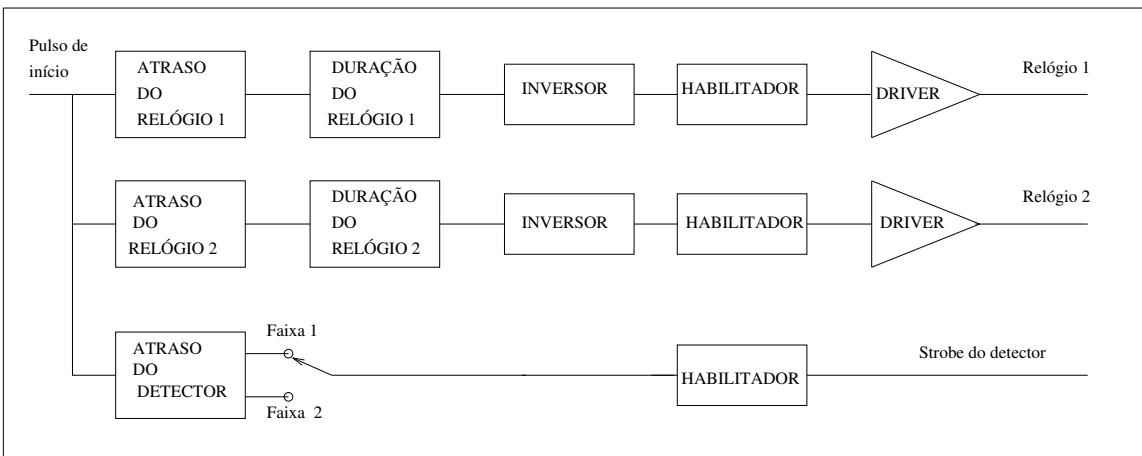


Figura 4.5: Os circuitos envolvidos na geração dos relógios e do strobe.

4.6.18 Modos de operação

O J325 possui dois modos de operação: *PARTITIONED* e *NON PARTITIONED*. A seleção do modo é feita pelo tipo de *Load Board* .

Modo separado (*partitioned*)

Neste modo de operação, os *drivers* são conectados em um terminal do componente e os comparadores conectados em outro terminal do componente. Assim, deve-se tomar um cuidado especial para se testar componentes que possuem em um único terminal, entrada e saída de sinais.

O *Load Board Partitioned* utiliza 24 dos 48 canais disponíveis de drivers/detectores e separa-os em 24 canais de drivers e 24 canais detectores. Os 24 canais restantes não podem ser divididos e podem ser usados ou como *driver* , ou como detector, ou como *driver/detector* . Neste modo separado tem -se então, 72 canais disponíveis. Os canais neste modo são distribuídos da seguinte maneira :

PALAVRA	CANAIS	TIPO
W1	1 ao 12	ENTRADA
W2	13 ao 24	SAÍDA
W3	25 ao 36	ENTRADA (DE W2)
W4	37 ao 48	SAÍDA (DE W1)
W5	49 ao 60	ENTRADA/SAÍDA
W6	61 ao 72	ENTRADA/SAÍDA
CLOCK 1	36	
CLOCK 2	35	

TIPO:

ENTRADA - Ligar na entrada do DUT

SAÍDA - Ligar na saída do DUT

Modo não separado (non partitioned)

Neste modo de operação um par de *drivers* e um par de comparadores são conectados em um terminal do componente sob teste, o que facilita o teste de componentes que possuam terminais comuns de entrada e saída. Usando o *Load Board Non Partitioned*, os 48 canais de *driver/detector* podem ser usados para qualquer combinação de entrada e saída. Os canais de *drivers* e de comparadores do J325 neste modo estão distribuídos da seguinte maneira:

PALAVRA	CANAIS	TIPO
W1	1 a 12	ENTRADA/SAÍDA
W2	13 a 24	ENTRADA/SAÍDA
W5	49 a 60	ENTRADA/SAÍDA
W6	61 a 72	ENTRADA/SAÍDA
CLOCK 1	24	
CLOCK 2	23	

4.6.19 Cargas externas

Quando o dispositivo sob teste necessitar de cargas externas, por exemplo, para teste de CI's coletor aberto, as resistências de carga devem ser colocadas no *Load Board* e conectadas ao componente através de relés, acionados via *software*.

4.7 Linguagem de Programação dos ATE

Para que o teste de um componente possa ser realizado de maneira automática, é necessário que um programa de teste, escrito em uma linguagem específica do testador, assuma o comando do equipamento, efetuando as medições previamente programadas.

A linguagem dos ATE's geralmente são linguagens de alto nível, baseadas em uma linguagem comercial como o Pascal ou o C. Além das instruções existentes nestas linguagens, são incluídas novas instruções, para comandar a instrumentação, as fontes de tensão e de corrente, os medidores de corrente e de tensão, o modo como o resultado do teste será mostrado, a interligação dos canais do testador com o circuito sob teste, etc. O fabricante do testador desenvolve um compilador

para a sua linguagem.

Serão descritos, a seguir, os principais comandos da linguagem utilizada no testador J325.

4.7.1 A linguagem de programação Pascal-T

Pascal-T é uma linguagem de alto nível, desenvolvida pela Teradyne, para programar sistemas automáticos de teste. Os programas podem ser escritos rapidamente e com precisão, pois as condições de teste são expressas com poucas instruções, baseadas na língua inglesa. Nesta linguagem os programas são estruturados, podendo ser lidos e entendidos com maior facilidade.

A estruturação de um programa é importante, pois a manutenção dos programas de testes freqüentemente não é realizada pelo autor. Um programa de teste completo pode ser escrito com as instruções do Pascal-T. Se houver necessidade, pode-se utilizar de instruções da linguagem *assembly*, para um controle mais fino das funções do sistema.

4.7.2 Estrutura do Programa

Um programa de teste escrito em Pascal-T é dividido em quatro partes :

- Comentários
- Declaração
- Ação
- Sub-rotinas

Comentários

Os comentários de um programa de teste em Pascal-T, geralmente a primeira página do programa, devem conter as seguintes informações :

- Nome do componente a ser testado
- Nome do programador e data na qual o programa foi escrito

- Lista de testes desempenhados pelo programa

A linguagem possui dois modos de inserir comentários, o primeiro é inserindo `//`, tudo que vier após este símbolo é comentário. O segundo é a palavra `COMMENT`, usado quando várias linhas de comentários serão escritas.

Declaração

As declarações são colocadas geralmente no início do programa. Elas dão informações ao compilador que o possibilita de interpretar as variáveis na seção de ação do programa. Uma variável pode ter no máximo 6 caracteres. As variáveis do Pascal-T podem ser de 5 tipos :

- Inteira (INT)
- Caracter (CHAR)
- Uma função do sistema de teste (SYS)
- Um *array* de inteiros (ARRAY OF INT)
- Um *array* de caracteres (ARRAY OF CHAR)

Uma variável é escrita da seguinte forma :

VAR (nome da variável):(tipo)

Como exemplo , um programa pode incluir as variáveis V1 e V2 para representar tensões e A1, A2 para representar caracteres que serão impressos. A declaração destas variáveis é :

VAR V1,V2 : SYS

VAR A1, A2 :CHAR

Ação

A parte de ação de um programa escrito na linguagem Pascal-T contém instruções ou procedimentos que estabelecem :

- As condições da instrumentação (tensões, correntes, estados lógicos)

- As condições de *Bin* (classificar o componente como bom ou ruim, de acordo com os testes, que passou ou falhou)
- Os valores limites (os valores máximos e mínimos para as tensões e correntes medidas)
- O teste (iniciar as rotinas que realizam os testes)
- Processamentos (dos valores dos parâmetros medidos e dos valores dos parâmetros que serão utilizados no teste)
- Decisões (uma ação subsequente do programa dependendo do resultado do teste)
- Comunicações de entrada e saída (ler dados/comandos do teclado e da fita magnética , e transmitir dados para o vídeo, para a fita e para a impressora)

Sub-rotinas

Quando uma seqüência de comandos for executada várias vezes em um programa de teste, esta seqüência deve ser escrita em forma de sub-rotina, e no programa principal deve ser incluído chamadas à sub-rotina, o que deixa o programa mais estruturado. O comando para acessar a sub-rotina é *CALL NOME* (parâmetros), onde *NOME* é nome da *PROCEDURE* e os parâmetros são as variáveis que serão passadas para a sub-rotina. O comando de término da sub-rotina é *END*.

4.7.3 Definições de canais

É simples e bastante útil no programa de teste escrever uma tabela de definição dos canais, que relaciona o pino do componente com o canal do sistema de teste. Esta tabela é chamada de *pinmap*, na linguagem do testador Teradyne A500. Para cada pino é dado um nome, similar ao nome do manual. Um exemplo de uma tabela de definição de canais está ilustrado abaixo:

INA1=CH 1 // PINO 1

INB1=CH 2 // PINO 2

OUT1=CH 13 // PINO 3

OUT2=CH 14 // PINO 4

```

INA2=CH 3 // PINO 5
INB2=CH 4 // PINO 6
INA3=CH 5 // PINO 8
INB3=CH 6 // PINO 9
OUT3=CH 15 // PINO 10
OUT4=CH 16 // PINO 11
INA4=CH 7 // PINO 12
INB4=CH 8 // PINO 13

```

Para os canais, as seguintes definições, que reúnem um grupo de canais em uma macro, podem ser feitas, o que facilita bastante o programa. O exemplo abaixo ilustra definições comumente usadas em um programa de teste.

```

OUTS=OUT1 OUT2 OUT3 OUT4
INA=INA1 INA2 INA3 INA4
INB=INB1 INB2 INB3 INB4
INS=INA INB

```

4.7.4 Teste de continuidade

Este teste é realizado para verificar que todos os pinos do dispositivo estão fazendo contacto com o *load board* e não há curto-circuitos ou circuitos abertos. A estação HV do J325 , possui instruções para realizar este teste. O exemplo abaixo ilustra o teste.

```

ECK INS OUTS //Conectar todas as saídas e entradas no circuito de continuidade
PAUSE 1MS // Pausa para estabilização
FTEST // Realiza o teste
EK0 // Desconecta o circuito de teste

```

4.7.5 Níveis de tensão das fontes

Os comandos de um programa em Pascal-T são escritos em letra maiúscula. As tensões e as correntes das quatro fontes de alimentação VS1, VS2, VS3 e VS4 são programadas como mostra os comandos abaixo:

VS1 5V CLAMP E3 // VS1 programada com 5V no *range* de 20.47V, e corrente de 20.0mA

VS2 -2V CLAMP E2 // VS2 programada com -2V no *range* de 2.047V, e corrente de 200mA

Na operação padrão o sistema opera na menor faixa de corrente, ou seja 2mA.

As fontes de tensão podem ser programadas por expressões como :

VS1= 2V + 3V

ou

VS1 = VCCmín

Para conectar a fonte no componente o comando é :

VCON 1 // Conectar a fonte VS1

VCON 1 2 // Conectar as fontes VS1 e VS2

VCON 0 // Desconectar todas as fontes

4.7.6 Níveis de tensão dos "drivers"

Os níveis de tensão dos *drivers* são programados da seguinte forma :

ADDR HI 2.4V LO 0.4V // Níveis do do *driver* A, 2.4V para o alto e 0.4V para o baixo.

BDDR HI 2.4V LO 0.4V // *Driver* B, 2.4V para o alto e 0.4V para o baixo.

Comentou-se acima, como programar os níveis de tensão dos *drivers* . Para conectá-los aos pinos do componente sob teste o comando é o seguinte :

CONN ADDR INPUTS

onde a macro INPUTS foi previamente definida como os pinos de entradas do dispositivo sob teste

CONN BDDR B1,B2,B3

Para os detectores segue-se o mesmo raciocínio, como abaixo:

DET HI VOHMIN LO VOLMAX // Detectores esperam 2.0V para nível alto e 0.4V para o nível baixo.

Os detectores estão programados para comparar a saída do componente com os valores acima programados. Se o nível alto do componente for maior ou igual a VOHMIN o componente passa, caso contrário será considerado como falha. Se o nível baixo do componente for menor ou igual a VOLMAX o componente passa, caso contrário será considerado como falha. Para conectar os detectores nos pinos do componente sob teste o comando é :

```
SELECT DET OUTPUTS
```

onde OUTPUTS são os pinos de saída do dispositivo sob teste. Para se desconectar os terminais do dispositivo dos *drivers* ou desabilitá-los dos detectores, usa-se os seguintes comandos:

```
CONN ADDR 0
```

```
CONN BDDR 0
```

```
SELECT DET 0
```

4.7.7 Medidas de tensões e de correntes

As medições de correntes e de tensões são programadas do seguinte modo :

- Conectar um ou mais pinos do dispositivo à AMU
- Ajustar as condições de cada pino do dispositivo
- Programar a unidade de medidas analógicas para forçar uma corrente ou uma tensão e colocar os limites do teste
- Efetuar a medida.

O exemplo abaixo mostra a medida de corrente do pino INA1:

```
CONN AMU INA1 // Conecta o pino INA1 do dispositivo à AMU
```

```
FORCE 5V // Forçar a tensão de 5V
```

```
MLIMIT LO 200UA HI 3MA// Definir os limites do teste.
```

MTEST // Iniciar a medição.

Como um outro exemplo temos a medida de tensão do pino INA1. Supondo que ele já tenha sido conectado à AMU, pela instrução acima.

FORCE 6MA // Forçar a corrente de 6mA no pino INA1 do dispositivo

MLIMIT LO 0V HI 400MV // Limitar a tensão medida entre 0 e 400mV

MTEST // Efetuar a medida.

Existem instruções mais poderosas que MTEST, como o AMTEST, STTEST,e o VTEST. O MTEST testa um canal. A instrução AMTEST testa vários canais, seguindo a ordem crescente de canais. Nesta instrução, os canais são conectados automaticamente à AMU. A instrução STTEST é semelhante a AMTEST só que a seqüência de canais é pré-definida nas DEFINIÇÕES MÚLTIPLAS do programa de teste. A instrução VTEST realiza as medições no modo diferencial.

4.7.8 Programando o teste funcional

Para escrever um programa para desempenhar um teste funcional em um dispositivo o programador deve definir :

- Tensões aplicadas aos terminais de VDD.
- Níveis lógicos alto e baixo do *driver*, que serão aplicados aos terminais de entrada do componente.
- Níveis de tensões alto e baixo do comparador que serão comparadas com a saída do componente.
- Cargas conectadas aos terminais de saída do componente
- Atraso de propagação necessário para o DUT alterar os níveis de tensão.

A seqüência de condições de entrada e níveis de saída esperados são comandadas pelos códigos do HILO. A seguinte sintaxe é usada:

CONN HILO

CH1 CH2 CH3 ...CHN T

.....

END

Os canais de 1 a 72 podem ser usados. Geralmente eles são definidos por nome. Aí vale a regra, os canais que forem citados na linha são forçados para AHI ou para BHI. Os que não forem serão forçados para ALO ou para BLO. O T aparece na linha onde combinações de estados de entrada e a respectiva condição de saída deve ser verificada, ou seja, o T habilita o *strobe* do comparador. Como exemplo temos um teste funcional de um CI NAND com 4 portas de duas entradas.

CONN HILO

INPUT T // todas as entradas em 1, saídas em 0

INAS OUTS T // entradas A alto, entradas B baixo, saídas alto

INBS OUTS T // entradas A baixo, entradas B alto, saídas alto

OUTS T // todas as entradas baixo, saídas alto

END

As macros INPUT, INAS, INBS e OUTS foram definidas anteriormente para representar apropriadamente os pinos de entradas e de saídas do dispositivo.

4.7.9 Temporização

Conforme comentado anteriormente, o J325 possui dois relógios e um *strobe* programáveis. Nos sinais de relógios, é possível programar o instante em que o pulso de relógio será iniciado e a sua duração. Nos testadores modernos todos os canais de *drivers* possuem esta facilidade. O instante em que o comparador será habilitado a realizar a leitura também deve ser programado. O comando para programar os relógios e o *strobe* é o TSEND. O exemplo abaixo ilustra essa programação. Os tempos são definidos geralmente em ns.

TSEND

WIDTH // Programar largura do pulso de relógio

W1 120. // Relógio 1 com largura de 120ns

W2 220. // Relógio 2 com largura de 220ns

DELAY // Programar atrasos

W1 435. // Relógio 1 com atraso de 435ns

W2 -10. // Relógio 2 com atraso de - 10ns

W5 410. E // *Strobe* com atraso de 410ns

Se não existir o ponto depois do número, este será compilado como octal. A última linha do comando TSEND deve ser terminada com o símbolo E.

4.8 MACROS

Um programa de teste pode, freqüentemente, ser simplificado utilizando macros para substituir comandos. Tomando como exemplo, o comando que define os níveis HI e LO do *driver A* :

ADDR HI 5V LO 0V

Pode ser substituído pela macro VIN.

A definição da macro é da forma :

VIN = ADDR HI 5V LO 0V

Após a macro ser definida, os níveis do *driver A* podem ser programados da seguinte forma :

VIN

o que simplifica o programa. Uma macro pode ser escrita com argumentos (parâmetros) que podem ser passados quando a macro é chamada em qualquer ponto do programa. Este tipo de macro é definido como abaixo

VIN = ADDR HI ?1 LO ?2 .

Depois da definição da macro, os níveis do *driver A* podem ser programados da seguinte forma :

VIN[5V,0V].

Durante a compilação, o primeiro valor que está entre colchetes 5V substituirá o ?1 e o segundo 0V substituirá o argumento ?2.

4.9 REFERÊNCIAS BIBLIOGRÁFICAS

- CORTNER, J. Max. *Digital test engineering* . " Wiley-Interscience ", 1987.
- TERADYNE, *J325 Maintenance system manual*. December 1982.
- TERADYNE, *J325 Programming class* . July 1984.
- TERADYNE, *J325 Programming Manual*. April 1983.

Capítulo 5

PROGRAMA DE TESTE - APLICAÇÕES

5.1 INTRODUÇÃO

Neste capítulo será apresentado um programa de teste para o componente 74HCT32. O equipamento automático de teste é o J325 da Teradyne e a linguagem de programação é o Pascal T. O componente escolhido para este trabalho é simples, o que facilita o entendimento do programa.

Este programa utiliza três macros : uma para programar o *strobe* do detector, uma para a numeração automática dos testes e a última para a medição das correntes de consumo, ICC. O teste funcional é realizado para três tensões de alimentação por uma sub-rotina ou *procedure*.

Tanto nas macros como nas sub-rotinas, está exemplificado o conceito de passagem de parâmetros.

Neste capítulo também será comentado o resultado deste programa (*datalog*) em 3 (três) componentes 74HCT32, numerados de 1 a 3 e o resultado de um teste de 6 (seis) componentes 62256. Os componentes numerados de 21 a 25 estão bons e o dispositivo 90 apresenta falha em todos os testes funcionais. Este componente é uma memória RAM estática , de 32 Kbytes e seu programa está no Anexo I.

Este programa utiliza instruções mais poderosas e recursos da linguagem *assembly* do J325. No Anexo I estão também os *datalogs* das medições realizadas.

5.2 PROGRAMA DE TESTE

O programa apresentado realiza um teste de continuidade, a fim de verificar a conexão do *Loadboard* com o dispositivo sob teste. Depois realiza um teste funcional com a VCC mínimo, VCC nominal e VCC máximo e mede as tensões de VOH e VOL para cada saída do componente, para VCC mínimo e máximo.

Os níveis de tensão de VIHmín e VILmáx são aplicados às entradas do componente como nível lógico “1” e nível lógico “0”, respectivamente. Com estes níveis de tensão de entrada o teste funcional é realizado.

As correntes de *Leakage* IIL e IIH são medidas.

O programa termina com três medidas diferentes da corrente de consumo, especi-

ficadas no manual do fabricante do componente. A figura 5.0 mostra o fluxograma do programa.

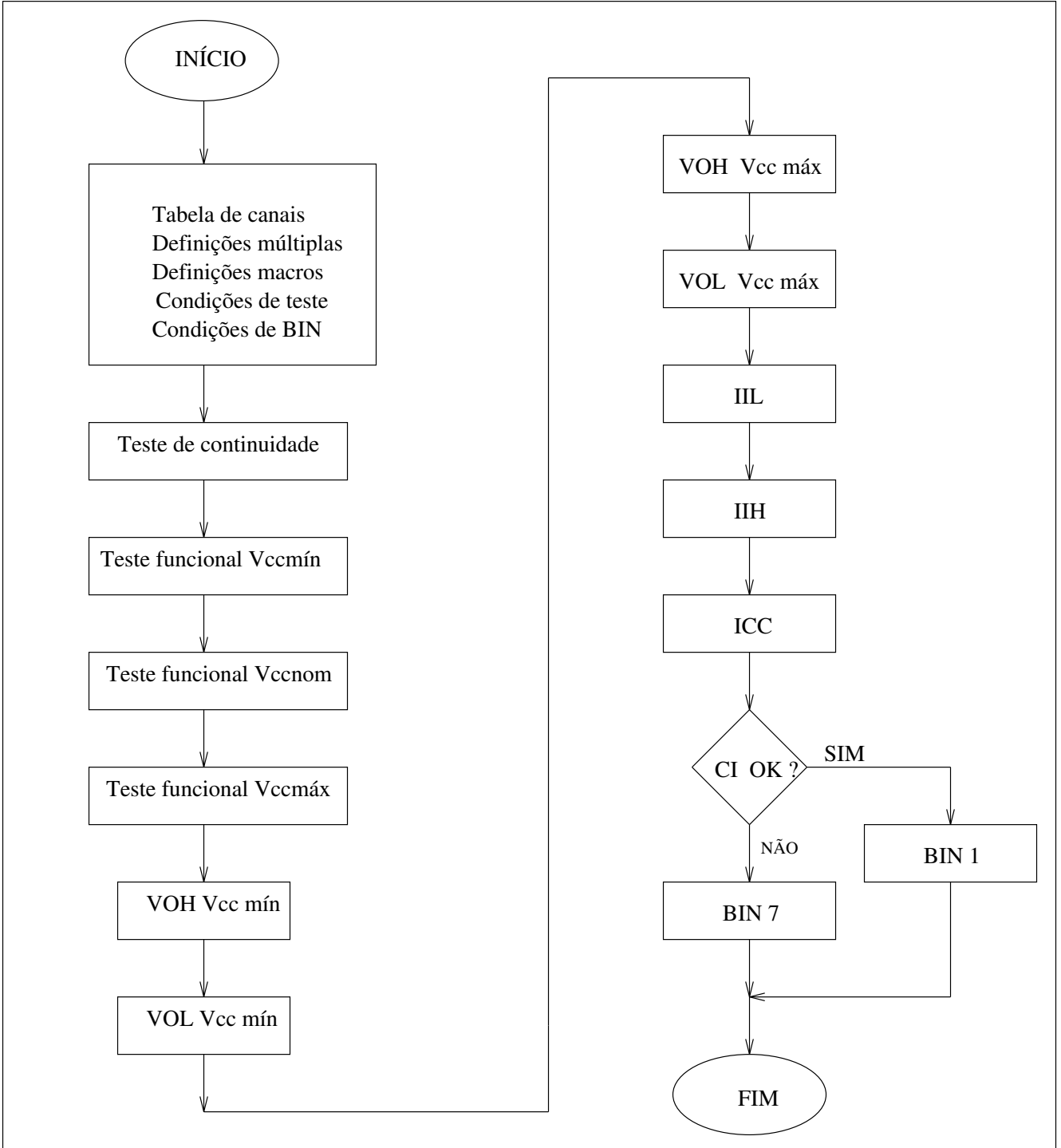


Figura 5.0: Fluxograma do programa de teste 74HCT32.

COMMENT

T E L E B R Á S - C P Q D

DEPARTAMENTO : REDE EXTERNA E SUPORTE TECNOLÓGICO - DRS

SEÇÃO : TECNOLOGIA DA QUALIDADE E DE EMPACOTAMENTO

APLICAÇÃO : INSPEÇÃO DE RECEBIMENTO E ENSAIOS DE QUALIFICAÇÃO

CÓDIGO COMERCIAL : 74HCT32

DESCRIÇÃO : 4 PORTAS OU DE DUAS ENTRADAS

CONDIÇÕES AMBIENTAIS : 25 GRAUS CELSIUS

BINNING

1.....APROVADO

7.....REJEITADO

12.....FALHA DE CONTINUIDADE

15.....SYSTEM ALARM

SWAP BLOCK.....829-068 - DIP 14-01

LOAD BOARD.....TC 396

PROGRAMADOR : ROGÉRIO LARA LEITE

DATA : 21/DEZ/93

TESTE N.....PARÂMETRO (CONDIÇÃO DE MEDIDA)

1.....CONTINUIDADE ENTRADAS E SAÍDAS

25.....FUNCIONAL (VCC=4.5V, VIH MÍN, VIL MÁX)

26.....FUNCIONAL (VCC=5.0V, VIH MÍN, VIL MÁX)

27.....FUNCIONAL (VCC=5.5V, VIH MÍN, VIL MÁX)

3X.....VOH <=VOH4 (VCC=4.5V, IOH4)

4X.....VOL <=VOL4 (VCC=4.5V, IOL4)

6X.....VOH <=VOH5 (VCC=5.5V, IOH5)

7X.....VOL <=VOL5 (VCC=5.5V, IOU TL5)

8X.....IIH <=IIH MÁX (VCC=5.5V, VIN=VCC)

9X.....IIL <=IIL MÁX (VCC=5.5V, VIN=0.0v)

100.....ICC <=ICCMÍN (VCC=5.5V, VIN=VCC)

101.....ICC <=ICCMÍN (VCC=5.5V, VIN=VGND)

11X.....ICC <=ICCMÁX (VCC=5.5V, VIN=2.4V POR PINO)

12X.....ICC <=ICCMÁX (VCC=5.5V, VIN=0.5V POR PINO)

CLAMP HV // UTILIZAR A ESTAÇÃO DE HV

// DEFINIÇÃO DE ALIMENTAÇÃO : VCC = VCC1 = PINO 14

// TERRA = GND = PINO 7

// TABELA DE DEFINIÇÃO DOS CANAIS

INA1= CH 8 // PINO 1

INA2= CH 9 // PINO 2

OUTA= CH 10 // PINO 3

INB1= CH 1 // PINO 4

INB2= CH 11 // PINO 5

OUTB= CH 12 // PINO 6

OUTC= CH 2 // PINO 8

INC1= CH 3 // PINO 9

INC2= CH 4 // PINO 10

OUTD= CH 5 // PINO 11

IND1= CH 6 // PINO 12

IND2= CH 7 // PINO 13

USING 1 TO 12 // CANAIS UTILIZADOS PELO SISTEMA

// DEFINIÇÕES MÚLTIPLAS

IN1= INA1 INB1 INC1 IND1

IN2= INA2 INB2 INC2 IND2

INPUT= IN1 IN2

OUT= OUTA OUTB OUTC OUTD

// DEFINIÇÕES MACRO

STROBE =// MACRO PARA PROGRAMAR O STROBE DO DETECTOR

TSEND

127 // PROGRAMAR DELAY

W5 ?1 E // RECEBER TEMPO EM NS

= // TERMINAR A MACRO

TESTN= // MACRO PARA PROGRAMAR A NUMERAÇÃO DOS TESTES

LOGP // PREPARAR DATALOG

CTN ?1 // RECEBER NÚMERO DO TESTE

= // TERMINAR A MACRO

MICC= // MACRO PARA MEDIDA DE ICC

CONN BDDR // LIMPAR O DRIVER B

CONN ADDR INPUT-?1 // TODAS ENTRADAS - A QUE FOI PASSADA EM A

CONN BDDR ?1 // E ENTRADA QUE FOI PASSADA EM B

MTEST // TESTAR

= // TERMINAR A MACRO

// CONDIÇÕES DE TESTE

VCCMÁX= 5.50V

VCCNOM= 5.00V

VCCMÍN= 4.50V

RANGE= E3

VMÁX= 5.50V

VGND= 0.0V

VIHMÍN=2.0V

VILMÁX= 0.8V

VOH4= 3.98V

VOH5= 4.98V

VOL4= 260MV

VOL5= 260MV

VINMÁX= 2.40V

VINMÍN= 0.50V

IIHMÁX= 100NA

IILMÁX= -100NA

IOH4= -4.00MA

IOH5= -4.80MA

IOL4= 4.00MA

IOL5= 4.80MA

ICCMÍN= E6 2.0UA

ICCMÁX= 1200.0UA

CATS 5 // NÚMERO DE FALHAS CATASTRÓFICAS

COUNTER 130 // NÚMERO MÁXIMO DE TESTES

ALO 100MV // DRIVER A

RBW 12 // BIN DE FALHA DE CONTINUIDADE

//-----CONTINUIDADE ENTRADAS E SAÍDAS-----

T1, CONN ECK INPUT OUT// CONECTAR ENTRADAS E SAÍDAS

PAUSE 1MS // PAUSA PARA ESTABILIZAÇÃO

FTEST // TESTAR

CONN ECK // DESCONECTAR O CIRCUITO DE CONTINUIDADE

LOGP // DATALOG

BW 1 2 3 4 5 6 // BIN PASSAGEM

PAUSE 1.0MS // PAUSA PARA ESTABILIZAÇÃO

//-----FUNCIONAL-----

T25, TESTN[25] // MACRO DE N. DE TESTE

STROBE[500.] // MACRO DO STROBE

ADDR HI VIHMIN LO VILMAX // PROGRAMAR OS DRIVERS

DET HI VOH4 LO VOL4 // PROGRAMAR OS DETECTORES

CONN ADDR INPUT // CONECTAR AS ENTRADAS

SELECT DET OUT // CONECTAR AS SAÍDAS

CALL FUNC (VCCMÍN) // TESTE FUNCIONAL COM VCC MÍNIMO


```

CALL FUNC (VCNOM) // TESTE FUNCIONAL COM VCC NOMINAL
CALL FUNC (VCCMÁX) // TESTE FUNCIONAL COM VCC MÁXIMO
//—————VOH4—————
T30, TESTN[30] // MACRO DE N. DE TESTE
SELECT DET // DESCONECTAR AS SAÍDAS
VS1 VCCMÍN CLAMP RANGE // PROGRAMAR FONTE PARA VCC MÍNIMO
FORCE IOH4 LO VOH4 // FORÇAR IOH E PROGRAMAR O LIMITE DE TENSÃO
CONN HILO INPUT END // ENTRADAS EM NÍVEL LÓGICO “1”
STEST OUT // MEDIR VOH NAS SAÍDAS
//—————VOL4—————
T40, TESTN[40] // MACRO DE N. DE TESTE
FORCE IOL4 HI VOL4 // FORÇAR IOL E PROGRAMAR O LIMITE DE TENSÃO
CONN HILO END // ENTRADAS EM NÍVEL LÓGICO “0”
STEST OUT // MEDIR VOL NAS SAÍDAS
//—————VOH5—————
T60, TESTN[60] // MACRO DE N. DE TESTE
SELECT DET // DESCONECTAR AS SAÍDAS
VS1 VCCMÁX CLAMP RANGE // PROGRAMAR FONTE PARA VCC MÁXIMO
FORCE IOH5 LO VOH5 // FORÇAR IOH E PROGRAMAR O LIMITE DE TENSÃO
CONN HILO INPUT END // ENTRADAS EM NÍVEL LÓGICO “1”
STEST OUT // MEDIR VOH NAS SAÍDAS
//—————VOL5—————
T70, TESTN[70] // MACRO DE N. DE TESTE
FORCE IOL5 HI VOL5 // FORÇAR IOL E PROGRAMAR O LIMITE DE TENSÃO
CONN HILO END // ENTRADAS EM NÍVEL LÓGICO “0”
STEST OUT // MEDIR VOL NAS SAÍDAS
//—————IIL—————
T80, TESTN[80] // MACRO DE N. DE TESTE

```

```

MPAUSE 20MS // PAUSA PARA ESTABILIZAÇÃO
CONN HILO INPUT END // OUTRAS ENTRADAS EM 1
FORCE VGND LO IILMÁX // FORÇAR VGND E PROGRAMAR LIMITE DE CORRENTE
STEST INPUT // MEDIR IIL NAS ENTRADAS
//-----III-----
T90, TESTN[90] // MACRO DE N. DE TESTE
CONN HILO END // OUTRAS ENTRADAS EM 0
FORCE VIHMIN LO IIHMÁX // FORÇAR VIHMIN E PROGRAMAR LIMITE DE CORRENTE
STEST INPUT // MEDIR III NAS ENTRADAS
//-----ICC-----
T100, TESTN[100] // MACRO DE N. DE TESTE
ADDR LO VGND HI VMÁX // PROGRAMAR OS DRIVERS A
BDDR LO VINMIN HI VINMÁX // PROGRAMAR OS DRIVERS B
CONN ADDR INPUT // CONECTAR ENTRADAS NO DRIVER A
PAUSE 1MS // PAUSA PARA ESTABILIZAÇÃO
CONN HILO INPUT END // ENTRADAS EM NÍVEL LÓGICO "1"
VTEST VS1 HI ICCMIN // MEDIR O CONSUMO
CONN HILO END // ENTRADAS EM NÍVEL LÓGICO "0"
VTEST VS1 HI ICCMIN // MEDIR O CONSUMO
T110, TESTN[110] // MACRO DE N. DE TESTE
CONN ADDR // DESCONECTAR O DRIVER A
PAUSE 1MS // PAUSA PARA ESTABILIZAÇÃO
GATES // DESLIGAR SISTEMA DE MEDIDAS
CONN AMU VS1 // CONECTAR O AMU NA FONTE VS1
FORCE VCCMÁX HI ICCMÁX //FORÇAR VCCMÁX E PROGRAMAR LIMITE DE CORRENTE
CONN HILO END // ENTRADAS EM NÍVEL LÓGICO "0"
GATES AMU // LIGAR A AMU
PO // MANTER A AMU LIGADA

```

```

MICC[INA1] // CHAMAR MACRO MICC COM INA1 = 2.4V
MICC[INA2] // CHAMAR MACRO MICC COM INA2 = 2.4V
MICC[INB1] // CHAMAR MACRO MICC COM INB1 = 2.4V
MICC[INB2] // CHAMAR MACRO MICC COM INB2 = 2.4V
MICC[INC1] // CHAMAR MACRO MICC COM INC1 = 2.4V
MICC[INC2] // CHAMAR MACRO MICC COM INC2 = 2.4V
MICC[IND1] // CHAMAR MACRO MICC COM IND1 = 2.4V
MICC[IND2] // CHAMAR MACRO MICC COM IND2 = 2.4V
T120, TESTN[120] // MACRO DE N. DE TESTE
CONN HILO INPUT END // ENTRADAS EM NÍVEL LÓGICO "1"
MICC[INA1] // CHAMAR MACRO MICC COM INA1 = 0.5V
MICC[INA2] // CHAMAR MACRO MICC COM INA2 = 0.5V
MICC[INB1] // CHAMAR MACRO MICC COM INB1 = 0.5V
MICC[INB2] // CHAMAR MACRO MICC COM INB2 = 0.5V
MICC[INC1] // CHAMAR MACRO MICC COM INC1 = 0.5V
MICC[INC2] // CHAMAR MACRO MICC COM INC2 = 0.5V
MICC[IND1] // CHAMAR MACRO MICC COM IND1 = 0.5V
MICC[IND2] // CHAMAR MACRO MICC COM IND2 = 0.5V
CONN AMU VS0 // DESCONECTAR A AMU
FIM, STOP // FIM DO PROGRAMA DE TESTE

//-----TESTE FUNCIONAL-----
VAR VCC:SYS // VARIÁVEL TIPO SISTEMA
PROCEDURE FUNC (VCC) // PROCEDIMENTO FUNC
VS1 VCC CLAMP RANGE // PROGRAMAR A FONTE VS1 COM VCC
GATES VS1 // LIGAR VS1
PAUSE 2MS // PAUSA PARA ESTABILIZAÇÃO
CONN HILO

```

```
NONE T
IND2 OUTD T
INC2 OUTC T
INC1 IND1 OUTC OUTD T
INB2 OUTB T
INB1 IND1 IND2 OUTB OUTD T
INB2 INC1 INC2 OUTB OUTC T
INB1 INB2 INC2 IND2 OUTB OUTC OUTD T
INA1 INA2 OUTA T
INA2 IND1 OUTA OUTD T
INA1 INC1 OUTA OUTC T
INA2 INC1 INC2 IND1 IND2 OUTA OUTC OUTD T
INA1 INA2 INB1 INB2 OUTA OUTB T
INA2 INB1 IND2 OUTA OUTB OUTD T
INA1 INB2 INC2 OUTA OUTB OUTC T
INA1 INA2 INB1 INC1 IND1 OUT T
INPUT OUT T END
FTEST
END // FIM DA PROCEDURE
```

A tabela 5.0 apresenta os vetores de teste utilizados pela *procedure func* para teste do circuito integrado 74HCT32. O padrão de teste foi resumido, mas procurando cobrir o maior número das possibilidades de falha.

VETOR	INA1	INA2	INB1	INB2	INC1	INC2	IND1	IND2	OUTA	OUTB	OUTC	OUTD
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0	0	0	1
2	0	0	0	0	0	1	0	0	0	0	1	0
3	0	0	0	0	1	0	1	0	0	0	1	1
4	0	0	0	1	0	0	0	0	0	1	0	0
5	0	0	1	0	0	0	1	1	0	1	0	1
6	0	0	0	1	1	1	0	0	0	1	1	0
7	0	0	1	1	0	1	0	1	0	1	1	1
8	1	1	0	0	0	0	0	0	1	0	0	0
9	0	1	0	0	0	0	1	0	1	0	0	1
10	1	0	0	0	1	0	0	0	1	0	1	0
11	0	1	0	0	1	1	1	1	1	0	1	1
12	1	1	1	1	0	0	0	0	1	1	0	0
13	0	1	1	0	0	0	0	1	1	1	0	1
14	1	0	0	1	0	1	0	0	1	1	1	0
15	1	1	1	1	1	1	1	1	1	1	1	1

Tabela 5.0: Vetores de teste para o 74HCT32.

5.3 RESULTADOS

Serão comentados neste item dois *datalogs*, o primeiro do 74HCT32 e o segundo da memória 62256. A listagem do programa 62256 e os *datalogs* estão apresentados no Anexo I.

5.3.1 "Datalog" do 74HCT32

Foram realizadas medições com o programa de teste 74HCT32 em 3 componentes. O componente de número 1 apresenta falha de IIL no pino 4; o de número 2 apresenta falha na corrente IIL nos pinos 4 e 5, indicado com um R nos testes 92 e 96 do *datalog* apresentado no anexo I.

Na primeira linha do *datalog* é apresentado o número do componente, o *mux* 1

que o programa foi carregado, o nome do programa, no caso 32HCT74, o lote e a data. Nas demais linhas estão os testes. O de número 1 é de continuidade, que está boa. Nos testes 25, 26 e 27 é realizado um teste funcional com tensão de alimentação de 4.5V, 5.0V e 5.5V respectivamente. Todos os 3 componentes passaram nestes testes. Os testes 30, 31, 32, 33 são as medições de VOH para VCC de 4.5V e IOH de -4.0mA de cada saída do componente.

Todos os componentes apresentaram valores maiores que os valores típicos deste parâmetro especificado no manual do componente. Os testes 40, 41, 42 e 43 são as medições de VOL para VCC de 4.5V e IOL de 4.0mA de cada saída do componente.

Todos os componentes apresentaram valores menores que os valores típicos deste parâmetro especificado no manual do componente. Os testes 60, 61, 62 e 63 são as medições de VOH para VCC de 5.5V e IOH de -4.8mA de cada saída do componente.

Todos os componentes apresentaram valores maiores que os valores típicos. Os testes 70, 71, 72 e 73 são as medições de VOL para VCC de 5.5V e IOL de 4.8mA de cada saída do componente. Todos os componentes apresentaram valores menores que os valores típicos. Os testes 80, 81, 82, 83, 84, 85, 86 e 87 medem I_{IH} para VCC de 5.5V V_{in} de 5.0V e V_{ILmáx} nas demais entradas. Os valores foram bem menores que 1.0 μ A especificado no manual do componente.

Cabe aqui um comentário quanto a precisão do equipamento para medir grandezas de correntes tão pequenas como I_{IL} e I_{IL} de circuitos CMOS, que deixa um pouco a desejar devido a idade do equipamento. Se a corrente de fuga for grande, o que interessa, o equipamento detecta, como mostra o resultado de I_{IL} para os componentes 1 e 2, que ultrapassaram o fundo de escala do medidor mostrado no teste 92, pino 4 do componente 1 e nos testes 92 e 96, pinos 4 e 5 do CI 2. Nos testes 90, 91, 92, 93, 94, 95, 96 e 97 I_{IL} é medido para VCC de 5.5V e V_i de 0.8 e V_{IHMáx} nas demais entradas. Os testes de 100 até 127 medem o consumo do componente nas três condições apresentadas no manual, ou seja, com VCC de 5.5V, entradas de VCC e sem carga. O resultado desta medição está no teste 100. O 101 altera o valor das entradas para 0.0V. Os testes 110, 111, 112, 113, 114, 115, 116 e 117 medem o consumo do componente com 0.5V por pino, com os demais em 0V. Os testes 120, 121, 122, 123, 124, 125, 126 e 127 medem o consumo do componente com 2.4V por pino, com os demais em 0V.

Como mostra o *datalog* nenhum componente apresentou problema de consumo.

5.3.2 "Datalog" do HM62256

Foram realizadas medições com o programa de teste 62256, que está no anexo I em 6 componentes numerados de 21 a 25 e 90. O componente de número 90 apresenta falha no teste de continuidade e em todos os testes funcionais, numerados de 10 até 33 como mostra o *datalog* apresentado. O R no *datalog* indica falha. Na primeira linha do *datalog* é apresentado o número do componente, o *mux* 4 que o programa foi carregado, o nome do programa, no caso 62256, o lote e a data. Nas demais linhas estão os testes. O de número 1 é de continuidade, que está boa, exceto no dispositivo 90. Nos testes 10 a 21 é realizado um teste funcional com tensão de alimentação de 4.5V, VIH de 2.20V, VIL de 0.80V escrevendo e lendo na memória 00H, FFH e 55H.

Nos testes 22 a 33 é realizado um teste funcional nas mesmas condições acima para VCC de 5.5V. Os componentes medidos passaram nestes testes, exceto o dispositivo de número 90. Os testes de 40 a 47 medem Voh para VCC de 4.5V, IOH de -1.0mA.

Todos os componentes apresentaram valores acima do típico, nenhuma falha. Nos testes de 50 até 57 o Vol da memória é medido, para VCC de 5.5V, Iol de 2.1mA. Também neste teste não houve falhas. Os resultados apresentados são inferiores ao máximo especificado no manual. Os testes de 60 a 77 realizam a medição de IIH para VCC de 5.5V e VIH de 5.0V, dos pinos de endereçamento e de controle. Os valores foram bem menores que $2.0\mu A$ especificado no manual do componente.

Os testes de 80 até 97 realizam a medição de IIL para VCC de 5.5V e 0.0V nas entradas dos pinos de endereçamento e de controle. Não houve falhas nestes testes. Os testes de 100 até 107 medem a corrente de fuga dos pinos de dados, para VCC de 5.5V forçando 5.5V nos dados.

Como mostra o *datalog*, não falhou nenhum componente neste teste. Os testes 110 a 117 realizam a mesma medição dos testes 100 a 107 mas 0.0V é forçado nos pinos de dados do componente. Não houve problemas neste teste também. Finalmente o consumo do componente é medido. Primeiro o consumo em *standby* ou seja, o circuito desabilitado e depois o consumo do circuito habilitado, neste caso realizando uma leitura.

Como mostra o *datalog* nenhum componente apresentou problema de consumo. É interessante observar que o dispositivo 90 apresentou um consumo em *standby* de 1.3mA, bem acima do valor típico do manual.

5.4 CONCLUSÃO

Com relação a utilização de equipamentos automáticos de teste em circuitos integrados digitais, podemos concluir que o teste de um circuito integrado digital exerce um papel importante nas diversas etapas da vida de um integrado, desde a depuração do protótipo na fase de projeto do *chip*, até a análise de falhas do circuito no campo.

O teste de circuitos integrados digitais aparentemente é um trabalho simples e fácil, mas na realidade não é, pois envolve vários fatores como:

- temperatura do ambiente de teste,
- *hardware* auxiliar ao teste,
- o programa de teste,
- o padrão para o teste funcional,
- a precisão e exatidão requerida para as medições paramétricas,
- a repetibilidade de uma medição,
- esquema de calibração e aferição do ATE,
- o perfeito funcionamento do equipamento automático de teste - ATE, etc ...

O custo de um teste, também, à primeira vista, pode parecer baixo, mas na realidade não é. Deve ser considerado durante o projeto do circuito integrado. Pode representar de 40 a 60% do custo do projeto.

O equipamento automático de teste - ATE é um equipamento poderoso, complexo e caro, que necessita para um perfeito funcionamento de condições ambientais controladas, implicando

na necessidade de uma sala com infra-estrutura de ar condicionado e elétrica adequadas ao ambiente de teste. Este equipamento necessita, também, de equipes de engenheiros e de técnicos para operação, manutenção, aferição/calibração e programação.

Os programas de teste necessitam também de engenharia para o seu desenvolvimento e validação. Em uma etapa posterior ao desenvolvimento do programa de teste, é importante ter pessoal especializado para manutenção dos *softwares* de teste e para a execução dos programas no ATE. Uma das partes mais complexas de um programa de teste é a elaboração do padrão funcional para o teste do dispositivo. Para realizar esta atividade, simuladores, linguagens de descrição de *hardware*, estações de trabalho, etc ... são algumas das ferramentas utilizadas pelo engenheiro.

Devido a grande complexidade dos circuitos integrados digitais, com escala de integração cada vez maior, é muito importante que os projetistas de circuitos integrados tenham em mente a dificuldade de se testar seus projetos. Por isso é fundamental projetar circuitos com a testabilidade embutida, ou seja facilitar o teste do CI. Facilitar o controle do estado lógico dos pinos de saídas, para facilitar as medições paramétricas é um exemplo. Daí a necessidade de engenheiros de teste participarem da equipe de projeto de determinado circuito integrado digital. A cerca de cinco anos passados, testar circuitos eletrônicos era uma atividade que acontecia no final de uma seqüência de eventos, envolvidos na pesquisa e desenvolvimento como : projeto, desenvolvimento do protótipo e produção. No passado isto era aceitável porque a complexidade dos circuitos eletrônicos digitais não era tão grande como hoje. A tecnologia dos circuitos integrados está evoluindo tanto que aquela perspectiva do passado está totalmente alterada. Hoje é imperativo para o projetista considerar a testabilidade nos primeiros estágios do projeto conceitual, devido ao alto custo de não se saber testar os complexos componentes, o que pode causar o colapso total do projeto.

Indústrias fabricantes de componentes LSI, VLSI e sistemas integrados acham que o alto custo do teste podem ser reduzidos somente reunindo as atividades de teste e de projeto, conduzindo para o conceito de “ Projeto testável” (*design for testability*).

5.5 REFERÊNCIAS BIBLIOGRÁFICAS

- TERADYNE, *J325 Programming Manual*. April 1983.
- TERADYNE, *J325 Programming class* . July 1984.
- NATIONAL Semiconductor. CMOS logic databook. 1988.
- HITACHI IC Memory N.1
- BENNETTS, R. G. *Design of testable logic circuits* . London, Addison-Wesley, 1984.