

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

**Estimativa de Capacitâncias e
Consumo de Potência em Circuitos
Combinacionais CMOS no
Nível Lógico**

por

JOÃO BAPTISTA DOS SANTOS MARTINS

Tese submetida à avaliação, como requisito parcial para
a obtenção do grau de Doutor em
Ciência da Computação

Prof. Dr. Ricardo Augusto da Luz Reis
Orientador

Prof. Dr. José Carlos Monteiro
Co-Orientador

Porto Alegre, julho de 2001

CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Martins, João Baptista dos Santos.

Estimativa de Capacitâncias e Consumo de Potência em Circuitos Combinacionais CMOS no Nível Lógico / por João Baptista dos Santos Martins - Porto Alegre: PPGC da UFRGS, 2001.

112 p.:il

Tese (doutorado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2001. Orientador: Reis, Ricardo Augusto da Luz.

1. Estimativa de Consumo de Potência. 2. Portas Lógicas. 3. Comprimento Médio das Interconexões. 4. Atividades de Comutação. 5. Capacitâncias. 6. CAD. 7. Microeletrônica. I. Reis, Ricardo Augusto da Luz. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitora: Profa. Wrana Panizzi

Pró-Reitor de Ensino: Prof. José Carlos Ferraz Hennemann

Pró-Reitor Adjunto de Pós-Graduação: Prof. Philippe Olivier A. Navaux

Diretor do Instituto de Informática: Prof. Philippe Olivier Alexandre Navaux

Coordenador do PPGC: Prof. Carlos Alberto Heuser

Bibliotecária Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

Agradecimentos

Inicialmente agradeço a CAPES pela bolsa e a Universidade Federal de Santa Maria pelo aporte financeiro durante o tempo que estive cursando o doutorado.

São muitas as pessoas a agradecer.

Começaria os agradecimentos pelo orientador Dr. Ricardo Reis pela sua orientação, entusiasmo, iniciativa, compreensão e amizade ao longo dos anos que trabalhamos juntos nesta Tese.

Ao pessoal do INESC de Portugal, especialmente ao professor Dr. José Carlos Monteiro pela sua valiosa e fundamental orientação na primeira parte deste trabalho. Também ao Ms. Luiz Guerra pelas dicas na parte de programação C e aos colegas do INESC pela hospitalidade e cordialidade com que fui recebido quando lá estive.

A todos os colegas do GME, com os quais convivi durante estes cinco anos, em especial ao Choi, Guntzel, Zeferino, Macarthy, Rolf e Gomes pelas ajudas e incentivo que me deram.

Ao Professor Dr. Fernando Moraes pela disponibilidade, compreensão, amizade e participação da orientação na segunda etapa desta Tese.

Aos demais colegas da UFRGS, em especial ao Ladeira, Jugurta, Hércules e Ceretta pela amizade e companheirismo.

A Ida, Bea, Zita e demais colegas da biblioteca pela cordialidade, simpatia e disponibilidade. Ao LO e sua turma de apoio pela simpatia e profissionalismo.

Ao meu irmão Paulo e sua esposa Nina pela recepção que tive em sua casa nos dois primeiros anos do Curso.

A minha mãe e em memória do meu pai, que tenho certeza, se estivesse junto de nós hoje estaria muito feliz por esta conquista.

Aos sogros Luiz e Cecy pelo incentivo.

A Deus, por ter me dado muita saúde, força, disposição e inteligência na difícil luta de realizar este Curso.

E finalmente as minha preciosidades, Luccianne, Leonardo, Thiago e Karolline pela compreensão nos vários momentos que foram privados da minha atenção e amor. Desta forma, dedico a eles esta Tese como fruto de muito trabalho.

Sumário

Lista de Figuras	7
Lista de Tabelas	9
Resumo	11
Abstract	12
1 Introdução	13
1.1 Contribuições da Tese	16
1.2 Organização da Tese	16
2 Fundamentação Teórica e Trabalhos Relacionados	17
2.1 Fontes de Consumo	17
2.1.1 Consumo de Potência devido às Correntes de Fuga	17
2.1.2 Consumo de Potência devido à Corrente de Curto-Circuito.....	17
2.1.3 Potência Dinâmica	19
2.1.4 Consumo de Potência devido aos <i>Glitching</i>	20
2.1.5 Delimitação do Trabalho de estimativa de Potência.....	20
2.2 Níveis de Estimativa de Potência	21
2.2.1 Estimativa de Potência ao Nível de Sistema.....	21
2.2.2 Estimativa de Potência ao Nível de Transferência entre Registradores	24
2.2.3 Estimativa de Potência ao Nível de Portas Lógicas.....	26
2.2.4 Estimativa de Potência ao Nível de Transistores	27
3 Estimativa de Potência ao Nível Lógico	28
3.1 Técnica Estatística de Estimativa de Potência	28
3.1.1 Formulação do Problema	29
3.1.2 Critério de Interrupção	30
3.1.3 <i>Setup</i> e Amostragem	30
3.2 Técnicas Probabilísticas	31
3.2.1 Simulação Probabilística.....	32
3.2.2 Densidade de Transição	32
3.2.3 BDD associado a probabilidade de comutação.....	33
3.2.4 Coeficiente de Correlação Temporal e Espacial.....	33
3.2.5 Estimativa Simbólica	34

4	Estimativa de Capacitâncias e Potência Dinâmica	37
4.1	Introdução	37
4.2	Ferramenta Auxiliar na Validação do Método Proposto	38
4.2.1	Descrição da Ferramenta TROPIC	39
5	Estimativa de Potência em Portas Padrões	40
5.1	Introdução	40
5.2	Capacitâncias Intrínsecas	40
5.3	Estimativa de Capacitância de Porta	41
5.3.1	Capacitâncias do INVERSOR CMOS.....	42
5.3.2	Capacitâncias da Porta Lógica NAND de duas entradas	42
5.3.3	Capacitâncias da Porta Lógica NAND de três entradas	46
5.3.4	Capacitâncias de Portas Lógicas NAND de n -entradas.....	50
5.3.5	Capacitâncias de Portas Lógicas NOR de “ n ” entradas	53
5.4	Potência Média Dissipada pela Porta	53
5.5	Validação do Método de Estimativa de Potência de Porta	53
5.5.1	Simulação Elétrica	55
5.5.2	Condições Gerais para Simulações de Circuitos	56
5.5.3	Resultados e Conclusões.....	56
5.6	Aproximações nas Dimensões dos Transistores	58
5.6.1	Inversor CMOS.....	59
5.6.2	NAND com duas entradas	59
5.6.3	NAND com número de entradas qualquer.....	60
5.7	Conclusões	61
6	Estimativa do Consumo de Potência devido as Interconexões	63
6.1	Introdução	63
6.2	Capacitâncias de Interconexões	63
6.3	Comprimento Médio de Interconexões	63
6.3.1	Tecnologia AMS 0,25 μm	64
6.3.2	Tecnologia AMS 0,35 μm	72
6.4	Avaliação da metodologia para obtenção do comprimento médio para a tecnologia 0,25μm	78
6.5	Capacitâncias devido às Interconexões	80

6.6	Potência devido às Interconexões	80
6.7	Avaliação da Estimativa de Capacitâncias de Interconexões	81
6.8	Estimativa da Potência Total	82
6.9	Estimativa do Consumo de Potência de Portas Complexas	83
7	Conclusões	85
7.1	Trabalhos Futuros	85
Anexo 1	87
Anexo 2	99
Referências	108

Lista de Figuras

FIGURA 2.1	– Inversor CMOS.....	18	
FIGURA 2.2	– Principais capacitâncias de um circuito inversor	19	
FIGURA 2.3	– Efeito do modelo de atraso na geração e propagação do <i>glitching</i>	20	
FIGURA 2.4	– Variações do posicionamento das células no <i>leiaute</i> de um circuito... ..	24	
FIGURA 2.5	– Entropia de uma variável Booleana versus probabilidade	25	
FIGURA 3.1	– Método de Monte Carlo	29	
FIGURA 3.2	– Tempos de amostragem e <i>setup</i>	31	
FIGURA 3.3	– Atividade de comutação de um circuito X tempo de propagação	32	
FIGURA 3.4	– BDD da função AND2 associado as Atividades de Comutação.....	33	
FIGURA 3.5	– Correlação Temporal	34	
FIGURA 3.6	– Correlação Espacial	34	
FIGURA 3.7	– Exemplo de circuito para simulação simbólica	34	
FIGURA 3.8	– Rede simbólica para modelo de atraso zero.....	35	
FIGURA 3.9	– Rede simbólica com modelo de atraso unitário.	36	
FIGURA 4.1	– Fluxograma do método proposto para estimativa da potência.....	38	
FIGURA 4.2	– Implementação do método matriz linear nas versões do TROPIC	39	
FIGURA 5.1	– Capacitâncias Parasitas Intrínsecas de um transistor MOSFET	40	
FIGURA 5.2	– Capacitâncias Parasitas Intrínsecas do Inversor CMOS	42	
FIGURA 5.3	– Capacitâncias Parasitas da porta lógica NAND2 CMOS	44	
FIGURA 5.4	– Capacitâncias Parasitas da porta lógica NAND3 CMOS	47	
FIGURA 5.5	– Validação do método proposto	54	
FIGURA 5.6	– Diagrama de estados para um circuito com duas entradas.....	55	
FIGURA 5.7	– Vetores de simulação para um circuito com duas entradas	56	
FIGURA 5.8	– Aproximações das Capacitâncias Intrínsecas	59	
FIGURA 5.9	– NAND com duas entradas e dimensões dos transistores estimadas	59	
FIGURA 6.1	– Capacitâncias devidas as interconexões.....	63	
FIGURA 6.2	– Comp. médio das redes versus núm. de trans. para <i>fanout</i>	igual a ZERO (tecnologia AMS 0,25 μ m).....	66
FIGURA 6.3	– Comp. médio das redes versus núm. de trans. para <i>fanout</i>	igual a UM (tecnologia AMS 0,25 μ m).....	67
FIGURA 6.4	– Comp. médio das redes versus núm. de trans. para <i>fanout</i>	igual a DOIS (tecnologia AMS 0,25 μ m).....	67
FIGURA 6.5	– Comp. médio das redes versus núm. de trans. para <i>fanout</i>	igual a TRÊS (tecnologia AMS 0,25 μ m)	68

FIGURA 6.6 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a QUATRO (tecnologia AMS 0,25 μ m).....	69
FIGURA 6.7 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a CINCO (tecnologia AMS 0,25 μ m).....	69
FIGURA 6.8 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a SEIS (tecnologia AMS 0,25 μ m).....	70
FIGURA 6.9 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> maior do que SEIS (tecnologia AMS 0,25 μ m).....	71
FIGURA 6.10 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a ZERO (tecnologia AMS 0,35 μ m).....	74
FIGURA 6.11 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a UM (tecnologia AMS 0,35 μ m).....	74
FIGURA 6.12 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a DOIS (tecnologia AMS 0,35 μ m).....	74
FIGURA 6.13 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a TRÊS (tecnologia AMS 0,35 μ m).....	75
FIGURA 6.14 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a QUATRO (tecnologia AMS 0,35 μ m).....	76
FIGURA 6.15 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a CINCO (tecnologia AMS 0,35 μ m).....	76
FIGURA 6.16 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> igual a SEIS (tecnologia AMS 0,35 μ m).....	77
FIGURA 6.17 – Comp. médio das redes versus núm. de trans. para <i>fanout</i> maior do que SEIS (tecnologia AMS 0,35 μ m).....	78
FIGURA 6.18 – Esquemas lógico e elétrico de uma porta complexa	83

Lista de Tabelas

TABELA 1.1 – Microprocessadores projetados sem redução de consumo de potência	13
TABELA 1.2 – Microprocessadores projetados para baixo consumo de potência	14
TABELA 5.1 – Número de entradas versus número de combinações de entrada	55
TABELA 5.2 – Consumo de Potência de Portas Lógicas CMOS convencionais	57
TABELA 5.3 – Características dos Circuitos <i>Benchmarks</i> Analisados	57
TABELA 5.4 – Estimativa da potência de porta em circuitos CMOS	58
TABELA 6.1 – Característica dos Circuitos usados no levantamento dos dados	64
TABELA 6.2 – Dados usados na geração dos leiautes dos circuitos para tecnologia AMS 0,25 μ m	65
TABELA 6.3 – Comprimento médio (μ m) das redes para <i>fanouts</i> iguais a 0, 1 e 2 (tecnologia 0,25 μ m)	66
TABELA 6.4 – Comprimento médio (μ m) das redes para <i>fanouts</i> iguais a 3, 4 e 5	68
TABELA 6.5 – Comprimento médio (μ m) das redes para <i>fanouts</i> iguais a 6 e maiores	70
TABELA 6.6 – Percentual de redes para diferentes <i>fanouts</i> por circuito	71
TABELA 6.7 – Tabela de referência para estimativa do comprimento médio das redes para tecnologia AMS 0,25 μ m	72
TABELA 6.8 – Tabela de referência para estimativa do comprimento médio das entradas para tecnologia AMS 0,25 μ m	72
TABELA 6.9 – Dados usados na geração dos leiautes dos circuitos para tecnologia AMS 0,35 μ m	73
TABELA 6.10 – Comprimento médio (μ m) das redes para <i>fanouts</i> iguais a 0, 1 e 2 (tecnologia 0,35 μ m)	73
TABELA 6.11 – Comprimento médio (μ m) das redes para <i>fanouts</i> iguais a 3, 4 e 5 (tecnologia 0,35 μ m)	75
TABELA 6.12 – Comprimento médio (μ m) das redes para <i>fanouts</i> iguais a 6 e maiores (tecnologia 0,35 μ m)	77
TABELA 6.13 – Tabela de referência para estimativa do comprimento médio das redes para tecnologia AMS 0,35 μ m	78
TABELA 6.14 – Comprimento médio de redes para circuitos <i>benchmarks</i> obtidos da extração elétrica versus estimativa lógica (tecnologia 0,25 μ m)	79
TABELA 6.15 – Análise do método para outros circuitos <i>benchmarks</i> (tecn. 0,25 μ m)	79
TABELA 6.16 – Comparação das capacitâncias estimadas ao nível lógico com capacitâncias extraídas do leiaute	81
TABELA 6.17 – Tempos de Processamento: Extração das Capacitâncias versus Estimativa Lógica	82
TABELA 6.18 – Estimativa da potência total de circuitos <i>benchmarks</i> ao nível lógico	82

TABELA 6.19 – Número de células possíveis em função da altura da célula 84

Resumo

Esta tese propõe o desenvolvimento de um método de estimativa de capacitâncias e de potência consumida nos circuitos combinacionais CMOS, no nível de portas lógicas. O objetivo do método é fazer uma previsão do consumo de potência do circuito na fase de projeto lógico, o que permitirá a aplicação de técnicas de redução de potência ou até alteração do projeto antes da geração do seu leiaute.

A potência dinâmica consumida por circuitos CMOS depende dos seguintes parâmetros: tensão de alimentação, frequência de operação, capacitâncias parasitas e atividades de comutação em cada nodo do circuito.

A análise desenvolvida na Tese, propõe que a potência seja dividida em duas componentes. A primeira componente está relacionada ao consumo de potência devido às capacitâncias intrínsecas dos transistores, que por sua vez estão relacionadas às dimensões dos transistores. Estas capacitâncias intrínsecas são concentradas nos nodos externos das portas e manifestam-se em função das combinações dos vetores de entrada. A segunda componente está relacionada às interconexões entre as células do circuito. Para esta etapa utiliza-se a estimativa do comprimento médio das interconexões e as dimensões tecnológicas para estimar o consumo de potência. Este comprimento médio é estimado em função do número de transistores e *fanout* das várias redes do circuito.

Na análise que trata das capacitâncias intrínsecas dos transistores os erros encontrados na estimativa da potência dissipada estão no máximo em torno de 11% quando comparados ao SPICE. Já na estimativa das interconexões a comparação feita entre capacitâncias de interconexões estimadas no nível lógico e capacitâncias de interconexões extraídas do leiaute apresentou erros menores que 10%.

Palavras-Chave: Estimativa de Consumo de Potência, Portas Lógicas, Comprimento Médio das Interconexões, Atividades de Comutação, Capacitâncias, CAD, Microeletrônica.

TITLE: “CAPACITANCES ESTIMATE AND POWER CONSUMPTION IN THE CMOS COMBINATIONAL CIRCUITS AT THE LOGICAL LEVEL”

Abstract

This Thesis proposes the development of a estimate method regarding capacitances and power consumed in the CMOS combined circuits at the logical level. The purpose of the method is to foresee the power consumption of the circuit during the logical project phase, since it will allow the application of power reduction techniques or the design alteration before its layout would be created.

The dynamic power consumed by CMOS circuits depends on the following parameters: power supply, operation frequency, parasite capacitances and switching activity in each circuit node.

The analysis developed here considers power as being divided in two components. The first one is related to power consumption as a result of transistors intrinsic capacitances. Such capacitances are related to the dimensions of the transistors themselves. Besides, the intrinsic capacitances are concentrated in the external nodes of the gates and are disclosed by the input vectors combinations. The second component is related to interconnections among the circuit cells. At this stage, their average length estimate and the technological dimensions are used to estimate the power consumption. Important to say that this average length is valued considering the number of transistors and fanout of the several circuit nets.

Concerning the analysis of the transistors intrinsic capacitances, the errors found in the wasted power estimate are at most around 11% when compared to the SPICE. However, concerning the interconnections estimate, the comparison drawn between capacitances of estimated interconnections at the logical level and capacitances of interconnections extracted from the layout presented errors under 10%.

Keywords: Power Consumption Estimate, Logical Gates, Interconnections Average Length, Commutation Activities, Capacitances, CAD, Microelectronics.

1 Introdução

Nos primórdios da microeletrônica a maior preocupação dos projetistas e pesquisadores na área de semicondutores estava relacionada à otimização de área útil de silício (ou germânio) ocupada pelos transistores e atraso na propagação do sinal. Otimizar área de silício significa redução na largura (W) e comprimento do canal (L) dos transistores de um circuito integrado, ou ainda, redução nos custos de prototipação. De outra maneira, sabe-se que transistores com W pequeno significa maior atraso do sinal, o que pode ocasionar problemas de transitórios e erros na funcionalidade de todo o circuito. Portanto, área e atraso são parâmetros conflitantes e que estão relacionados diretamente na confecção do leiaute do circuito.

Com o passar dos anos e avanço tecnológico na área, especialmente devido ao aumento significativo da demanda em computação de alta velocidade (multimídia envolvendo áudio e vídeo), sistemas de comunicação sem fio, telemetria, etc .. o consumo da potência dissipada no circuito integrado foi outro fator que adquiriu importância significativa no projeto de circuitos integrados.

O consumo de potência produz sobreaquecimento, diminuindo a confiabilidade do circuito e a vida útil do sistema. Técnicas de resfriamento forçada podem ser usadas, porém conduzem a custos elevados e geram produtos com grandes dimensões e inviáveis em termos de praticidade de uso. Por esta razão, existe uma grande demanda de pesquisas com o objetivo de desenvolver técnicas de redução da potência de um circuito.

As técnicas de redução de potência desenvolvidas tratam da redução do consumo em diferentes níveis de projeto e vão desde a concepção de um sistema em alto nível até a sua fabricação ao nível de processo físico-químico.

As Tabelas 1.1 e 1.2 fazem uma comparação entre processadores projetados sem o uso de técnicas de redução de consumo e com o uso de técnicas de redução de consumo [BEL 95], respectivamente. Observa-se pela tabelas reduções consideráveis na potência dissipada entre as duas análises.

TABELA 1.1 – Microprocessadores projetados sem redução de consumo de potência

Processador	Velocidade (MHz)	Tecnologia (microm)	Vdd (volts)	Pico de Potência(V)	Referência
Intel Pentium	66	0,8	5	16	[SPE 93]
DEC-21064	200	0,75	3,3	30	[DOB 92]
DEC-21164	300	0,5	3,3	50	[BOW 95]
PowerPC620	133	0,5	3,3	30	[BEA 95]
UltraSparc	167	0,45	3,3	30	[CHA 95]

TABELA 1.2 - Microprocessadores projetados para baixo consumo de potência

Processador	Velocidade (MHz)	Tecnologia (microm)	Vdd Volts	Pico de Potência (Volts)	Referência
PowerPC603	80	0,5	3,3	2,2	[GER 94]
IBM486SLC2	66	0,8	3,3	1,8	[BEC 94]
MIPS R4200	80	0,64	3,3	1,8	[YEU 94]

Como veremos no capítulo 2, a frequência de processamento do sinal, a tensão de alimentação e as capacitâncias parasitas são os maiores responsáveis pelo consumo da potência nos circuitos digitais.

Em especial, as capacitâncias parasitas estão relacionadas diretamente as dimensões e quantidade de transistores integrados no circuito e ao número de interconexões.

É de extrema importância o desenvolvimento de ferramentas de CAD (*Computer Aided Design*) que auxiliem no projeto visando a estimativa e redução da potência dissipada, especialmente pela grande complexidade dos circuitos atuais. Não são raros nos dias de hoje, processadores e circuitos digitais com milhões de transistores operando em frequências da ordem de gigahertz (GHz) .

Várias são as técnicas usadas na redução do consumo de potência, no entanto antes de propor novas técnicas é fundamental estimar em um tempo mínimo de avaliação, quanto de potência será consumida e quais as partes do circuito que terão maior consumo.

Quando fala-se em estimativa da potência dissipada, refere-se ao consumo médio de potência e não a potência de pico que pode acontecer em alguns pontos do circuito durante curtíssimo intervalo de tempo. Estes picos de corrente são denominados de transitórios e não serão tratados neste trabalho.

O objetivo deste trabalho é estimar rapidamente a potência consumida por um circuito CMOS (*Complementar MOSFET*), o que possibilitará a implementação de alterações no circuito na fase de projeto. Estimar rapidamente a potência significa reduzir o tempo de projeto evitando realimentações no processo de síntese de leiaute. O resultado imediato desta ação será a redução nos custos de projeto. Através de métodos e algoritmos de estimativa de consumo no nível de portas lógicas bem definidos, será possível atingir este objetivo.

Conforme citado anteriormente, existem muitas técnicas utilizadas para estimar a potência de circuitos CMOS, que vão desde o nível do leiaute até o nível de sistema pela descrição comportamental ou estrutural. Esta análise é vista em [NEM 96], [MAR 96], [MAC 98], [NEM 99], [HIS 98].

Em [NEM 96] é apresentada uma técnica de estimativa de potência que trabalha ao nível de transferência entre registradores (RTL) e utiliza-se das propriedades de entropia do circuito como medida da atividade de comutação.

Em uma estimativa de consumo pós-leiaute é necessário efetuar uma extração elétrica dos transistores e capacitâncias do circuito, utilizando-se de uma modelagem física e funcional dos transistores. O SPICE [NAG 75] é uma das ferramentas usadas para estimar potência neste nível, porém sua utilização está limitada ao tempo de

processamento devido ao número de entradas e complexidade do circuito. Simulação de circuitos de grande complexidade demandam um tempo muito grande de processamento e muitas vezes sem solução.

Esta tese trata do desenvolvimento de um algoritmo capaz de estimar as capacitâncias intrínsecas dos transistores e de interconexões de circuitos digitais CMOS, possibilitando desta forma estimar a sua potência no nível de portas lógicas. Neste nível a estimativa de potência está baseada nas atividades e capacitâncias de comutação das portas lógicas, onde as informações de leiaute não são oferecidas, pois os circuitos não são gerados fisicamente. Nas ferramentas existentes, capacitâncias são estimadas de um modo empírico e com pouca precisão. Um exemplo é a ferramenta *Power Estimate* [MON 97] que encontra-se integrada na ferramenta de CAD denominada de SIS [SEN 92].

De maneira geral, as atividades de comutação são estimadas usando-se basicamente dois métodos matemáticos: o estatístico e o probabilístico.

O **método estatístico** apresentado em [BUR 93] consiste na simulação do circuito através da avaliação dos ganhos de correntes de saída do circuito pelas correntes de entrada. O valor médio das correntes na saída multiplicada pela tensão de alimentação representa a potência média do circuito. A eficiência do método de simulação está diretamente relacionada a qualidade das informações dos vetores de entradas aplicados pelo simulador. Desta forma, é importante simular um grande número de combinações de entradas a que o circuito pode ser submetido.

Existem duas maneiras de proceder esta simulação:

Uma delas é usar o método exaustivo que prevê todas as combinações de entradas possíveis, o que inviabiliza o processo de simulação nos circuitos que possuem muitas entradas.

A outra maneira, é usar um programa gerador de estímulos. Neste caso, as entradas são geradas aleatoriamente e as saídas avaliadas continuamente até que um número pré-definido de estímulos seja alcançado. Este método apresenta maior imprecisão do que o exaustivo, mas produz resultados satisfatórios para aqueles casos em que a precisão dos resultados não é o objetivo maior, mas simplesmente a indicação dos nodos do circuito onde o consumo de potência é maior.

De uma forma geral, o critério de escolha do tipo de simulação depende basicamente do número de entradas do circuito. Para um circuito que possui um número pequeno de entradas o método mais usado é o exaustivo. Para um circuito com muitas entradas o método utilizado é o da geração de vetores aleatórios, independentemente da tecnologia, estilo de projeto, funcionalidade ou arquitetura.

O **método probabilístico** de estimativa de potência apresentado em [MON 97] é baseado na probabilidade de comutação dos sinais elétricos nas portas lógicas do circuito. Um sinal quando comuta de um nível lógico "0" para "1" ou vice-versa sofre um atraso. Este atraso é analisado em três formas: atraso zero, atraso unitário e atraso genérico.

Atraso zero é aquele que considera a comutação instantânea dos sinais de um circuito quando este faz uma transição de "0" lógico para "1" lógico ou vice-versa.

Atraso unitário considera que todas as portas lógicas de um circuito possuem o mesmo valor de atraso.

Atraso genérico considera que cada porta lógica possui seu próprio atraso, em função das suas capacitâncias e resistências de comutação.

O atraso genérico possibilita estimar as atividades de comutação dos nodos internos das portas lógicas. Alguns autores apresentam variações do método, [DIN 98] por exemplo, desenvolve uma técnica de estimativa de potência no nível de portas lógicas baseada na noção de probabilidades das correntes de saída da porta, cuja representação é feita através de formas de onda, que modelam todos os possíveis eventos da saída em cada nodo do circuito. As formas de onda em cada nodo são obtidas pela segmentação da forma de onda do circuito de acordo com os valores iniciais e finais, e compactação de todas as formas de onda de cada particionamento em uma simples forma de onda lógica agrupada.

[LIN 94] desenvolve a modelagem comportamental de potência de uma porta lógica, através de um modelo denominado de CBPE (*Cell Based Power Estimation*) e de um gráfico de transições de estados modificado STGPE (*State Transition Graphic Power Estimation*). O número de atividades em cada transição do STGPE é estimado em função das probabilidades de comutação dos sinais de entrada e densidades de transição da porta lógica. Densidade de transição de uma porta lógica é definida como o número de transições do sinal sobre a porta lógica pelo tempo que está sendo submetido por unidade de tempo. A potência consumida é calculada pela soma das energias dissipadas em cada transição de estado do STGPE.

O método probabilístico trata apenas das transições dos sinais nas portas lógicas não estimando capacitâncias, o que torna o método pouco preciso.

1.1 Contribuições da Tese

Esta Tese apresenta as seguintes contribuições :

Estimativa das capacitâncias intrínsecas e **potência de dissipada em portas lógicas** de circuitos combinacionais CMOS no nível lógico;

Estimativa das capacitâncias e **potência dissipada nas interconexões** de circuitos combinacionais CMOS no nível lógico;

Estimativa da **potência total** consumida nos circuitos combinacionais CMOS no nível lógico.

1.2 Organização da Tese

A Tese está organizada da seguinte forma: os capítulos 2 e 3 fazem uma revisão de conceitos e técnicas de estimativa de potência nos vários níveis, em especial no nível de portas lógicas. O capítulo 4 destaca a importância de estimar capacitâncias no nível de portas lógicas, descreve a ferramenta auxiliar na geração automática de leiaute. O capítulo 5 analisa a estimativa do consumo médio de potência de circuitos, a qual está baseada nas capacitâncias intrínsecas de cada porta lógica e combinação dos vetores de entrada. O capítulo 6 trata da estimativa de potência devido às interconexões entre as portas lógicas e da estimativa de consumo total de potência do circuito no nível lógico. O capítulo 7, conclui o trabalho destacando contribuições e perspectivas de trabalhos futuros.

2 Fundamentação Teórica e Trabalhos Relacionados

Este capítulo revisa brevemente os principais trabalhos relacionados à estimativa de potência em circuitos CMOS. Inicialmente é efetuada uma rápida revisão das principais fontes de consumo de potência em circuitos CMOS.

2.1 Fontes de Consumo

As principais fontes de consumo de potência em circuitos formados de portas lógicas CMOS são:

- Correntes de fuga, que ocasionam uma dissipação de potência estática;
- Correntes de curto-circuito;
- Carga e descarga das capacitâncias durante a transição do sinal lógico;
- Transições espúrias do sinal .

A potência total consumida por um circuito CMOS assume a seguinte expressão:

$$P = P_{fuga} + P_{cc} + P_{din} + P_{haz} \quad (1)$$

onde:

P = potência total dissipada no circuito

P_{fuga} = potência estática devido as correntes de fuga

P_{cc} = potência dissipada devido a corrente de curto circuito

P_{din} = potência dinâmica dissipada devido a carga e descarga das capacitâncias

P_{haz} = potência devido aos *hazards*

2.1.1 Consumo de Potência devido às Correntes de Fuga

Idealmente, circuitos CMOS não deveriam dissipar potência estática, pelo fato de teoricamente não existir corrente no caminho entre a tensão de alimentação da porta e o terra. Na prática, esta condição jamais acontecerá, pois a comutação dos sinais nos transistores nunca é ideal. Existirão correntes de fuga na porta e injeção de corrente no substrato do transistor MOS, o que elevará o valor da componente estática de potência [RAB 96]. Este tipo de consumo de potência não será tratado nesta tese.

2.1.2 Consumo de Potência devido à Corrente de Curto-Circuito

Quando os transistores de um circuito CMOS comutam de um nível “0” para “1” ou vice-versa, os tempos de descida e subida do sinal não são instantâneos. Em curtíssimo intervalo de tempo, os transistores *pmos* e *nmos* conduzirão simultaneamente. Durante este tempo, fluirá uma pequena corrente de curto-circuito entre V_{dd} e terra, independente da carga ou descarga das capacitâncias do circuito.

Considere o inversor estático CMOS da Figura 2.1, cuja entrada sofre uma transição de $0 \rightarrow 1$. Inicialmente a capacitância de carga é muito grande, de modo que o

tempo de descida do sinal na saída é significativamente maior do que o tempo de subida na entrada. Como a tensão fonte-dreno no transistor *pmos* é zero, durante este período não haverá condução de corrente, e portanto a corrente de curto-circuito também será zero. No caso inverso, onde a capacitância de saída é muito baixa e o tempo de descida na saída é pequeno em relação ao tempo de subida na entrada, a tensão dreno fonte do transistor *pmos* é igual a V_{dd} durante este período de transição, o que eleva a corrente de alimentação ao seu valor máximo.

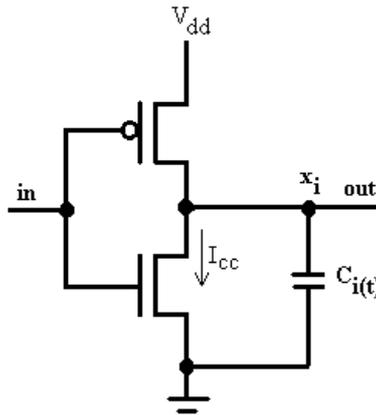


FIGURA 2.1 - Inversor CMOS

A dissipação de potência devido às correntes de curto-circuito pode ser minimizada através do ajuste entre os tempos de subida e descida dos sinais durante a sua comutação. Normalmente, o consumo de potência de curto-circuito não é preocupante quando comparado ao consumo de potência dinâmica. Normalmente, a potência de curto circuito (P_{cc}) não chega a 10% da potência dinâmica [RAB 96].

O consumo da potência estática modelado por [VEE 84] é o seguinte:

$$P_{cc} = \frac{\beta}{12} (V_{dd} - V_{thn} - V_{thp})^3 \frac{\tau}{T} \quad (2)$$

onde:

P_{cc} = potência estática consumida devido a corrente de curto-circuito [W]

β = parâmetro físico que depende de W e L do transistor – $k'(W/L)$ - [A/V²]

V_{dd} = tensão de alimentação [V]

V_{thn} = tensão efetiva de condução do transistor *nmos* [V]

V_{thp} = tensão efetiva de condução do transistor *pmos* [V]

τ = tempo de transição do sinal [s]

T = período do sinal de entrada (tempo médio entre transições) [s]

2.1.3 Potência Dinâmica

A maior parcela do consumo de potência nos circuitos CMOS deve-se a comutação dos transistores quando o sinal de entrada variar do nível lógico "0" para o nível lógico "1" ou vice-versa. A potência dinâmica depende de quatro fatores: tensão de alimentação, frequência de operação, capacitâncias associadas aos nodos e atividades de comutação dos transistores.

A primeira análise de um circuito inversor (Figura 2.2) considera que todas as capacitâncias estão representadas na saída através de duas capacitâncias equivalentes, C_1 entre o ponto "O" e o terra, e C_2 entre "O" e a tensão de alimentação V_{dd} .

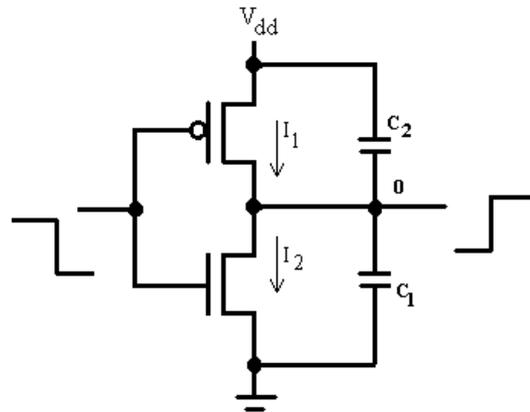


FIGURA 2.2 - Principais capacitâncias de um circuito inversor

Considere também que, inicialmente a entrada está em "1" e a saída em "0". Isto produz uma tensão de saída baixa e uma mudança nas cargas dos capacitores. A carga de C_1 ($C_1 \cdot V_{dd}$) é descarregada para o terra devido ao transistor *nmos* estar em curto circuito (I_2). A carga de C_2 é carregada de "0" para $C_2 \cdot V_{dd}$. Na próxima fase quando a entrada estiver num nível baixo, C_2 estará em curto circuito através do transistor *pmos* e C_1 estará carregando até atingir $C_1 \cdot V_{dd}$. Como resultado, teremos a carga total do circuito representada por $(C_1 + C_2) \cdot V_{dd}$. A energia total neste caso é $(C_1 + C_2) \cdot V_{dd}^2$ e a potência $f_c \cdot (C_1 + C_2) \cdot V_{dd}^2$. As capacitâncias dos nodos na saída é o resultado da soma de $C_1 + C_2$.

A fórmula da potência dinâmica [CIR 87] é representada por:

$$P_{din} = 0.5 f_c V_{dd}^2 \sum_{i=1}^n C_i \alpha_t(x_i) \quad (3)$$

onde:

P_{din} = Potência dinâmica consumida pelo circuito [W]

f_c = frequência de operação [Hz]

V_{dd} = tensão de alimentação do circuito [V]

n = número total de portas do circuito

C_i = capacitância equivalente no nodo x_i [F]

$\alpha_t(x_i)$ = atividade de comutação de cada porta

2.1.4 Consumo de Potência devido aos *Glitching*

Hazards ou *glitching* são transições espúrias e transitórias que surgem em uma porta lógica quando acontece uma transição de sinal num modelo de atraso de propagação diferente de zero [ROY 98]. As contribuições básicas de *hazards* para a potência consumida em circuitos digitais CMOS são dadas pela:

- 1) Geração de *hazard* estático na saída da porta;
- 2) A propagação de um *hazard* através de uma porta.

A Figura 2.3 ilustra o efeito do modelo de atraso na geração de um *glitching* e sua propagação.

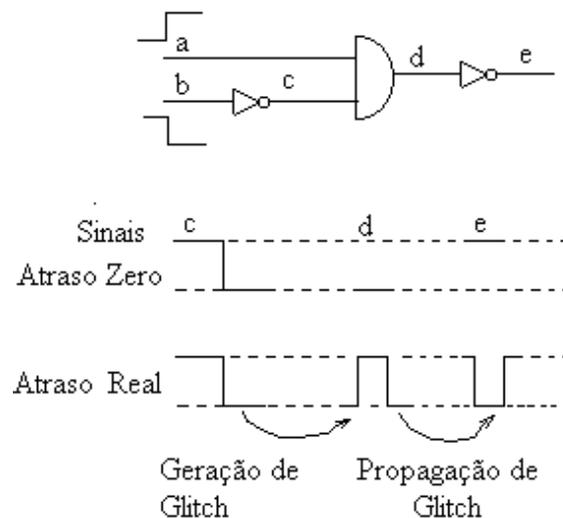


FIGURA 2.3 – Efeito do modelo de atraso na geração e propagação do *glitching*

A potência dissipada devido aos *hazards* pode representar de 15% a 20% da potência total do circuito [BEN 94], dependendo do modelo de atraso adotado na simulação e profundidade lógica. O cálculo ou estimativa das atividades dos *glitching* em um circuito é geralmente muito difícil, pois depende da equalização entre os tempos de subida e descida do sinal nos níveis lógicos internos, o que requer cuidados na caracterização ao nível de circuito ou lógica, bem como conhecimento da estrutura interna do circuito.

2.1.5 Delimitação do Trabalho de estimativa de Potência

Neste trabalho a potência dissipada será estimada considerando que:

- 1) A potência analisada será a dinâmica, ou seja, devida apenas às correntes de carga e descarga dos capacitores;
- 2) As correntes de curto circuito e de fuga serão desprezíveis;
- 3) os circuitos serão combinacionais construídos de portas lógicas CMOS convencionais, ou seja **inversores, nand's, nor's, and's** e **or's**;
- 4) A tensão de alimentação e a frequência de operação serão valores fixos;
- 5) Os *hazards* ou *glitching* serão desprezados, tendo em vista a complexidade no estabelecimento da relação de tempos entre sinais internos do circuito.

2.2 Níveis de Estimativa de Potência

A potência dissipada por um circuito CMOS pode ser estimada em diferentes níveis de abstração. O primeiro compromisso a ser analisado é a complexidade computacional versus precisão. Esta seção tem por objetivo fazer uma revisão dos principais métodos usados na estimativa de potência desde o nível de sistema até o nível de transistor.

2.2.1 Estimativa de Potência ao Nível de Sistema

Neste nível, o leiaute não foi gerado ainda e portanto não existem valores de capacitâncias extraídas. As atividades de comutação usam modelos teóricos baseados na entropia, que será vista na seção 2.2.2, enquanto que as capacitâncias usam modelos de predição de interconexões que se baseiam no comprimento dos fios.

A predição ou estimativa do comprimento das interconexões tem sido estudada em vários contextos, como por exemplo, roteamento de *gate array* [GAM 81], leiaute hierárquico *top down* [DON 79], [FEV 82] e *floorplaning* [HEB 96].

Esta predição pode ser realizada antes, durante ou depois do processo de posicionamento das células na planta baixa, tornando necessário o desenvolvimento de modelos de interconexões. Conforme [PED 99], os modelos de interconexões são divididos em três categorias: empíricos, teóricos e usuais

2.2.1.1 Modelos Empíricos de Interconexões

Os primeiros modelos de interconexões ao nível de sistema baseavam-se em estimativas empíricas e foram desenvolvidos por Rent nos anos 60 [BAK 90]. Em 1971 Landman e Russo [LAN 71] descreveram uma relação entre número médio de terminais (T) de um circuito e número médio de portas lógicas (ou blocos lógicos) do circuito. Esta relação matematicamente é definida por:

$$T = t \cdot B^p \quad (4)$$

Sendo chamada Regra de Rent.

Onde:

T = Quantidade de pinos de entrada e saída

t = Tamanho médio da célula

B = Quantidade de células

p é o expoente de Rent

O valor do expoente de Rent depende da complexidade topológica das interconexões e qualidade do posicionador (*placement*). Quanto mais complexo for o circuito maior é o valor de p , estando limitado a um valor máximo igual a 1 [CHR2000]. A regra de Rent foi usada por [CHR2000] para circuitos homogêneos e [ZAR 98] para heterogêneos, ambos com o propósito de estimar comprimento das interconexões.

Um circuito é dito heterogêneo quando sua arquitetura possui um conjunto de células lógicas com diferentes funções (caso de um microprocessador) enquanto que quando apresenta células com mesma função é dito circuito homogêneo (memória).

A regra de Rent foi usada pela primeira vez em 1979 por Donath [DON 79] com o propósito de estimar o comprimento das interconexões de circuitos. A idéia era simples, ou seja, o circuito foi particionado hierarquicamente em partes iguais, repetidos recursivamente até que todas as portas lógicas estivessem marcadas em uma grade. O número médio de interconexões entre as partes foi então, estimado pela regra de Rent e o comprimento médio estimado em função da distribuição uniforme destas conexões na grade. Baseado nesta abordagem, uma série de somas conduziu ao cálculo do comprimento total das interconexões. A divisão deste somatório pelo número total de interconexões forneceu o comprimento médio padrão (R_m) que foi definido por:

$$R_m = \frac{2}{9} \left(7 \log_4 N_g - \frac{1 - N_g^{p-1.5}}{1 - 4^{p-1.5}} \right) \frac{1 - 4^{p-1}}{1 - N_g^{p-1}} \quad (5)$$

Onde:

R_m é comprimento médio padrão

N_g é o número de portas lógicas

p é o expoente de Rent

E o comprimento médio das interconexões em valores unitários será:

$$l_m = R_m d_g \quad (6)$$

Onde :

l_m é o comprimento médio em valores unitários

d_g é a dimensão da porta lógica em μm .

2.2.1.2 Modelos Teóricos de Interconexões

Os modelos teóricos de interconexões estão baseados em fórmulas matemáticas cujas variáveis derivam de parâmetros extraídos das características físicas dos transistores e das interconexões.

São divididos em duas categorias: determinísticos e estocásticos.

Nos modelos determinísticos os parâmetros de cálculo são extraídos do projeto através de medições e ajuste de curvas.

Nos modelos estocásticos o comprimento das interconexões é calculado através de dados probabilísticos das células, extraídos de uma biblioteca padrão. Estas células são selecionadas aleatoriamente a partir de um modelo estocástico e contém informações probabilísticas de cada célula. [SAS 86] modelou as interconexões como sendo formadas de fios com comprimentos médios, obtidos de fórmulas estatísticas para largura do canal de roteamento, probabilidade de roteamento e comprimento dos fios através da distribuição de Weibull.

[HEL 77] e [KUR 89] apresentam um estimador de área para circuitos com leiautes decorrentes do uso de células padrões (*standard cell*). Assumem que as linhas possuem tamanhos iguais, células com duas entradas, passo de pinos constantes, redes com dois pinos e mínimo caminho de conexão linear. Seus modelos assumem que uma interconexão que começa no *slot* do pino i genérico de comprimento l possui probabilidades de comutação definidas por $p_B(i)$ e $p_L(l)$, cujas variáveis são aleatórias e independentes. Sugerem uma distribuição uniforme para $p_B(i)$ e geométrica para $p_L(l)$. Baseado nestas considerações a área de roteamento é estimada. Este modelo, entretanto requer conhecimento prévio do comprimento médio das interconexões.

2.2.1.3 Modelos Práticos de Interconexões

Modelos práticos e usuais na análise das interconexões são aqueles que trabalham com informações físicas do leiaute das células do circuito e estrutura das interconexões [ARO 96].

Estes modelos são baseados em características de projeto (regras de projeto), combinando-as com abstrações de posicionamento e processos de roteamento. Neste método não considera-se a distribuição do comprimento das interconexões.

Sechem [SEC 87] apresenta um estimador de comprimento de interconexões, o qual assume que as células do circuito assumem um posicionamento quadrado, estando posicionadas em uma grade bidimensional. Os diferentes posicionamentos das células formam uma caixa limitante (*bounding box*) ficando armazenados numa tabela de referência (*look-up table*). Todos os pinos pertencentes à mesma rede são cobertos por esta caixa e o seu semi-perímetro define o comprimento médio de interconexão para a referida rede.

Considere por exemplo, o circuito da Figura 2.4 (a). Supondo que necessita-se posicionar as portas numa superfície bi-dimensional e que cada porta lógica seja representada por retângulos. Estes retângulos deverão ser posicionados no leiaute de forma a minimizar as interligações entre si. O posicionamento dos retângulos na planta baixa do circuito juntamente com o roteamento determinam o comprimento das interconexões do circuito.

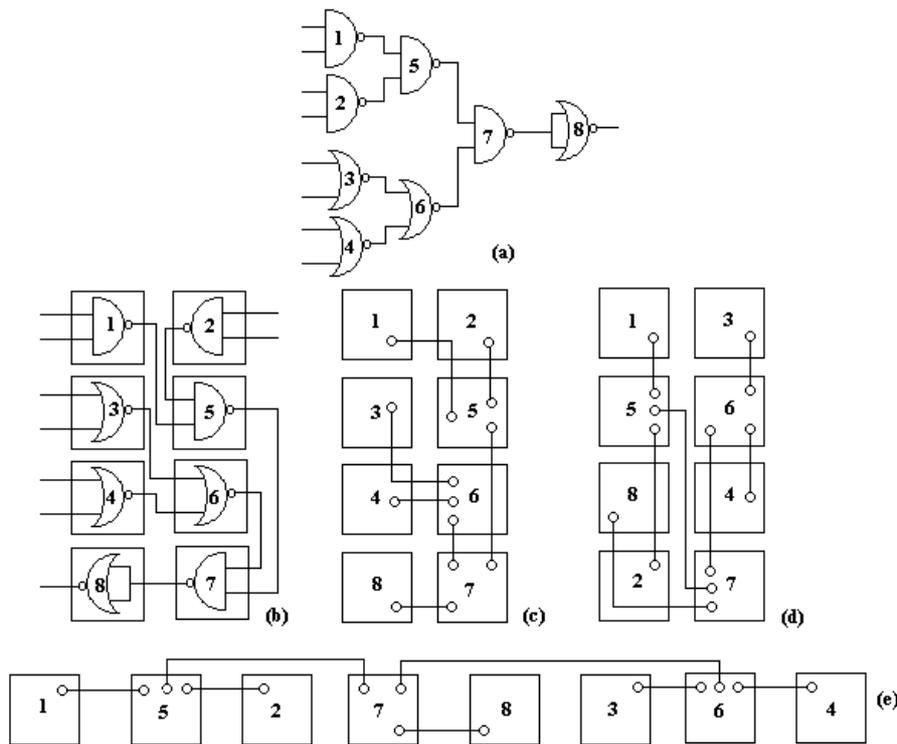


FIGURA 2.4 – Variações do posicionamento das células no *leiaute* de um circuito

(a) – Representação de um circuito no nível lógico

(b),(c),(d) e (e) – Variações do posicionamento das células com diferentes comprimentos de interconexões

Observa-se na Figura 2.4, itens (b), (c), (d) e (e) que para diferentes posicionamentos das células teremos diferentes comprimentos de interconexões. Portanto, o algoritmo de posicionamento de uma ferramenta de CAD é importantíssimo na otimização das interconexões de um circuito.

Chen e Bushell [CHE 88] introduziram um estimador de área com posicionamento aleatório das interconexões, considerando que estas fazem parte das mesmas trilhas. [PED 99] apresenta um modelo que relaciona as propriedades do projeto físico com as características estruturais do projeto lógico, visando estimar precisamente o comprimento das interconexões com leiautes gerados a partir de células padrões. Os resultados da predição são obtidos na análise do *netlist* gerado.

2.2.2 Estimativa de Potência ao Nível de Transferência entre Registradores

Nesta seção o problema de estimar potência está relacionado às informações de entropia do circuito e energia dissipada na transmissão da informação [MAR 96]. Inicialmente, a entropia foi usada em problemas de estimativa de área, análise de *timing* [LAN 95], [CHA 95] e testabilidade [CHA 96].

2.2.2.1 Entropia

Entropia $H_{(x)}$ é um processo que caracteriza a aleatoriedade ou incerteza de uma seqüência de vetores de estímulo aplicados em um circuito. Está relacionada diretamente à atividade de comutação do circuito, ou seja, quanto maior a atividade de comutação maior a entropia. É usada no modelo teórico [COV 91] como medida de capacidade da informação.

Se x é uma variável Booleana randômica com probabilidade p de estar no nível alto, isto é, $P=\{x=1\}=p$, então a entropia H de x é definida como:

$$H_{(x)} = p \cdot \log_2 \frac{1}{p} + (1-p) \cdot \log_2 \frac{1}{(1-p)} \quad (7)$$

onde \log_2 é o logaritmo de base 2 e p a probabilidade do evento ocorrer. Uma plotagem da entropia $H_{(x)}$ versus probabilidade é mostrada na Figura 2.5.

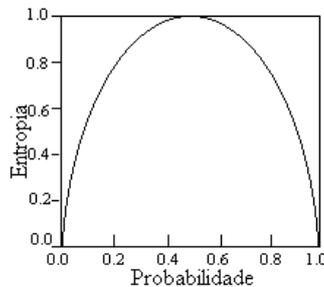


FIGURA 2.5 – Entropia de uma variável Booleana versus probabilidade

A função $H_{(x)}$ tem um valor máximo para probabilidade igual a 0,5, o que significa que um maior número de informações é carregado quando $p = 0,5$. Genericamente, se uma variável discreta possui “ n ” diferentes valores a entropia é igual a:

$$H_{(x)} = \sum_{i=1}^n p_i \cdot \log_2 \frac{1}{p_i} \quad (8)$$

onde p_i é a probabilidade do i -ésimo valor de x .

Então, cada variável Booleana tem uma função de entropia associada, cujo valor é determinado pela probabilidade do valor assinalado pela variável. Seja $Y = f(X)$ uma função Booleana onde X é um vetor Booleano de $n.b$ valores e Y um vetor com $m.b$ valores. Uma função $f(.)$ pode ser implementada por um circuito lógico de n entradas e m saídas. Então, X terá 2^n valores e a entropia de entrada é igual a:

$$H_{(X)} = \sum_{i=1}^{2^n} p_i \cdot \log_2 \frac{1}{p_i} \quad (9)$$

E Y terá 2^m valores e a entropia de saída é igual a:

$$H_{(Y)} = \sum_{i=1}^{2^m} p_i \cdot \log_2 \frac{1}{p_i} \quad (10)$$

A **entropia na saída** de um circuito combinacional é sempre menor ou no máximo igual à **entropia na entrada**, portanto $H(Y) \leq H(X)$ [COV 91].

A entropia está também associada à predição de área de silício de uma função Booleana. Desta forma, estabelece-se a seguinte relação [PIP 77]:

$$A \propto \frac{2^n}{n} H(Y) \quad (11)$$

Para pequenos circuitos com n portas lógicas ($n \leq 10$) observa-se empiricamente que $2^n H(Y)$ fornece uma boa medida de área [CHE 90].

Considere agora um circuito lógico combinacional composto de N portas lógicas CMOS, cujos nodos de saída são denotados por x_i , $i = 1, 2, \dots, N$. Se $D(x)$ é a densidade de transição do nodo, a potência média dissipada no circuito será igual a:

$$P_{din} = \frac{1}{2} V_{dd}^2 \sum_{i=1}^N C_i D(x_i) \quad (12)$$

Onde:

V_{dd} é a tensão de alimentação [V]

N é o número total de nodos do circuito

C_i é a capacitância total no nodo x_i [F]

$D_{(x_i)}$ é a densidade de transição do sinal

Ou ainda:

$$P_{din} \propto \sum_{i=1}^N C_i \cdot D(x_i) \approx D \sum_{i=1}^N C_i \quad (13)$$

onde D é a densidade média de transição no nodo, definida por:

$$D = \frac{1}{N} \sum_{i=1}^N D(x_i) \quad (14)$$

De modo que a potência média será proporcional a:

$$P_{media} \propto Ax D \quad (15)$$

onde A é uma estimativa de área do circuito representada pelo somatório das capacitâncias no nodo :

$$\sum_{i=1}^N C_i \quad (16)$$

O principal resultado deste método de estimativa está na relação entre densidade média de transistores e entropia a eles associada.

2.2.3 Estimativa de Potência ao Nível de Portas Lógicas

A abordagem probabilística ao nível de portas lógicas [CIR 87], [GHO 92] considera a potência dissipada nos nodos internos das portas de um circuito e atividades

de comutação do sinal em cada nodo. Uma vez determinada a atividade de comutação em cada nodo do circuito, a capacitância pode ser estimada como:

$$C_{estimada} = \sum p_{i0}(1 - p_{i0})C_i \quad (17)$$

onde:

p_{i0} é a probabilidade do nodo i estar no estado ZERO e C_i a capacitância de comutação associada a este nodo. A análise de estimativa de potência neste nível de abstração será melhor trabalhado no Capítulo 3.

A capacitância de comutação está diretamente relacionada à capacitância das portas lógicas e às capacitâncias de interconexões. Esta última pode ser estimada no nível lógico como função do comprimento médio das interconexões e posicionamento das células na planta baixa.

2.2.4 Estimativa de Potência ao Nível de Transistores

Ao nível de transistores a potência de um circuito CMOS é calculada utilizando-se da modelagem matemática dos transistores e da extração elétrica das capacitâncias parasitas. Neste nível, é possível estimar não só a potência dinâmica do circuito, mas também a potência de curto-circuito. A ferramenta mais conhecida e confiável que estima potência ao nível de transistores é o SPICE [NAG 75]. Apesar da sua precisão, as suas limitações estão na modelagem de circuitos complexos e combinação dos vetores de entrada, além da necessidade de que seja realizada a etapa da geração do leiaute do circuito e extração elétrica, o que demanda considerável tempo de processamento.

3 Estimativa de Potência ao Nível Lógico

As técnicas atuais de estimativa de potência dinâmica no nível lógico estão baseadas nas atividades de comutação de cada porta do circuito, onde as capacitâncias de comutação são definidas empiricamente.

São classificadas em duas categorias: estatísticas e probabilísticas.

3.1 Técnica Estatística de Estimativa de Potência

O método de Monte Carlo [BUR 93], [NAJ 94] é usado como técnica estatística para estimar potência consumida em circuitos CMOS.

O diagrama de blocos da Figura 3.1 dá uma visão global do método. Os blocos de *setup* e amostragem dos dados fazem parte da execução da simulação lógica, onde os vetores de teste são gerados aleatoriamente. O valor da potência consumida no final da etapa de amostragem, é usado como critério de decisão para interromper o processo ou fazer nova iteração na execução das etapas de *setup* e amostragem. A decisão é tomada baseada no desvio padrão dos valores de potência encontrados após sucessivas iterações.

A potência consumida é o valor médio da potência obtida na etapa de amostragem. A etapa de *setup* é uma componente crítica de nossa abordagem e serve para dois propósitos.

1) No princípio da execução da simulação, o circuito apresenta baixa comutação, que aumenta à medida que todos os nodos são alcançados pelo sinal. Assim o circuito é simulado até que todos os nodos comutem a uma determinada taxa em um certo tempo durante a etapa de *setup*.

2) Os valores de potência observados ao término de intervalos de amostragens sucessivas, são amostras de variáveis aleatórias independentes. Esta consideração é importante na confiabilidade do critério de interrupção da estimativa. Assim, a etapa do *setup* garante medições de potência, assegurando a veracidade do critério estatístico de interrupção.

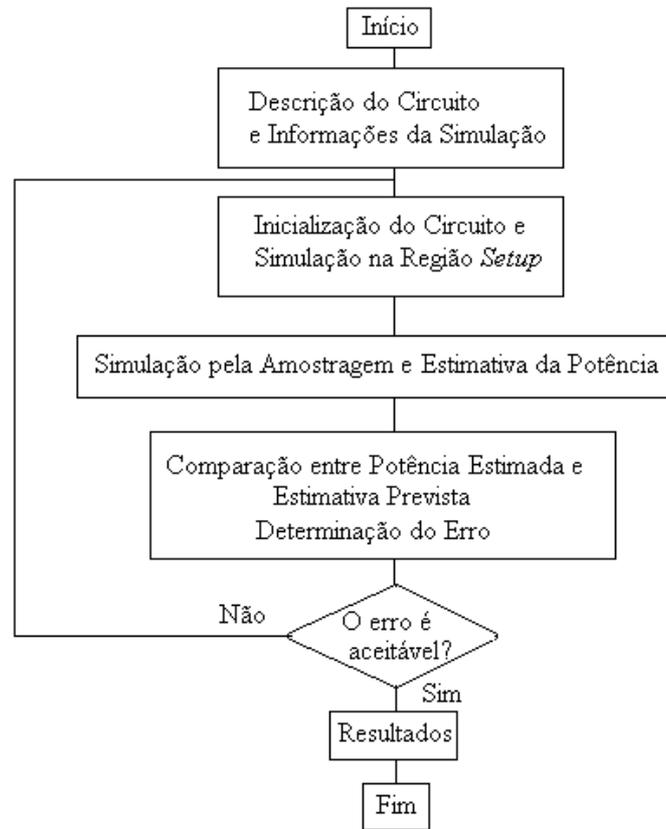


FIGURA 3.1 – Método de Monte Carlo

3.1.1 Formulação do Problema

Considere um circuito digital com m nodos internos, $x_i(t)$ o sinal lógico para o nodo i no instante t e $t \in (-\infty, +\infty)$ e $n_{xi}(T)$ o número de transições de x_i no intervalo de tempo $(-T/2, +T/2]$. Se considerar somente a contribuição da carga e descarga das correntes, a potência média dissipada no nodo i , durante o intervalo acima será igual a:

$$P_i = \frac{1}{2} V_{dd}^2 C_i \frac{n_{xi}(T)}{T} \quad (18)$$

Onde :

P_i = potência média no nodo i

C_i = capacitância associada ao nodo i

V_{dd} = tensão de alimentação do circuito

n_{xi} = número de transições de sinal lógico x aplicado ao nodo i

A potência média dissipada em todo o circuito durante o intervalo de tempo T será igual a :

$$P_T = \frac{V_{dd}^2}{2} \sum_{i=1}^m C_i \frac{n_{xi}(T)}{T} \quad (19)$$

E a potência média dissipada no intervalo de tempo $(-T/2, + T/2]$ será igual a:

$$P = \lim_{T \rightarrow \infty} P_T = \frac{V_{dd}^2}{2} \sum_{i=1}^m C_i \lim_{T \rightarrow \infty} \frac{n_{xi}(T)}{T} \quad (20)$$

3.1.2 Critério de Interrupção

O critério de interrupção do processo de coleta de dados assume algumas considerações:

- Suponha que a potência total (P_T) seja representada por uma curva normal de distribuição durante um tempo, definimos então, o erro percentual desejado de interrupção (ε). Desta forma o circuito é simulado até que a seguinte condição é alcançada:

$$\frac{t_{\alpha/2} s_T}{n_T \sqrt{N}} < \varepsilon \quad (21)$$

onde:

$t_{\alpha/2}$ = valor entre 2,0 e 5,0 obtido de uma distribuição normal t para um nível de confiabilidade igual a $(1-\alpha) \times 100\%$

N = número de simulações

T = larguras de amostragem

n_T = média das amostragens

s_T = desvio padrão entre as amostras

Isolando-se N na equação acima, definimos que o número de simulações desejado para uma interrupção pré-definida será igual a:

$$N \approx \left(\frac{t_{\alpha/2} \cdot s_T}{\varepsilon \cdot n_T} \right)^2 \quad (22)$$

O número de iterações varia de circuito para circuito e depende exclusivamente da complexidade do mesmo. Para circuitos médios o número de iterações é da ordem de milhares, o que demanda um grande tempo para simulação.

3.1.3 Setup e Amostragem

O bloco de *setup* trata da geração dos vetores de entrada, do valor da potência inicial e dos valores da potência limite, as quais vão determinar o ponto de interrupção do processo, bem como os valores de potência obtidos.

Duas condições devem ser observadas nesta fase:

1) Durante todo o intervalo de amostragem, os sinais $x_i(t)$ devem ser amostrados quando o processo estiver em regime permanente.

2) As diferentes amostragens de potência devem feitas através de variáveis aleatórias e independentes.

Suponha que os vetores de entrada do circuito estão num intervalo de tempo de simulação entre 0 e $-\infty$ e que as amostras do processo em regime permanente $x_i(t)$ são obtidos durante o intervalo positivo. Considere uma entrada positiva de um inversor com um atraso (t_d). A partir da condição de que as entradas são um processo permanente para $t \geq 0$, suas saídas deverão estar em regime permanentes, ou seja, $t \geq t_d$. Usando-se um modelo de atraso para cada porta, pode-se repetir este argumento num tempo maior, chegando-se as seguintes conclusões:

- Se o atraso máximo das entradas primárias no nodo i é $T_{max,i}$, então o processo $x_i(t)$ torna-se permanente para $t \geq T_{max,i}$.
- Se o atraso máximo interno do circuito é $T_{max} = \max (T_{max,i})$, então o intervalo de amostragem deve ser iniciado somente após $t \geq T_{max}$. Isto garante a condição anterior.

Para garantir a condição acima, a simulação é reiniciada em cada fase de *setup*. Como resultado, o eixo de tempo é dividido em regiões sucessivas, caracterizando região de *setup* e região de amostragem, conforme ilustrado na Figura 3.2.

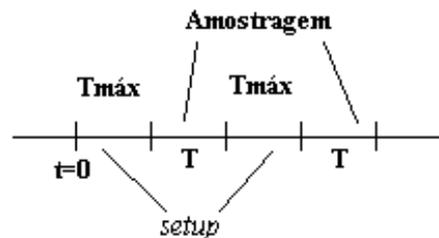


FIGURA 3.2 – Tempos de amostragem e *setup*

Deve-se analisar cuidadosamente as larguras da região de *setup* e região de amostragem T . Para a região de amostragem do sinal, dois fatores devem ser considerados: a aproximação atual e o tempo total de simulação. A largura da região de amostragem afeta o erro da aproximação normal. O valor mínimo de T que permite a aproximação da potência normal é fortemente dependente do circuito e pode variar de zero a infinito.

Minimizar tempo de simulação significa minimizar $N_x(T_{max}+T)$. N_x é um fator que depende de s_T e decresce com o aumento de T . A velocidade da redução do tempo de simulação depende do tamanho e complexidade do circuito em análise. Estes dois fatores indicam uma solução ótima para T , embora não sejam genéricos.

3.2 Técnicas Probabilísticas

As **técnicas probabilísticas** [MON 97] tratam das probabilidades de comutação dos sinais nos nodos de entrada(s) e saída de portas lógicas. Requerem do usuário especificações comportamentais típicas das entradas do circuito. O método é baseado na correlação temporal e espacial dos sinais de entrada e sua precisão é limitada pela qualidade dos modelos de atrasos e especificações das entradas.

3.2.1 Simulação Probabilística

O método de simulação probabilística [NAJ 93] permite ao usuário a especificação típica do comportamento do sinal para as entradas do circuito através da probabilidade de comutação aplicada à forma de onda. A probabilidade da forma de onda é definida como uma seqüência de valores que indica a probabilidade com que o sinal está no nível lógico “1” em um intervalo de tempo, e a probabilidade com que faz as transições entre “0” e “1”. Isto possibilita, calcular as correntes médias drenadas pelas portas, representadas através de uma proporcionalidade com as atividades de comutação e tempo de propagação do sinal (Figura 3.3). Esta forma de onda é usada para computar a potência média dissipada em cada porta e a potência média de todo o circuito.

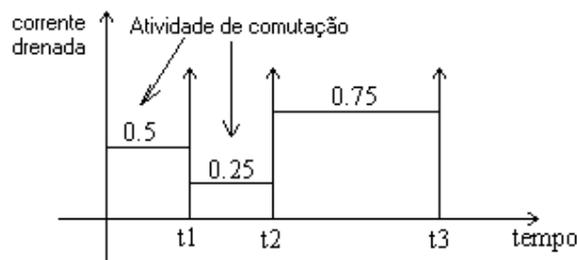


FIGURA 3.3 - Atividade de comutação de um circuito X tempo de propagação

3.2.2 Densidade de Transição

O número médio de transições de um sinal por segundo em relação a um nodo da porta é chamado de densidade de transição [NAJ 93]. Considere y uma função Booleana, dependente de x , então a derivada parcial Booleana de y em relação a x é:

Por definição:

$$\left(\frac{\partial y}{\partial x}\right) = y|_{x=1} \oplus y|_{x=0} \quad (23)$$

onde \oplus é a função OR EXCLUSIVO.

A densidade de transição da saída y é definida como sendo:

$$D(y) = \sum P\left(\frac{\partial y}{\partial x}\right).D(x_i) \quad (24)$$

Como exemplo, considere o caso de uma porta lógica AND de 2 entradas:

$$y = x_1 \cdot x_2 \quad (25)$$

$$\left(\frac{\partial y}{\partial x_1}\right) = x_2 \quad (26)$$

$$\left(\frac{\partial y}{\partial x_2} \right) = x_1 \quad (27)$$

logo :

$$D(y) = P(x_2).D(x_1) + P(x_1).D(x_2) \quad (28)$$

Onde $D(y)$ é a densidade de transição da saída y , $P(x_2)$ a probabilidade da entrada x_2 variar, $D(x_1)$ a densidade do sinal associado a entrada x_1 , $P(x_1)$ a probabilidade da entrada x_1 variar e $D(x_2)$ a densidade do sinal associado a entrada x_2 .

3.2.3 BDD associado a probabilidade de comutação

Uma maneira de representar circuitos CMOS é através dos Diagramas de Decisões Binárias (BDD) [DEV 94], [MIC 94]. A técnica consiste em usar correlação temporal e espacial, juntamente com a representação do circuito lógico na forma de BDD's em cada nodo. A topologia do circuito define uma função Booleana correspondente a cada nodo e expressa o estado permanente em relação às entradas primárias em termos de probabilidades de comutação. Por exemplo, a porta **and** de duas entradas (Figura 3.4) pode ser representada em termos de BDD e probabilidades de comutação.

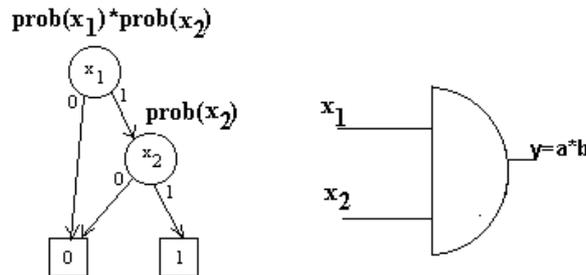


FIGURA 3.4 - BDD da função AND2 associado as Atividades de Comutação

3.2.4 Coeficiente de Correlação Temporal e Espacial

As técnicas probabilísticas usam propriedades estatísticas nas entradas do circuito e propriedades estruturais e funcionais da rede, calculando as atividades de comutação em cada nodo. Métodos probabilísticos na estimativa das atividades de comutação em cada nodo do circuito, envolvem estimativa da probabilidade do sinal variar do nível lógico “1” para o lógico “0”, ou vice-versa.

A correlação temporal (Figura 3.5) é aquela que dois vetores de entrada de um circuito estão relacionados em instantes diferentes. As probabilidades de transição são propagadas e computadas em todos os nodos internos do circuito. Representamos estas probabilidades como $\text{prob}^{01}(x)$, $\text{prob}^{10}(x)$, $\text{prob}^{00}(x)$, $\text{prob}^{11}(x)$, respectivamente. Teremos sempre $\text{prob}^{01}(x) = \text{prob}^{10}(x)$. Em geral, o método probabilístico ignora este tipo de correlação.

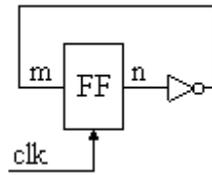


FIGURA 3.5 – Correlação Temporal

A correlação espacial (Figura 3.6) é aquela em que dois ou mais vetores de entrada estão relacionados entre si na saída do circuito, no mesmo espaço, podendo assumir o valor lógico “1” mas não independentes.

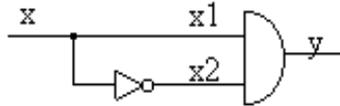


FIGURA 3.6 - Correlação Espacial

3.2.5 Estimativa Simbólica

Inicialmente desenvolvido para circuitos combinacionais [GHO 92], representa as portas internas de um circuito através de um conjunto de funções booleanas. Um vetor de entradas é fornecido, as probabilidades de cada porta comutar são calculadas e no final é feito o somatório das probabilidades das portas correspondentes a um ciclo de relógio. [MON 97] define rede simbólica como sendo um circuito lógico cujos valores de entradas colocados na porta da rede original assumem em diferentes instantes de tempo um dado par de vetores de entrada.

A simulação simbólica trabalha com os seguintes modelos de atrasos: atraso zero, atraso unitário e atraso genérico.

3.2.5.1 Atraso Zero

Considere o circuito mostrado na Figura 3.7 e assumamos que as transições nas portas ocorram instantaneamente, sem nenhum atraso na propagação do sinal.

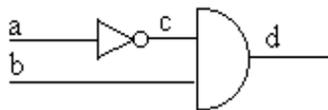


FIGURA 3.7 - Exemplo de circuito para simulação simbólica

Para o modelo de atraso zero, a rede simbólica da Figura 3.7, corresponde à duplicação do circuito original (Figura 3.8), onde uma cópia do circuito recebe o primeiro vetor de entradas e a outra cópia o segundo. A avaliação das transições é realizada por uma porta XOR, cujas entradas são os nodos correspondentes do circuito original, com a diferença de que sua avaliação ocorre em instantes diferentes.

Sabe-se que uma porta XOR só assume o valor 1 na saída, quando nas entradas houver 0 e 1. Portanto, isto significa que quando a saída da porta XOR estiver setada em 1 ocorreu uma transição do sinal naquele nodo. Deve-se então, construir o BDD da função para esta situação e calcular as probabilidades de transição para este nodo. As variáveis deste BDD são as probabilidades de transições.

Em cada nodo que acontece uma comutação, é adotado o mesmo procedimento. A atividade de comutação total do circuito será a soma do comutação individual de cada nodo. Na Figura 3.8, as entradas $a_{(0)}$ e $b_{(0)}$ correspondem ao primeiro vetor de entradas e $a_{(t)}$ e $b_{(t)}$ ao segundo. Se a saída e_c avalia para 1, então o sinal c do circuito original realiza uma transição para aquele par de vetores aplicados. Similarmente a mesma análise é feita para as saídas e_a , e_b e e_d .

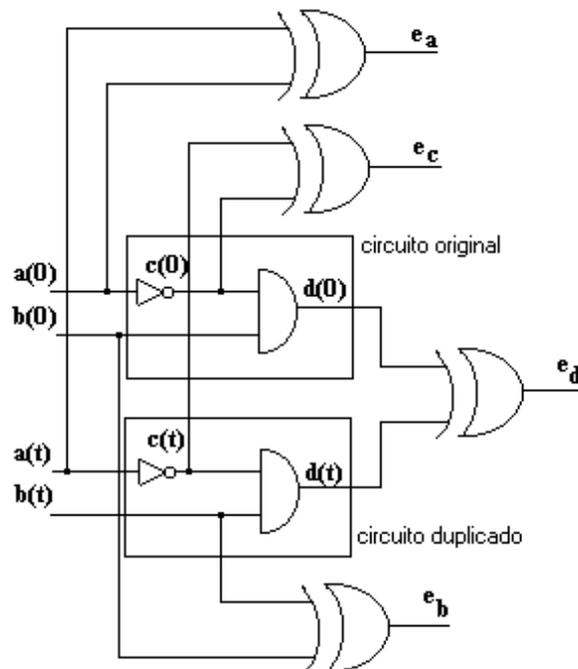
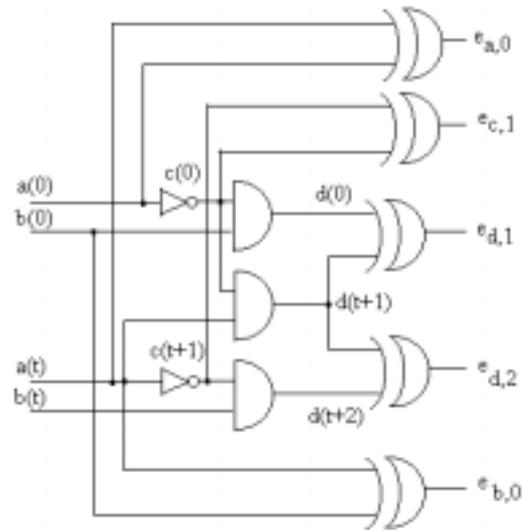


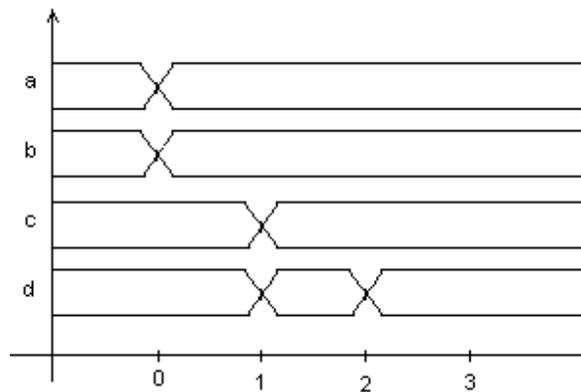
FIGURA 3.8 - Rede simbólica para modelo de atraso zero

3.2.5.2 Atraso Unitário e Genérico

Considere o caso onde as transições não acontecem instantaneamente, ou seja, existe um atraso originado pelas capacitâncias intrínsecas dos transistores das portas. Neste caso teremos múltiplas transições em resposta à seqüência de vetores de entrada. A Figura 3.9(a) apresenta uma rede simbólica com modelo de atraso unitário.



(a)



(b)

FIGURA 3.9 - Rede simbólica com modelo de atraso unitário.

(a) Representação Lógica

(b) Sinais de entrada

$c_{(0)}$ e $d_{(0)}$ são valores iniciais dos nodos c e d , respectivamente. Analisando a Figura 3.9 (b), observa-se que no instante 1, o nodo c terá o valor $c_{(t+1)}$ e d o valor $d_{(t+1)}$. Logo, $e_{c,1} = c_{(0)} \oplus c_{(t+1)}$ avaliando para 1 somente se o nodo c realiza uma transição no instante 1. Similarmente a mesma análise é feita para o nodo d no instante 1. No instante 2, o nodo d assumirá o valor $d_{(t+2)}$. Novamente, $e_{d,2} = d_{(t+1)} \oplus d_{(t+2)}$ dá a condição para d comutar no instante 2. A comutação total na saída da porta d será a soma $e_{d,1}$ e $e_{d,2}$.

Este método modela os *glitching* e os BDD's em termos probabilísticos, cuja correlação espacial está implícita no modelo.

4 Estimativa de Capacitâncias e Potência Dinâmica

4.1 Introdução

Conforme visto nos capítulos anteriores, a capacitância é um dos principais fatores no consumo de potência em circuitos CMOS. A extração de capacitâncias ao nível de leiaute fornece valores precisos, no entanto o tempo de processamento das ferramentas que executam este tipo de tarefa é elevadíssimo para circuitos muito complexos o que pode onerar significativamente o projeto de um circuito integrado.

A vantagem na estimativa de capacitância e potência no nível lógico está no tempo de processamento, contrapondo a precisão dos resultados. Mas, o mais significativo é que este tipo de análise permite tomada de decisões ao nível de circuito lógico, sem a necessidade de ir até a síntese do leiaute e voltar ao nível lógico para modificações.

O capítulo 2 considerou para análise da potência, apenas a capacitância de carga concentrada no **nodo de saída** de uma porta lógica. A estimativa de potência no nível lógico considera que em cada nodo externo de uma porta existe uma capacitância equivalente e uma atividade de comutação associada. Portanto, neste nível faz-se necessário o dimensionamento de capacitâncias equivalentes e concentradas em todos os nodos externos às portas lógicas do circuito.

Este trabalho tem por objetivo estimar potência dinâmica no nível lógico. Nesta estimativa a tensão de alimentação, frequência de operação e atividades de comutações são valores fornecidos.

A potência dinâmica dissipada em um circuito CMOS no nível lógico é formada por duas componentes:

potência de porta que considera apenas as dimensões dos transistores, como W (largura do canal), L (comprimento do canal), perímetro e área do dreno (p_d e a_d) e perímetro e área da fonte (p_s e a_s). Neste tipo de potência as capacitâncias envolvidas são as intrínsecas aos próprios transistores, ou seja: capacitância *gate-dreno*, capacitância *gate-fonte*, capacitância *dreno-substrato* e capacitância *fonte-substrato*.

potência das interconexões que leva em consideração a fiação de roteamento entre as células do circuito. Dois tipos de capacitâncias parasitas são envolvidas nesta análise. A primeira é a capacitância de roteamento propriamente dita que surge entre os nodos dos transistores e o terra. A segunda, é a capacitância de *crossstalk* entre fiações de roteamento. A nossa análise considera apenas **capacitância de roteamento propriamente dita**.

O trabalho desenvolvido nesta Tese está dividido em duas partes:

- A primeira trata de estimar capacitâncias e potência de porta em função das combinações dos vetores de entrada;
- a segunda estima capacitância de roteamento e potência das interconexões a partir da estimativa do comprimento médio das interconexões.

Os dados necessários na modelagem das capacitâncias equivalentes de porta e de interconexões são obtidos pelo fluxograma apresentado na Figura 4.1.

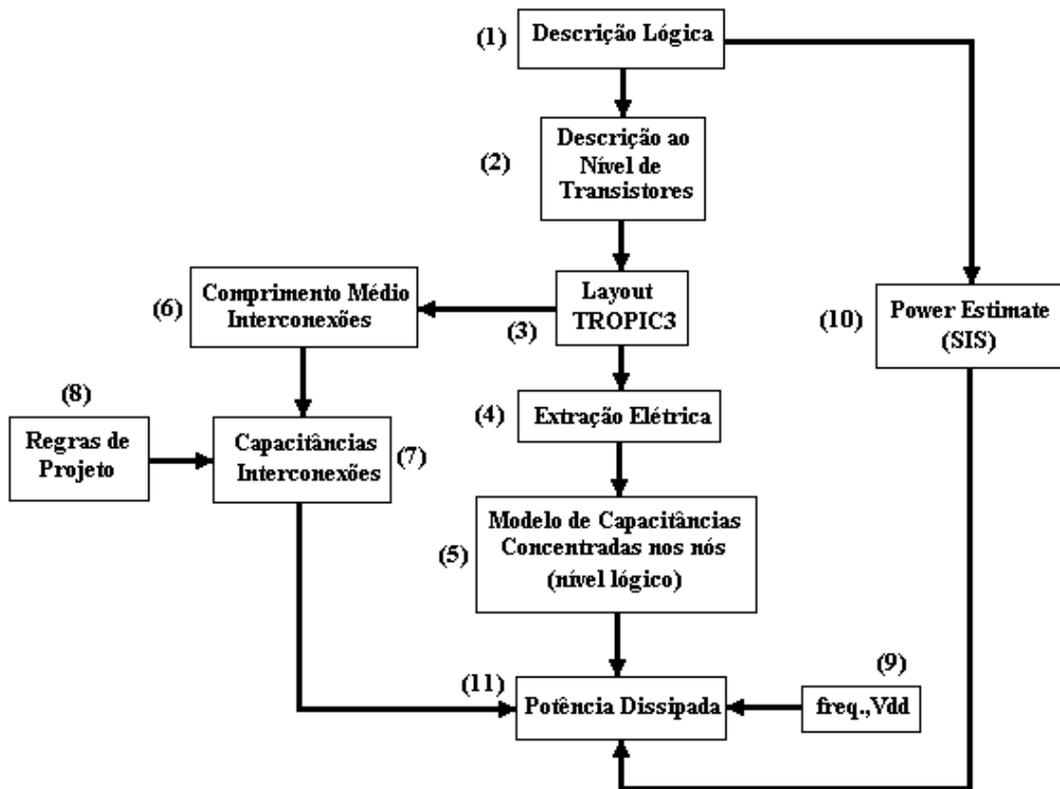


FIGURA 4.1 – Fluxograma do método proposto para estimativa da potência

O circuito é descrito no formato de portas lógicas (BLIF) (1) sendo convertido para o formato SPICE estruturado (SIM) (2). A ferramenta TROPIC através deste formato, realiza o posicionamento e o roteamento das células, gerando o leiaute do circuito (3). Esta ferramenta extrai também informações dos comprimentos das redes entre células (6) e das suas capacitâncias de roteamento, e fornece dados construtivos dos transistores como área do dreno e fonte (A_D , A_S) e perímetro do dreno e fonte (P_D , P_S), além da largura e comprimento do canal (L , W) (4).

A partir dos dados dos transistores as capacitâncias intrínsecas são modeladas no nível lógico (5), enquanto que as informações dos comprimentos das redes permite a estimativa das capacitâncias de cada rede (7). A modelagem das redes necessita ainda das regras de projeto (8) para a tecnologia em uso.

De posse das capacitâncias modeladas para cada nodo, bem como atividade de comutação obtida através da ferramenta (*Power Estimate*) (10), frequência de operação e tensão de alimentação (9), as potências em cada nodo são obtidas (11).

4.2 Ferramenta Auxiliar na Validação do Método Proposto

A validação de nossa proposta de estimativa de consumo ao nível lógico passa pela comparação das estimativas com valores medidos pós leiaute. A geração de leiautes no estudo comparativo é efetuada com a ferramenta TROPIC. Esta seção trata de uma breve descrição desta ferramenta.

4.2.1 Descrição da Ferramenta TROPIC

TROPIC é a uma ferramenta de síntese automática de leiaute de circuitos integrados, que utiliza-se de bibliotecas virtuais, permitindo a síntese de células complexas, cujo método empregado é o de macro-células, onde todas as células do circuito são sintetizadas em bandas, sendo o roteamento realizado entre as difusões N e P dos transistores.

A versão TROPIC2 [MOR 93] gera um leiaute simbólico sem regras de projeto definidas e sem extração de capacitâncias e transistores. A extração de capacitâncias neste caso, só será possível utilizando-se de outras ferramentas que fazem a compactação e a extração elétrica, como por exemplo CADENCE.

A versão TROPIC3 [MOR 99] gera o leiaute completo do circuito sem necessidade de compactação. Neste caso a extração de capacitâncias é realizada pela ferramenta EXT, a qual encontra-se dentro do mesmo pacote do TROPIC3. A ferramenta LASCA [FER2000] extrai capacitâncias de roteamento e de *crosstalk*.

Os transistores em ambas as versões, são implementados horizontalmente, paralelos às linhas de alimentação, conforme Figura 4.2. A técnica de posicionamento dos transistores é denominada matriz linear.

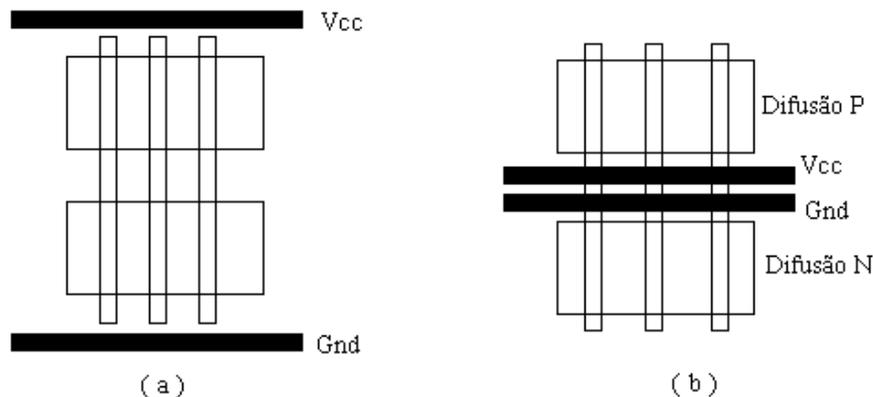


FIGURA 4.2 – Implementação do método matriz linear nas versões do TROPIC

(a) Linhas externas de alimentação – TROPIC2

(b) Linhas internas de alimentação entre os transistores-TROPIC3

Duas alternativas de implementação do estilo de matriz linear são possíveis:

A primeira alternativa (Figura 4.2.a) adotada no TROPIC2 apresenta as linhas de alimentação externa aos transistores e células com altura constante. O problema desta implementação ocorre quando o roteamento é feito entre os transistores NMOS e PMOS, obrigando o seu afastamento dos planos N e P, com um conseqüente aumento no comprimento do polisilício vertical que une os transistores. Esse aumento no comprimento do polisilício introduz capacitâncias e resistências parasitas que degradam o desempenho elétrico do circuito, além de uma perda grande de área.

A segunda alternativa adotada no TROPIC3, com linhas de alimentação entre os transistores (Figura 4.2.b), permite manter o comprimento fixo nas linhas verticais de polisilício proporcional, com um afastamento entre os planos de difusão N e difusão P definido pelas regras de projeto.

5 Estimativa de Potência em Portas Padrões

5.1 Introdução

O objetivo deste capítulo é estimar o consumo de potência dinâmica de porta no nível lógico nas portas lógicas de circuitos combinacionais CMOS. Este modelo é baseado na combinação de vetores de entrada e leva em consideração apenas as dimensões dos transistores (pd, ad, ps e as) e dados da tecnologia envolvida no processo de fabricação (W e L). As capacitâncias parasitas são estimadas e concentradas nos nodos externos das portas lógicas do circuito [MAR2000a], [MAR2000b].

O modelo será validado usando-se a estimativa de potência no nível de transistores. A ferramenta SPICE [NAG 75] será usada para validar o modelo desenvolvido. Os dados referentes às dimensões dos transistores são obtidos da extração elétrica após a geração do leiaute do circuito e dependem da tecnologia de fabricação e da maneira como é realizado o roteamento das células do circuito.

5.2 Capacitâncias Intrínsecas

Um transistor tipo MOSFET (Figura 5.1) possui diversas capacitâncias parasitas que são intrínsecas aos materiais de fabricação. As principais capacitâncias são: capacitância *gate-fonte* (C_{GS}), capacitância *gate-dreno* (C_{GD}), capacitância *gate-substrato* (C_{GB}), capacitância *fonte-substrato* (C_{SB}) e capacitância *dreno-substrato* (C_{DB}).

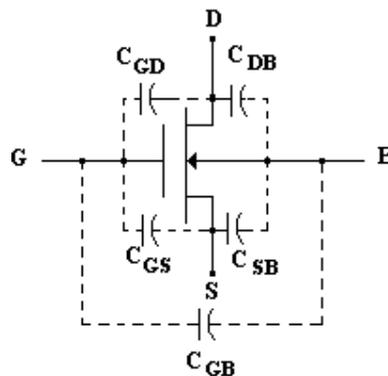


FIGURA 5.1 – Capacitâncias Parasitas Intrínsecas de um transistor MOSFET

Estas capacitâncias são função da tensão *gate-fonte* (V_{GS}) aplicada nos transistores. Devido a sua não linearidade, seus valores também serão não lineares. Neste trabalho estas capacitâncias parasitas serão linearizadas através do uso das seguintes fórmulas [UYE 92]:

$$C_{GDn(p)} = C_{GSn(p)} = \frac{1}{2} C_{ox} * Wn(p) * Ln(p) \quad (29)$$

$$C_{DBn(p)} = K_{1/2}(V_1, V_2)C_{j0}A_{Dn(p)} + K_{1/3}(V_1, V_2)C_{jsw}l_{Dn(p)} \quad (30)$$

$$C_{SBn(p)} = K_{1/2}(V_1, V_2)C_{j0}A_{Sn(p)} + K_{1/3}(V_1, V_2)C_{jsw}l_{Sn(p)} \quad (31)$$

$$K_{1/3}(V_1, V_2) = \frac{3C_{j0}V_{bi}}{2(V_2 - V_1)} \left[\left(1 + \frac{V_2}{V_{bi}}\right)^{2/3} - \left(1 + \frac{V_1}{V_{bi}}\right)^{2/3} \right] \quad (32)$$

$$K_{1/2}(V_1, V_2) = \frac{2C_{j0}V_{bi}}{(V_2 - V_1)} \left[\left(1 + \frac{V_2}{V_{bi}}\right)^{1/2} - \left(1 + \frac{V_1}{V_{bi}}\right)^{1/2} \right] \quad (33)$$

onde:

$C_{DBn(p)}$ = capacitância dreno-substrato para o transistor *nmos* (*pmos*) [F]

$C_{SBn(p)}$ = capacitância fonte-substrato para o transistor *nmos* (*pmos*) [F]

$C_{GDn(p)}$ = capacitância *gate*-dreno para o transistor *nmos* (*pmos*) [F]

$Ln(p)$ = comprimento do canal do transistor *nmos* (*pmos*) [m]

$Wn(p)$ = largura do canal do transistor *nmos* (*pmos*) [m]

C_{j0} = capacitância do transistor por unidade de área [F/m²]

C_{jsw} = capacitância lateral do transistor por unidade de perímetro [F/m]

$A_{Dn(p)}$ = Área do dreno do transistor *nmos*(*pmos*) [m²]

$A_{Sn(p)}$ = Área da fonte do transistor *nmos*(*pmos*) [m²]

$l_{Dn(p)}$ = Perímetro do dreno do transistor *nmos* (*pmos*) [m]

$l_{Sn(p)}$ = Perímetro da fonte do transistor *nmos* (*pmos*) [m]

V_{bi} = Tensão de junção [V]

C_{ox} = Capacitância do óxido da porta [F]

5.3 Estimativa de Capacitância de Porta

As capacitâncias consideradas no nível lógico de um circuito são aquelas concentradas nos nodos de entrada e saída de uma porta lógica referenciadas ao terra. Com exceção do inversor, as demais portas lógicas convencionais (**and**, **or**, **nand** e **nor**) possuem nodos internos. Estas portas lógicas possuem capacitâncias associadas cujo efeito depende da combinação de vetores aplicados nas entradas.

Nesta seção, descreveremos um método matemático que estima a capacitância equivalente em cada nodo externo da porta lógica. Isto permite analisar o efeito das capacitâncias internas sobre a capacitância equivalente. Serão analisadas as portas convencionais **inversor**, **nand2**, **nand3**, **nand** de “*n*” entradas. A mesma análise pode ser feita para portas NOR’s, pois as mesmas são duais.

5.3.1 Capacitâncias do INVERSOR CMOS

O inversor CMOS (Figura 5.2) apresenta quatro nodos externos: entrada do sinal (x), saída do sinal (y), alimentação (V_{dd}) e terra.

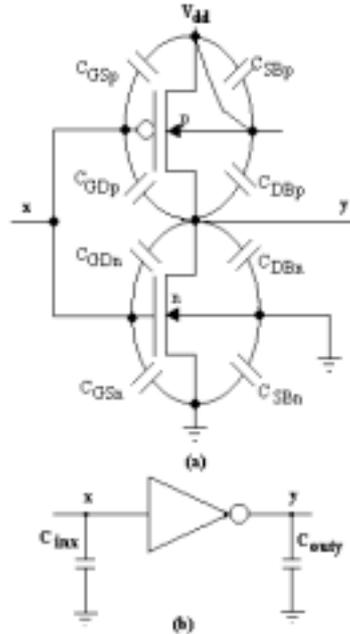


FIGURA 5.2 – Capacitâncias Parasitas Intrínsecas do Inversor CMOS

(a) Representação ao nível de Transistores

(b) Representação ao nível de Portas Lógicas

Usando-se o Teorema de Thévenin da análise de circuitos, pode-se dizer que as capacitâncias equivalentes para a entrada do sinal x (C_{inx}) e saída y (C_{outy}) são iguais a:

$$C_{inx} = C_{GSp} + C_{GDp} + C_{GSn} + C_{GDn} \quad (34)$$

$$C_{outy} = C_{DBp} + C_{GDp} + C_{GDn} + C_{DBn} \quad (35)$$

onde:

C_{GDp} , C_{GSp} e C_{DBp} = capacitância *gate-dreno*, capacitância *gate-fonte* e capacitância *dreno-substrato* para o transistor *pmos*, respectivamente.

C_{GDn} , C_{GSn} e C_{DBn} = capacitância *gate-dreno*, capacitância *gate-fonte* e capacitância *dreno-substrato* para o transistor *nmos*, respectivamente.

5.3.2 Capacitâncias da Porta Lógica NAND de duas entradas

No caso de uma porta lógica **nand** de duas entradas (Figura 5.3), o número de nodos aumenta para seis, sendo eles: um ligado ao terra, um ligado à alimentação V_{dd} , um nodo interno entre os transistores *nmos* e três externos (dois ligados à entrada e um à saída). A influência deste nodo interno sobre os demais nodos externos normalmente não é analisada por outros modelos, mas é considerando em nosso trabalho.

As capacitâncias equivalentes aos nodos externos da porta, correspondem ao efeito da manifestação das capacitâncias intrínsecas dos transistores em função dos vetores de entrada. As capacitâncias dos nodos internos dependem das combinações de vetores de entrada. No caso da **nand2**, estas combinações serão iguais a : 00, 01, 10, 11. A Figura 5.3.a ilustra uma porta **nand2** ao nível de transistores com suas respectivas capacitâncias intrínsecas e a Figura 5.3.b, a sua representação lógica equivalente e capacitâncias equivalentes. As capacitâncias intrínsecas foram definidas na seção 5.2 e dependem das dimensões físicas dos transistores e características da tecnologia do processo de fabricação. As análises a seguir, consideram probabilidades de comutação nas entradas das portas lógicas iguais a 0,5 .

A terminologia adotada na análise das capacitâncias intrínsecas são as seguintes:

$C_{GDn1 (GDn2)}$ = capacitância *gate-dreno* para os transistores mn1 e mn2 respectivamente.

$C_{GSn1 (GSn2)}$ = capacitância *gate-fonte* para os transistores mn1 e mn2 respectivamente.

$C_{DBn1 (DBn2)}$ = capacitância *dreno-substrato* para os transistores mn1 e mn2 respectivamente.

$C_{SBn1 (SBn2)}$ = capacitância *fonte-substrato* para os transistores mn1 e mn2 respectivamente.

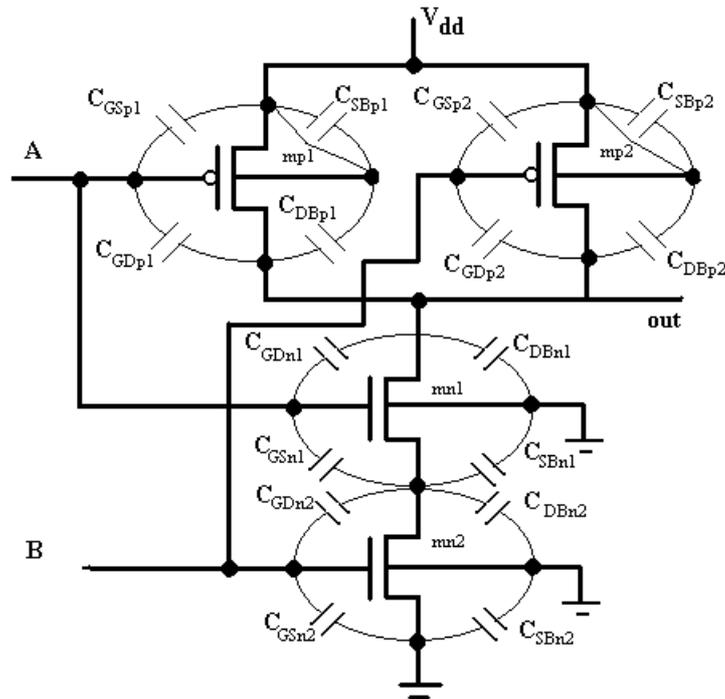
C_{GDp} , C_{GSp} , C_{DBp} e C_{SBp} = capacitância *gate-dreno*, capacitância *gate-fonte*, capacitância *dreno-substrato* e capacitância *fonte-substrato* para transistores *pmos*, respectivamente.

C_A = capacitância de porta equivalente à entrada A para a rede de transistores *nmos*.

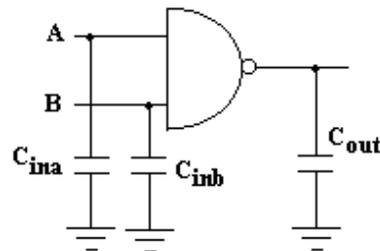
C_B = capacitância de porta equivalente à entrada B para a rede de transistores *nmos*.

C_{ina} = Capacitância total de porta da entrada A considerando as redes de transistores *nmos* e *pmos*.

C_{inb} = Capacitância total de porta da entrada B considerando as redes de transistores *nmos* e *pmos*.



(a)



(b)

FIGURA 5.3 – Capacitâncias Parasitas da porta lógica NAND2 CMOS

(a) Representação ao nível de Transistores

(b) Representação no nível lógico

Capacitância equivalente de porta referente à entrada “A”

Pela Figura 5.3.a, observa-se que as combinações de entradas **00** e **10** faz com o transistor **mn2** esteja na situação de corte e a capacitância equivalente C_{a1} para a rede de transistores **nmos** referente a entrada “A” é igual a:

$$C_{a1} = 0.5 \left(C_{GDn1} + \frac{C_{GSn1} * C_T}{C_{GSn1} + C_T} \right) \quad (36)$$

Para as combinações **01** e **11** o mesmo transistor mn2 estará na situação de saturação e a capacitância equivalente para a rede de transistores *nmos* C_{a2} é igual a:

$$C_{a2} = 0.5(C_{GDn1} + C_{GSn1}) \quad (37)$$

Como resultado, as duas situações deverão ser consideradas, portanto:

$$C_A = C_{a1} + C_{a2} = 0.5(C_{GDn1} + \frac{C_{GSn1} * C_T}{C_{GSn1} + C_T}) + 0.5(C_{GDn1} + C_{GSn1}) \quad (38)$$

onde C_T é igual a:

$$C_T = C_{SBn1} + C_{DBn2} + C_{GDn2} \quad (39)$$

A capacitância total equivalente da entrada “A” (C_{ina}) considerando as redes *pmos* e *nmos* será igual a:

$$C_{ina} = C_A + C_{GSp1} + C_{GDP1} \quad (40)$$

Capacitância equivalente de porta referente à entrada “B”

A mesma análise é feita para a segunda entrada, ou seja:

Para as combinações de entradas **00** e **01** o transistor mn1 está na situação de corte e a capacitância equivalente C_{b1} é igual a:

$$C_{b1} = 0.5(C_{GSn2} + \frac{C_{GDn2} * C_{T1}}{C_{GDn2} + C_{T1}}) \quad (41)$$

Para as combinações **10** e **11** o transistor mn1 está na situação de saturação e a capacitância equivalente C_{b2} é igual a:

$$C_{b2} = 0.5(C_{GDn2} + C_{GSn2}) \quad (42)$$

Como resultado as situações devem ser consideradas, ou seja:

$$C_B = C_{b1} + C_{b2} = 0.5(C_{GDn2} + \frac{C_{GSn2} * C_{T1}}{C_{GSn2} + C_{T1}}) + 0.5(C_{GDn2} + C_{GSn2}) \quad (43)$$

onde C_{T1} é igual a:

$$C_{T1} = C_{GSn1} + C_{SBn1} + C_{DBn2} \quad (44)$$

E a capacitância equivalente considerando as redes de transistores *pmos* e *nmos* é igual a:

$$C_{inb} = C_B + C_{GSp2} + C_{GDp2} \quad (45)$$

Capacitância equivalente referente à saída da porta

Olhando-se na saída da porta lógica em direção à entrada e aplicando-se novamente o Teorema de Thévenin, observa-se que as capacitâncias intrínsecas C_{GDn1} , C_{DBn1} , C_{GDp1} , C_{GDp2} , C_{DBp1} e C_{DBp2} estão em paralelo e portanto a capacitância equivalente ao nodo de saída será igual a:

$$C_{out} = C_{GDn1} + C_{DBn1} + C_{GDp1} + C_{GDp2} + C_{DBp1} + C_{DBp2} \quad (46)$$

onde:

C_{out} = Capacitância equivalente concentrada no nodo de saída.

C_{GDn1} = Capacitância *gate-dreno* do transistor *nmos* ligado à saída.

C_{DBn1} = Capacitância *dreno-substrato* do transistor *nmos* ligado à saída.

C_{GDp1} , C_{GDp2} = Capacitância *gate-dreno* dos transistores *mp1* e *mp2* (*pmos*) ligados à saída, respectivamente.

C_{DBp1} , C_{DBp2} = Capacitância *dreno-substrato* dos transistores *mp1* e *mp2* (*pmos*) ligados à saída, respectivamente.

5.3.3 Capacitâncias da Porta Lógica NAND de três entradas

A porta **nand3** possui oito nodos, sendo seis externos e dois internos.

Os nodos externos referem-se a alimentação V_{dd} , ao terra, as três entradas (A, B e C) e a saída (out). Os nodos internos estão entre os transistores *mn1* e *mn2* e *mn2* e *mn3* (Figura 5.4). Oito combinações possíveis de entradas serão consideradas: 000, 001, 010, 011, 100, 101, 110 e 111. As considerações feitas na seção anterior que analisou a **nand2**, são feitas aqui desde o uso do Teorema de Thévenin até as considerações de probabilidade de comutação nas entradas da porta iguais a 0,5. A Figura 5.4.a, apresenta a porta **nand3** ao nível de transistores com as suas capacitâncias intrínsecas e a Figura 5.4.b a mesma porta ao nível lógico e suas capacitâncias equivalentes.

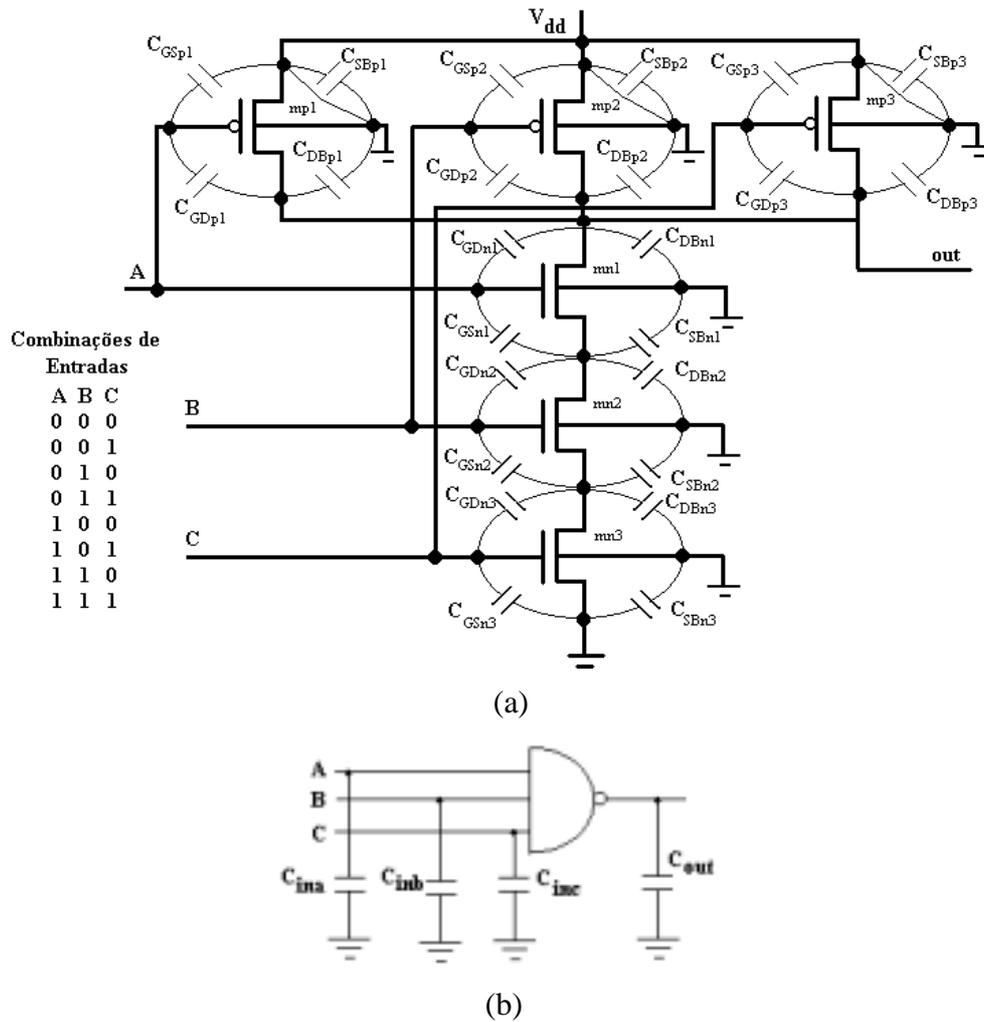


FIGURA 5.4 – Capacitâncias Parasitas da porta lógica NAND3 CMOS

(a) Representação da porta NAND3 ao nível de transistores

(b) Representação da porta NAND3 no nível lógico

Capacitância equivalente à entrada “A”

A análise da entrada “A” significa verificar a influência das entradas B e C sobre A. Para a rede de transistores *nmos* cujas combinações de entradas correspondem a 000, 001, 100 e 101 o transistor mn2 está no corte e portanto pouco interessa o estado do transistor mn3 e a capacitância equivalente leva em conta C_{SBn1} , C_{GDn2} , C_{DBn2} e C_{GSn1} . Esta situação corresponde a 50% das combinações possíveis das entradas conforme fórmula abaixo:

$$C_{a1} = \frac{1}{2} \frac{C_{GSn1} * C_T}{C_{GSn1} + C_T} \quad (47)$$

onde:

$$C_T = C_{SBn1} + C_{GDn2} + C_{DBn2} \quad (48)$$

Para as combinações de entradas iguais a **011** e **111**, mn2 e mn3 estão saturados, o que corresponde a 25% do total de combinações e a capacitância equivalente leva em conta apenas C_{GSn1} . A capacitância equivalente corresponde a:

$$C_{a2} = \frac{1}{4} C_{GSn1} \quad (49)$$

Os restantes 25% correspondem a situação em que mn2 está saturado e mn3 está no corte, ou seja, **010** e **110**. Neste caso C_{SBn1} , C_{GDn2} , C_{DBn2} , C_{GSn2} , C_{SBn2} , C_{GDn3} e C_{DBn3} influenciam na capacitância equivalente, ou seja:

$$C_{a3} = \frac{1}{4} \frac{C_{GSn1} * C_{T1}}{C_{GSn1} + C_{T1}} \quad (50)$$

onde:

$$C_{T1} = C_{SBn1} + C_{GDn2} + C_{DBn2} + C_{GSn2} + C_{SBn2} + C_{GDn3} + C_{DBn3} \quad (51)$$

Logo a capacitância equivalente total para a entrada “A” será o somatório de todas estas situações, ou seja:

$$C_A = C_{GDn1} + \frac{1}{4} C_{GSn1} + \frac{1}{2} \frac{C_{GSn1} * C_T}{C_{GSn1} + C_T} + \frac{1}{4} \frac{C_{GSn1} * C_{T1}}{C_{GSn1} + C_{T1}} \quad (52)$$

A capacitância equivalente total da entrada “A” considerando as redes pmos e nmos será igual a:

$$C_{ina} = C_A + C_{GSp1} + C_{GDp1} \quad (53)$$

Capacitância equivalente à entrada “B”

Para a entrada “B” as mesmas considerações são feitas em termos de influência dos transistores mn1 e mn3 sobre a entrada B. Para as combinações **000** e **010**, os transistores mn1 e mn3 estão em corte correspondente a 25% das possíveis combinações. Desta forma a capacitância equivalente para a entrada B (C_{b1}) será:

$$C_{b1} = \frac{1}{4} \left(\frac{C_{GDn2} * C_{T2}}{C_{GDn2} + C_{T2}} \right) + \frac{1}{4} \left(\frac{C_{GSn2} * C_{T3}}{C_{GSn2} + C_{T3}} \right) \quad (54)$$

onde:

$$C_{T2} = C_{GSn1} + C_{SBn1} + C_{DBn2} \quad (55)$$

$$C_{T3} = C_{GDn3} + C_{DBn3} + C_{SBn2} \quad (56)$$

Para as entradas **101** e **111**, mn1 e mn3 estão saturados, correspondentes a 25% das possíveis combinações. Logo a capacitância equivalente será:

$$C_{b2} = \frac{1}{4}C_{GDn2} + \frac{1}{4}C_{GSn2} \quad (57)$$

O transistor mn1 está no corte e mn2 está saturado quando as entradas estiverem marcadas em **001** e **011**, logo a capacitância equivalente corresponde a 25% das possíveis combinações:

$$C_{b3} = \frac{1}{4} \left(\frac{C_{GDn2} * C_{T2}}{C_{GDn2} + C_{T2}} \right) + \frac{1}{4}C_{GSn2} \quad (58)$$

E finalmente, para as entradas **100** e **110**, o transistor mn1 está saturado e o transistor mn2 no corte correspondente a 25% das possíveis combinações de entradas.

$$C_{b4} = \frac{1}{4}C_{GDn2} + \frac{1}{4} \left(\frac{C_{GSn2} * C_{T3}}{C_{GSn2} + C_{T3}} \right) \quad (59)$$

A capacitância equivalente a entrada “B” para a rede de transistores *nmos* é o somatório de todas as combinações de entradas, ou seja:

$$C_B = C_{b1} + C_{b2} + C_{b3} + C_{b4} = \frac{1}{2}C_{GDn2} + \frac{1}{2} \frac{C_{GDn2} * C_{T2}}{C_{GDn2} + C_{T2}} + \frac{1}{2}C_{GSn2} + \frac{1}{2} \frac{C_{GSn2} * C_{T3}}{C_{GSn2} + C_{T3}} \quad (60)$$

A capacitância total equivalente a entrada “B” incluindo a rede *pmos* é igual a:

$$C_{inb} = C_B + C_{GSp2} + C_{GDP2} \quad (61)$$

Capacitância equivalente à entrada “C”

Para a entrada “C” os transistores a serem analisados na rede *nmos* serão nm1 e nm2. As combinações **000**, **001**, **100** e **101** fazem com que o transistor mn2 esteja no corte. Desta forma a condição de mn1 para esta situação não importa. Esta situação corresponde a 50% das combinações possíveis das entradas conforme fórmula abaixo:

$$C_{c1} = \frac{1}{2} \frac{C_{GDn3} * C_{T4}}{C_{GDn3} + C_{T4}} \quad (62)$$

onde:

$$C_{T4} = C_{SBn1} + C_{GDn2} + C_{DBn2} \quad (63)$$

As combinações **010** e **011** deixam mn1 no corte e mn2 na saturação, o que corresponde a:

$$C_{c2} = \frac{1}{4} \frac{C_{GDn3} * C_{T5}}{C_{GDn3} + C_{T5}} \quad (64)$$

onde:

$$C_{T5} = C_{SBn1} + C_{GDn2} + C_{DBn} + C_{GDn2} + C_{GSn1} + C_{SBn1} + C_{DBn2} \quad (65)$$

E por fim as combinações **110** e **111** deixam mn1 na saturação e mn2 no corte. Desta forma a capacitância equivalente será:

$$C_{c3} = \frac{1}{4} C_{GDn3} \quad (66)$$

A capacitância equivalente a entrada “C” para a rede *nmos* será igual a:

$$C_C = C_{c1} + C_{c2} + C_{c3} = C_{GSn3} + \frac{1}{4} C_{GDn3} + \frac{1}{2} \frac{C_{GDn3} * C_{T4}}{C_{GDn3} + C_{T4}} + \frac{1}{4} \frac{C_{GDn3} * C_{T5}}{C_{GDn3} + C_{T5}} \quad (67)$$

A capacitância total equivalente a entrada “C” incluindo a rede *pmos* é igual a:

$$C_{inc} = C_C + C_{GSp3} + C_{GDp3} \quad (68)$$

Capacitância equivalente à saída da porta

A capacitância equivalente no nodo de saída da porta será:

$$C_{out} = C_{GDn1} + C_{DBn1} + C_{GDp1} + C_{GDp2} + C_{GDp3} + C_{DBp1} + C_{DBp2} + C_{DBp3} \quad (69)$$

5.3.4 Capacitâncias de Portas Lógicas NAND de *n*-entradas

Para o caso de uma porta **nand** com *n*-entradas (sendo “*n*” um inteiro e maior que 1), probabilidade de comutação igual a 0,5, considerando todas as combinações possíveis de entradas, as capacitâncias equivalentes a rede *pmos* para cada entrada será modelada genericamente conforme demonstrado a seguir:

Teremos três situações a analisar:

1) Capacitância equivalente a entrada cujo transistor *nmos* está ligado à saída da porta;

2) Capacitância equivalente à entrada cujo(s) transistor(es) *nmos* são intermediários entre a saída e o terra da porta;

3) Capacitância equivalente à entrada cujo transistor *nmos* está ligado ao terra.

Em cada uma destas situações todas as combinações de entradas são analisadas de maneira a determinar a situação de corte ou saturação daqueles transistores não pertencentes a entrada que está sendo analisada.

Capacitância equivalente à primeira entrada (transistor conectado a saída)

$$C(0) = C_{GD(0)} + \frac{2}{2^n} C_{GS(0)} + \frac{1}{2^{n-1}} \sum_{q=1}^{n-1} \left(2^{n-q} \frac{C_{GS(0)} * C_{Ti(q-1)}}{C_{GS(0)} + C_{Ti(q-1)}} \right) \quad (70)$$

$$C_{Ti(0)} = C_{SB(0)} + C_{GD(1)} + C_{DB(1)} \quad (71)$$

Para q=1 a n-2

$$C_{Ti(q)} = C_{Ti(0)} + \sum_{i=1}^q (C_{GS(i)} + C_{SB(i)} + C_{GD(i+1)} + C_{DB(i+1)}) \quad (72)$$

Capacitância equivalente às entradas intermediárias ($0 < i < n-1$)

$$C(i) = \frac{2^{n-i}}{2^n} C_{GD(i)} + \frac{1}{2^n} \sum_{q=1}^i \left(2^{n-q} \frac{C_{GD(i)} * C_{T_GD(q-1)}}{C_{GD(i)} + C_{T_GD(q-1)}} \right) + \frac{2^{i+1}}{2^n} C_{GS(i)} + \frac{1}{2^n} \sum_{q=n-1}^{i+1} \left(2^q \frac{C_{GS(i)} * C_{T_GS(n-1-q)}}{C_{GS(i)} + C_{T_GS(n-1-q)}} \right) \quad (73)$$

onde:

$$C_{T_GD(0)} = C_{DB(i)} + C_{GS(i-1)} + C_{SB(i-1)} \quad (74)$$

Para q=1 a i-1

$$C_{T_GD(q)} = C_{T_GD(0)} + \sum_i^{i-q-1} (C_{GS(i-2)} + C_{SB(i-2)} + C_{GD(i-1)} + C_{DB(i-1)}) \quad (75)$$

onde:

$$C_{T_GS(0)} = C_{GD(i+1)} + C_{SB(i)} + C_{DB(i+1)} \quad (76)$$

Para $q=1$ a $i - 1$

$$C_{T_GS(q)} = C_{T_GS(0)} + \sum_i^q (C_{GS(i+1)} + C_{SB(i+1)} + C_{GD(i+2)} + C_{DB(i+2)}) \quad (77)$$

Capacitância equivalente à última entrada (transistor ligado ao terra - $i = n - 1$):

$$C(n-1) = \frac{2}{2^n} C_{GD(n-1)} + C_{GS(n-1)} + \frac{1}{2^n} \sum_{q=1}^{n-1} \left(2^{n-q} \frac{C_{GD(n-1)} * C_{Tf(q-1)}}{C_{GD(n-1)} + C_{Tf(q-1)}} \right) \quad (78)$$

$$C_{Tf(0)} = C_{GS(n-2)} + C_{SB(n-2)} + C_{DB(n-1)} \quad (79)$$

Para $q=1$ a $n-2$

$$C_{Tf(q)} = C_{Tf(0)} + \sum_{i=1}^q (C_{GD(n-1-i)} + C_{DB(n-1-i)} + C_{GS(n-2-i)} + C_{SB(n-2-i)}) \quad (80)$$

onde:

$C_{GD(0), (i), (n-1)}$ = Capacitância *gate-dreno* do transistor *nmos* ligado à saída, transistor(es) intermediário(s) e transistor ligado ao terra, respectivamente.

$C_{GS(0), (i), (n-1)}$ = Capacitância *gate-fonte* do transistor *nmos* ligado à saída, transistor(es) intermediário(s) e transistor ligado ao terra, respectivamente.

$C_{SB(0), (i), (n-1)}$ = Capacitância *fonte-substrato* do transistor *nmos* ligado à saída, transistor(es) intermediário(s) e transistor ligado ao terra, respectivamente.

$C_{DS(0), (i), (n-1)}$ = Capacitância *dreno-substrato* do transistor *nmos* ligado à saída, transistor(es) intermediário(s) e transistor ligado ao terra, respectivamente.

$C_{(0),(i), (n-1)}$ = Capacitância de porta concentrada no nodo da primeira entrada, entrada intermediária “i” e última entrada.

n = número de entradas da porta lógica

Capacitância Equivalente à Saída

A capacitância equivalente na saída da porta será igual a:

$$C_{out} = C_{GD(0)} + C_{DB(0)} + \sum_{i=0}^{n-1} (C_{GDp(i)} + C_{DBp(i)}) \quad (81)$$

onde:

C_{out} = Capacitância equivalente à saída da porta lógica

$C_{GD(0)}$ = Capacitância *gate-dreno* do transistor *nmos* ligado à saída

$C_{DB(0)}$ = Capacitância *dreno-substrato* do transistor *nmos* ligado à saída

$C_{GD(i)}$ = Capacitância *gate*-dreno do transistor “*i*” *pmos* ligado à saída

$C_{DB(i)}$ = Capacitância dreno-substrato do transistor “*i*” *pmos* ligado à saída

5.3.5 Capacitâncias de Portas Lógicas NOR de “*n*” entradas

As portas lógicas **nor**'s são duais das portas **nand**'s, com isto as capacitâncias internas surgirão na rede de transistores *pmos* e as mesmas considerações em relação aos vetores de entrada deverão ser feitas. As fórmulas para a rede *pmos* consideram a influência das entradas nos transistores da rede *pmos*, pelo fato de que os mesmos encontram-se em série.

5.4 Potência Média Dissipada pela Porta

A estimativa das capacitâncias equivalentes em cada nodo de uma porta lógica associada à sua atividade de comutação, tensão de alimentação e frequência, possibilita estimar a potência dinâmica consumida pela porta em cada nodo. Logo, a potência total da porta ou circuito será o somatório da potência consumida em cada nodo externo. Na análise feita na seção anterior, não estão sendo consideradas as capacitâncias de interconexões, portanto usando a fórmula 3 apresentada no capítulo 2 temos a estimativa da potência consumida devido apenas às capacitâncias intrínsecas será igual a:

$$P_{in(n)(m)} = 0.5 f_c V_{dd}^2 \cdot C_{in(n)(m)} \cdot \alpha_m \quad (82)$$

onde:

$P_{in(n)(m)}$ é igual a estimativa da potência média de porta referente ao nodo “*n*” da porta lógica “*m*”

f_c é a frequência de operação

V_{dd} é a tensão de alimentação do circuito

$C_{in(n)(m)}$ é a estimativa de capacitância de porta referente ao nodo “*n*” de entrada da porta lógica “*m*” ou da saída “out”.

α é a atividade de comutação da porta lógica “*m*”

“*n*” é igual ao número de entrada da porta lógica

“*m*” é igual à porta lógica em análise

5.5 Validação do Método de Estimativa de Potência de Porta

A validação das fórmulas apresentadas nas seções 5.3 e 5.4 é feita pelo uso de uma ferramenta que simula a potência de circuitos CMOS ao nível elétrico (SPICE). O procedimento adotado de validação está baseado no fluxograma mostrado na Figura 5.5. Nesta figura três ferramentas de CAD são usadas: O TROPIC2 [MOR 93] que gera o leiaute do circuito, a ferramenta comercial CADENCE que extrai as capacitâncias parasitas e identifica os transistores em cada porta lógica e o *Power Estimate* [MON 97] que fornece as atividades de comutação em cada nodo do circuito no nível lógico.

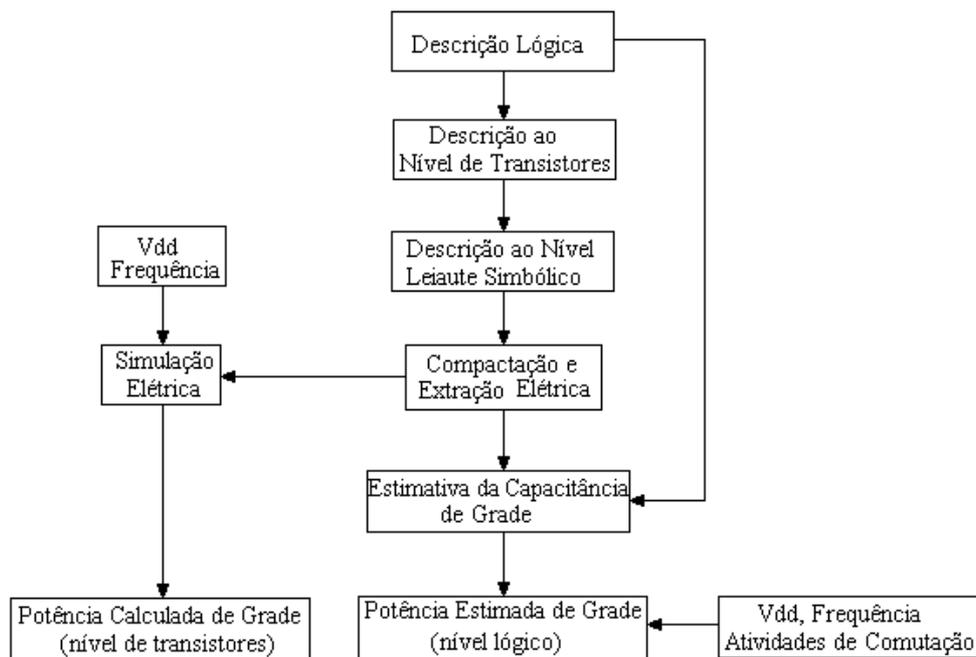


FIGURA 5.5 – Validação do método proposto

O algoritmo de validação do modelo proposto possui a seguinte seqüência de operações:

- 1) Leitura do circuito no formato lógico (blif);
- 2) Conversão do arquivo de formato lógico (blif) para o formato SPICE (sim);
- 3) Uso do TROPIC para geração de um arquivo simbólico ao nível de leiaute.
- 4) Compactação do arquivo simbólico e geração do leiaute para uma dada tecnologia (CADENCE).
- 5) Extração das capacitâncias parasitas e transistores;
- 6) Relação de cada porta lógica com seus respectivos transistores.
- 7) Estimativa da capacitância equivalente em cada nodo da porta lógica pertencente ao circuito, pelo uso das fórmulas propostas na seção anterior.

As atividades de comutação em cada nodo fornecido pelo *Power Estimate*, associado as capacitâncias estimadas, frequência e tensão de alimentação possibilitam estimar a potência dissipada no circuito.

A mesma estimativa é realizada ao nível de transistores pelo SPICE.

E por fim os resultados da estimativa lógica e elétrica são analisados e comparados.

5.5.1 Simulação Elétrica

O SPICE [NAG 75] é uma ferramenta que analisa um circuito ao nível de transistores através da modelagem matemática e física de seus componentes. Os vetores de entrada para simulação devem levar em consideração todas as transições possíveis de estado dependendo do número de entradas do circuito. Este procedimento é fundamental pois estamos comparando simulação elétrica com simulação lógica.

O número de vetores de entrada a ser usado na simulação elétrica depende do número de entradas. A Tabela 5.1 apresenta uma relação entre estes dois parâmetros. Observe que, uma última transição deve ser adicionada a combinação normal de vetores, à fim de que uma última transição do sinal seja computada.

TABELA 5.1 – Número de entradas versus número de combinações de entrada

Número Entradas	Número Combinações	Comb. Extra	Total Combinações	Tempo Inicial de Simulação	Tempo Final de Simulação
1	4	1	5	1/2 t	4t + 1/2t
2	16	1	17	1/2 t	16t + 1/2t
3	64	1	65	1/2 t	64t + 1/2t
n	$2\exp(2*n)$	1	$2\exp(2*n) + 1$	1/2 t	$2\exp(2*n) t + 1/2t$

Por exemplo, um circuito com duas entradas terá o diagrama de estados representado na Figura 5.6 e o formato de onda das entradas para a simulação elétrica representado na Figura 5.7.

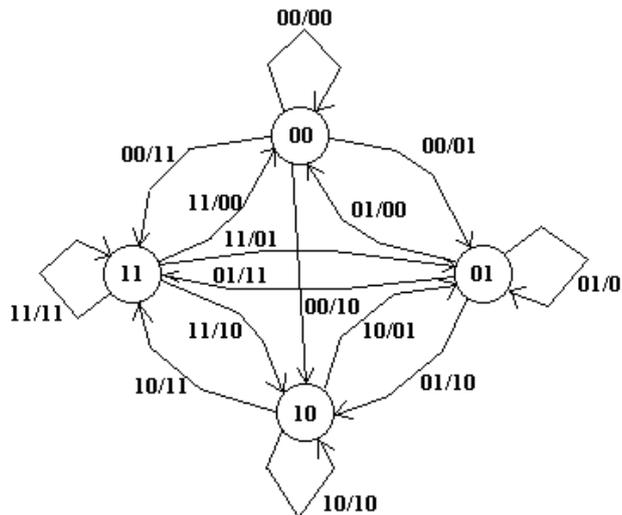


FIGURA 5.6 – Diagrama de estados para um circuito com duas entradas

Pelo diagrama da Figura 5.6, observa-se que se as entradas estiverem no estado 00, poderão mudar para o estado 01, 11, 10 ou permanecer no próprio estado 00. Da mesma forma se o estado for 01, a mudança de estado ocorrerá para 10, 11 ou permanecer no próprio estado. E assim sucessivamente, de modo que serão 16 possíveis mudanças de estado. Um combinação extra permite computar a última variação do sinal e desta forma todo o espectro possível de combinações estarão cobertos.

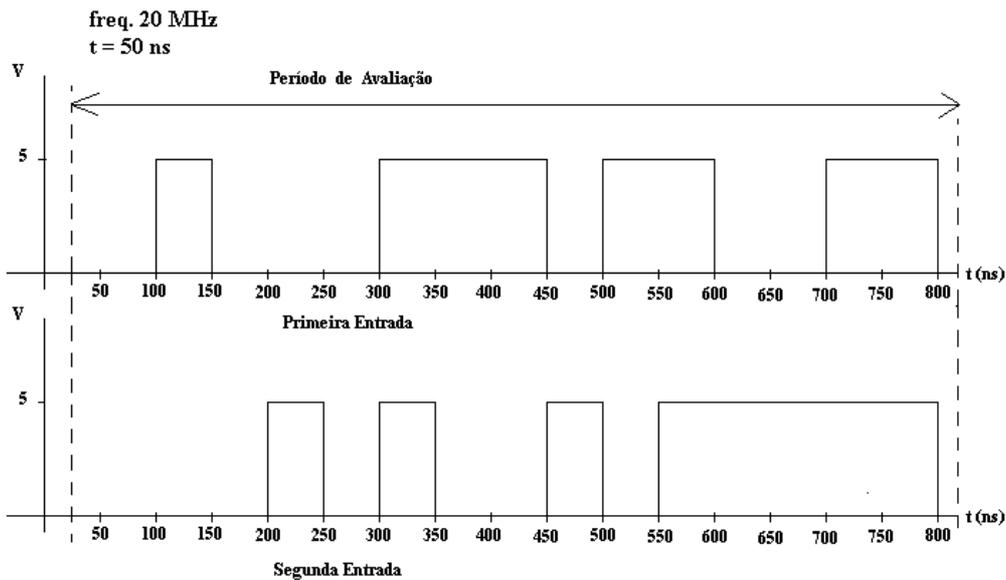


FIGURA 5.7 – Vetores de simulação para um circuito com duas entradas

Na simulação elétrica do exemplo citado, os vetores de entrada do circuito são representados por tensões moduladas, conforme Figura 5.7, de modo a contemplar todas as possíveis mudanças de estado que causam uma transição do sinal.

5.5.2 Condições Gerais para Simulações de Circuitos

Portas lógicas CMOS convencionais e circuitos *benchmarks* são simulados. Os leiautes são gerados, os transistores obtidos e as capacitâncias extraídas. A simulação considerou:

- tecnologia de fabricação igual a 0,8 μm ;
- frequência de operação igual a 20MHz;
- tensão de alimentação igual a 5V;
- atividades de comutação em cada nodo da porta lógica gerados pelo Power *Estimate* [MON 97] considerando atraso zero para cada porta;
- estimativa do consumo médio de potência dinâmica de porta.

5.5.3 Resultados e Conclusões

Os dados apresentados na Tabela 5.2 mostram resultados de simulações realizadas com portas lógicas convencionais (**inv**, **nand2**, **nand3**, **nand4**, **and2**, **and3**, **and4**, **nor2**, **nor3**, **nor4**, **or2**, **or3** e **or4**). Inicialmente, gerou-se o leiaute destas portas para uma tecnologia de 0,8 μm , extraiu-se as capacitâncias e obteve-se os transistores no modelo elétrico. Na sequência, foi realizada a estimativa de potência de porta ao nível lógico utilizando-se dos modelos desenvolvidos na seção 5.3. A validação do modelo proposto foi realizada pela simulação elétrica das portas usando-se a ferramenta SPICE. Os erros entre os métodos elétrico e lógico variam entre 0,25% (**and2**) e 3,97% (**nand4**) apresentados na Tabela 5.2. Este valores são muito importantes pois validam a teoria proposta na seção anterior.

TABELA 5.2 – Consumo de Potência de Portas Lógicas CMOS convencionais

Potência Dissipada			
Nome do Circuito	Simulação Lógica (microW)	Simulação Elétrica (microW)	Erro %
inv	0,139	0,141	1,42
nand2	2,92	2,88	1,39
nand3	4,76	4,60	3,48
nand4	6,28	6,04	3,97
and2	4,07	4,08	0,25
and3	4,48	4,59	2,40
and4	5,00	4,91	1,83
nor2	3,31	3,22	2,80
nor3	4,36	4,3	1,40
no4	5,89	5,76	2,26
or2	4,42	4,38	0,91
or3	4,77	4,73	0,85
or4	5,81	5,68	2,29

Já na Tabela 5.3, apresenta-se as características de circuitos um pouco mais complexos que serão usados como forma de também validar este método. São circuitos *benchmarks* MCNC'91, sendo eles: C17, cm138a, cm42a, decod e majority. A escolha destes circuitos foi baseada em duas condições:

- 1) baixo número de entradas, tendo em vista a complexidade na geração dos vetores de entrada;
- 2) baixo nível lógico, devido às considerações de atraso zero para as portas lógicas do circuito.

TABELA 5.3 – Características dos Circuitos *Benchmarks* Analisados

Nome do Circuito	Número de Portas Lógicas	Número de Transistores	Nós	Número de Capac. de Interconexões
C17	6	24	19	60
cm138a	13	96	56	231
cm42a	16	100	56	241
decod	22	146	80	288
majority	7	46	30	124

Estes circuitos apresentam erros na estimativa da potência, variáveis entre 2,47% (cm138a) a 9,89% (majority), quando comparados os métodos lógico e elétrico (Tabela 5.4). Esta taxa de erros percentuais é importante, apesar dos circuitos serem pequenos e não haver considerações de capacitâncias de interconexões, porque validam mais uma vez o método proposto.

TABELA 5.4 – Estimativa da potência de porta em circuitos CMOS

Nome do Circuito	Nível Lógico	Potência Dissipada		
		Simulação Lógica (microW)	Simulação Elétrica (microW)	Erro %
C17m	3	30,21	32,54	-7,16
cm42a	3	114,72	120,54	-4,85
decod	3	221,77	234,24	-5,32
majority	4	58,49	64,91	-9,89
cm138a	3	96,44	99,4	-2,47

Na próxima seção as fórmulas apresentadas serão adaptadas para uma situação em que as dimensões dos transistores como W, L, perímetro e área do dreno, perímetro e área da fonte são valores médios obtidos da geração de vários leiautes, com informações da tecnologia de geração do leiaute.

5.6 Aproximações nas Dimensões dos Transistores

Nas seções 5.3 e 5.4, desenvolveu-se um modelo matemático que estima as capacitâncias de porta no nível lógico. Estas capacitâncias estão concentradas nos nodos externos das portas lógicas, dependem das dimensões dos transistores e capacitâncias parasitas. A validação do modelo foi feita extraindo-se estas capacitâncias e transistores pelo uso de um extrator elétrico. No entanto, sabe-se que no nível lógico as capacitâncias não são extraídas do leiaute, pois não existe leiaute gerado. Existe então, a necessidade de **estimar** às dimensões dos transistores usando-se valores médios de área e perímetro do dreno, área e perímetro da fonte. O comprimento (L) e largura do canal (W) são fornecidos pela tecnologia.

Deste modo, as fórmulas deduzidas nas seções 5.3 e 5.4 serão aproximadas, e adequadas a esta situação conforme demonstrado abaixo:

$$C_{GSn} = C_{GDn} = CA_n$$

$$C_{SBn} = CB_n$$

$$C_{DBn} = CC_n$$

$$C_{GSp} = C_{GDp} = CA_p$$

$$C_{SBp} = CB_p$$

$$C_{DBp} = CC_p$$

Onde:

$CA_{n(p)}$ representa a capacitância *gate-fonte* do transistor *nmos* (ou *pmos*).

$CB_{n(p)}$ representa a capacitância *fonte-substrato* do transistor *nmos* (ou *pmos*).

$CC_{n(p)}$ representa a capacitância *dreno-substrato* do transistor *nmos* (ou *pmos*).

5.6.1 Inversor CMOS

O inversor apresentado na seção 5.3.1 é adaptado aqui para a situação em que as dimensões dos transistores são estimadas pelo valor médio (Figura 5.8) e não mais pelos valores originados do leiaute. Portanto, as fórmulas de capacitâncias equivalentes são definidas genericamente como:

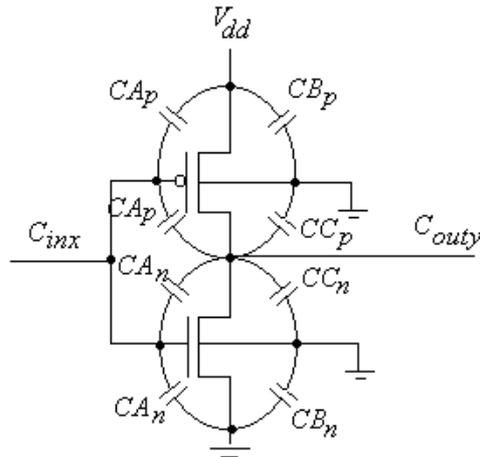


FIGURA 5.8- Aproximações das Capacitâncias Intrínsecas

$$C_{inx} = 2 * CA_n + 2 * CA_p \quad (83)$$

$$C_{outy} = CC_p + CA_p + CA_n + CC_n \quad (84)$$

5.6.2 NAND com duas entradas

Da mesma forma que o inversor, as fórmulas deduzidas para a porta NAND com duas entradas podem ser aproximadas com dados obtidos da estimativa das dimensões médias dos transistores e tecnologia de fabricação. A Figura 5.9, apresenta o modelo apresentado na seção 5.3.2, porém com as devidas aproximações.

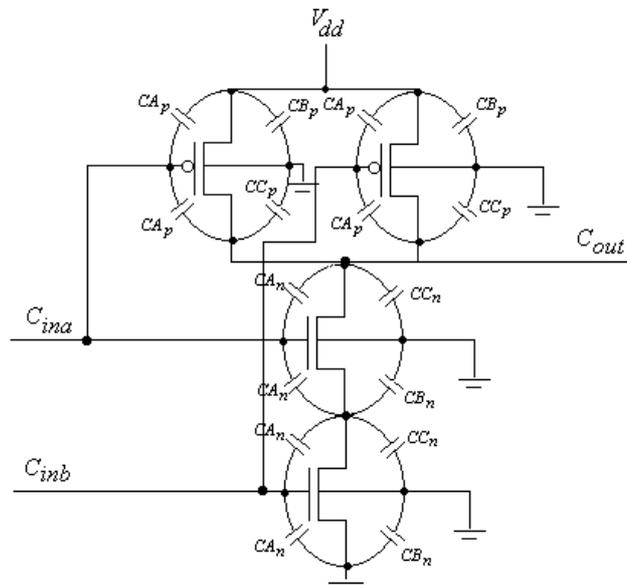


FIGURA 5.9 – NAND com duas entradas e dimensões dos transistores estimadas

$$C_{ina} = C_{inb} = \frac{3}{2}CA_n + \frac{1}{2}\left(\frac{CA_n * C_T}{CA_n + C_T}\right) + 2CA_p \quad (85)$$

$$C_T = CA_n + CB_n + CC_n \quad (86)$$

$$C_{out} = CA_n + CC_n + 2CA_p + 2CC_p \quad (87)$$

5.6.3 NAND com número de entradas qualquer

Nesta seção as fórmulas desenvolvidas em 5.3.4 são aproximadas considerando as estimativas das dimensões médias dos transistores para a situação da porta lógica **nand** com “*n*” entradas.

Capacitância referente à primeira entrada (transistor conectado a saída)

$$C(0) = CAN_{(0)} + \frac{2}{2^n}CAN_{(0)} + \frac{1}{2^{n-1}}\sum_{q=1}^{n-1}\left(2^{n-q}\frac{CAN_{(0)} * C_{Ti(q-1)}}{CAN_{(0)} + C_{Ti(q-1)}}\right) \quad (88)$$

onde:

$$C_{Ti(q)} = CAn + CBn + CCn + (n-2)(2CAn + CBn + CCn) \quad (89)$$

Capacitância referente as entradas intermediárias ($0 < i < n-1$):

$$C(i) = \frac{2^{n-i}}{2^n}CAN_{(i)} + \frac{1}{2^n}\sum_{q=1}^i\left(2^{n-q}\frac{CAN_{(i)} * C_{T-GD(q-1)}}{CAN_{(i)} + C_{T-GD(q-1)}}\right) + \frac{2^{i+1}}{2^n}CAN_{(i)} + \frac{1}{2^n}\sum_{q=n-1}^{i+1}\left(2^q\frac{CAN_{(i)} * C_{T-GS(n-1-q)}}{CAN_{(i)} + C_{T-GS(n-1-q)}}\right) \quad (90)$$

onde:

Para $q=1$ a $i-1$

$$C_{T-GD(q)} = C_{T-GD(0)} + \sum_i^{i-q-1}\left(2CAn_{n(i-2)} + CBn_{n(i-2)} + CCn_{n(i-1)}\right) \quad (91)$$

$$C_{T-GS(0)} = CAn_{n(i+1)} + CBn_{n(i)} + CCn_{n(i+1)} \quad (92)$$

e

$$C_{T-GS(q)} = C_{T-GS(0)} + \sum_i^q\left(2CAn_{n(i+1)} + CBn_{n(i+1)} + CCn_{n(i+2)}\right) \quad (93)$$

Capacitância referente à última entrada (transistor conectado ao terra)

$$C(n-1) = \frac{2}{2^n} CAn_{(n-1)} + CAn_{(n-1)} + \frac{1}{2^n} \sum_{q=1}^{n-1} \left(2^{n-q} \frac{CAn_{(n-1)} * C_{Tf(q-1)}}{CAn_{(n-1)} + C_{Tf(q-1)}} \right) \quad (94)$$

e

$$C_{Tf(0)} = CAn_{(n-2)} + CBn_{(n-2)} + CCn_{(n-1)} \quad (95)$$

Para q=1 a n-2

$$C_{Tf(q)} = C_{Tf(0)} + \sum_{i=1}^q (2CAn_{(n-1-i)} + CBn_{(n-1-i)} + CCn_{(n-2-i)}) \quad (96)$$

Genericamente, as capacitâncias de porta equivalentes a NAND's com "n" entradas são definidas como:

$$Cin(0) = C(0) + n(2CAp) \quad (97)$$

$$Cin(i) = C(i) + n(2CAp) \quad (98)$$

.....

$$Cin(n) = C(n-1) + n(2CAp) \quad (99)$$

A capacitância equivalente à saída é igual a:

$$Cout = CAn + CCn + n(CAp + CCP) \quad (100)$$

As mesmas considerações são feitas para a porta lógica tipo **nor**, pois devido a sua dualidade em relação a porta **nand** a malha pmos está com os transistores em série e a malha nmos com os transistores em paralelo. Para este caso, as estimativas das dimensões dos transistores seguem o mesmo procedimento adotado nesta seção.

5.7 Conclusões

Neste capítulo foi desenvolvido um método de estimativa de capacitâncias de porta e potência dissipada em portas lógicas, no nível lógico. Não levou-se em consideração as capacitâncias das interconexões. O método leva em conta apenas as combinações de vetores de entrada em circuitos que possuem portas lógicas convencionais CMOS. As capacitâncias dos nodos internos são refletidos às entradas de acordo com estes vetores de entrada.

O método foi validado pelo uso de um simulador elétrico reconhecido no meio acadêmico SPICE (*model2 pmos level=2 e model3 nmos level=2*). Os erros ocorridos são devidos às aproximações feitas nas capacitâncias intrínsecas e aos valores submicrônicos de capacitâncias, de maneira que uma pequena variação produz erros consideráveis.

Pelo fato de não haver necessidade da geração do leiaute do circuito, este método é extremamente rápido na avaliação da potência dissipada. O que não ocorre numa simulação elétrica.

6 Estimativa do Consumo de Potência devido as Interconexões

6.1 Introdução

O objetivo deste capítulo é estimar a contribuição das interconexões no consumo de potência em circuitos combinacionais CMOS no nível lógico, vindo complementar o modelo desenvolvido no capítulo anterior. O método proposto para estimar as **capacitâncias das interconexões** e **potência das interconexões** baseia-se na estimativa do **comprimento médio** das interconexões.

Este capítulo inicialmente faz uma revisão dos conceitos e tipos de capacitâncias que surgem em decorrência das interconexões, e após apresenta um modelo de estimativa de capacitâncias e potência no nível lógico.

6.2 Capacitâncias de Interconexões

Capacitâncias de interconexões surgem quando é feita a conexão entre as diversas células de um circuito.

As capacitâncias (Figura 6.1) que surgem devido às interconexões [CHE 92] são:

Capacitância de overlap (C_{23a} , C_{21a}) – é a capacitância devido ao cruzamento entre dois condutores em planos diferentes.

Capacitância lateral (C_{221}) – é capacitância que surge entre dois condutores no mesmo plano.

Capacitância de Borda (*Fringing*) (C_{23fr} , C_{21fr}) – é a capacitância devido ao acoplamento entre dois condutores em diferentes planos (face inferior do metal em um nível superior e face lateral do metal num nível inferior).

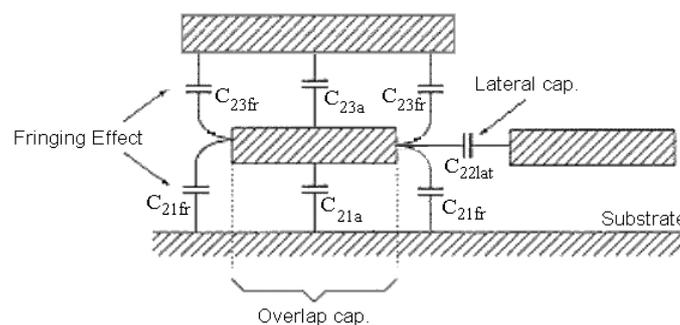


FIGURA 6.1 – Capacitâncias devidas as interconexões

6.3 Comprimento Médio de Interconexões

Uma interconexão é um fio metálico que interliga uma ou várias células de um circuito. O número de interconexões de um circuito depende da maneira como as células são posicionadas na planta baixa e como é feito o roteamento. O roteamento depende da qualidade do algoritmo implementado [SHE 93]. O comprimento médio das

interconexões de um circuito é igual à média do comprimento de todas as suas interconexões [PED 89].

As interconexões de roteamento são responsáveis pelo surgimento de capacitâncias parasitas de roteamento.

Nesta seção será desenvolvido um método de estimativa do comprimento das interconexões que baseia-se na relação entre comprimento de interconexão, *fanout* das redes (número de saídas de cada porta) e número de transistores.

Esta relação é obtida pelo levantamento de dados oriundos de vários circuitos. Estes dados de comprimentos de interconexões são obtidos após a geração completa do leiaute do circuito. O TROPIC3 [MOR99] é usado na geração do leiaute e serve como fonte de informações sobre comprimento das interconexões de cada rede em cada circuito analisado.

Os circuitos usados como fonte de dados foram escolhidos dos *benchmarks* MCNC'91, com diversos tamanhos e complexidades, conforme demonstrado na Tabela 6.1. Os circuitos foram gerados para duas tecnologias de fabricação: 0,25 μ m e 0,35 μ m da Empresa AMS (*AustriaMicroSystems*). O objetivo do uso destas duas tecnologias é verificar se existe uma relação entre elas.

6.3.1 Tecnologia AMS 0,25 μ m

A Tabela 6.1 fornece dados do conjunto de circuitos envolvidos na geração dos leiautes, com informações do número e tipo de portas lógicas e número de transistores. O número de transistores dos circuitos combinacionais analisados variam de 192 transistores (cmb_a) até 2690 transistores (C1908a).

TABELA 6.1 – Característica dos Circuitos usados no levantamento dos dados

Nome Circuito	Número Portas	Número Transistores	Número Inversores	Número NAND	Número NOR
cmb_a	48	192	18	18	12
parity_a	75	240	30	45	0
pcler8_a	127	412	58	35	34
frg1_a	151	536	64	37	50
count_a	174	602	79	33	62
comp_a	224	656	78	87	59
5xp1	299	798	199	54	46
C432a	259	902	102	85	72
9sym	409	1092	272	70	67
9sym_3x3	339	1156	100	124	115
bw	468	1256	308	94	66
C880a	474	1552	154	158	162
alu2_a	469	1712	184	130	155
C499a	605	2038	223	238	144
x4_a	545	2160	224	156	165
c1355	604	2244	130	472	2
C1908a	728	2690	204	291	233

As informações obtidas do levantamento de dados visam a elaboração de uma **tabela de referência** (*look-up-table*) que será usada na estimativa do comprimento das interconexões. O processo completo de levantamento desta tabela consiste:

1) na geração automática do leiaute dos circuitos apresentados na Tabela 6.1, cuja tecnologia de fabricação é igual a 0,25 μ m. Estes leiautes foram otimizados para que tenham o menor comprimento médio das interconexões do circuito, e para isto o bloco lógico a que pertence deverá ter dimensões quadradas, ou seja, a largura e comprimento devem ter as mesmas dimensões (ou aproximadas). Esta otimização é obtida pela escolha apropriada do número de bandas, quando o leiaute é gerado pela ferramenta de geração. A Tabela 6.2 mostra dados referentes à geração do leiaute de cada circuito, como por exemplo número de bandas, largura e comprimento de cada bloco lógico e semi-perímetro;

2) para cada leiaute gerado, é feita a filtragem das redes com diferentes *fanouts* e os seus respectivos comprimentos médios de interconexões. As Tabelas 6.3, 6.4, 6.5 e 6.6 apresentam um resumo destes levantamentos;

3) plotagem de curvas que relacionam comprimento médio das interconexões em função do número de transistores do circuito para redes com diferentes *fanouts* (Figuras 6.2, 6.3, 6.4, 6.5, 6.6, 6.7, 6.8 e 6.9);

4) na análise destas curvas, estabelecendo **tabelas de referências** em função do *fanout* (Tabela 6.7) e *fanin* (Tabela 6.8), a serem usadas na **estimativa do comprimento médio** para cada rede do circuito.

TABELA 6.2 – Dados usados na geração dos leiautes dos circuitos para tecnologia AMS 0,25 μ m

Nome do Circuito	Número de Bandas	Largura do bloco (micrometro)	Comprimento do Bloco (micrometro)
cmb_a	4	43,70	55,40
parity_a	4	52,00	53,40
pcler8_a	5	72,00	73,00
frg1_a	5	91,70	79,00
count_a	6	86,80	89,60
comp_a	6	94,70	84,60
5xp1	7	107,45	101,20
C432a	7	109,45	113,20
9sym	8	126,00	122,80
bw	9	129,45	144,40
C880a	9	148,45	156,40
alu2_a	9	163,45	159,40
C499a	10	175,00	168,00
x4_a	10	182,00	173,00
c1355	10	178,70	168,00
C1908a	11	208,00	192,60

TABELA 6.3 – Comprimento médio (μm) das redes para *fanouts* iguais a 0, 1 e 2 (tecnologia 0,25 μm)

Nome do Circuito	Número de Trans.	Redes c/ Fanout "0"	Comp. Médio (micra)	Redes c/ Fanout "1"	Comp. Médio (micra)	Redes c/ Fanout "2"	Comp. Médio (micra)
cmb_a	192	4	26,90	42	33,00	2	50,60
parity_a	240	1	17,20	60	28,50	14	51,90
pcler8_a	412	17	35,50	101	33,10	1	47,20
frg1_a	536	3	17,20	144	36,00	4	51,90
count_a	602	16	26,70	128	41,10	16	41,50
comp_a	656	1	16,60	158	35,40	31	57,90
5xp1	798	1	17,20	271	31,50	10	52,80
C432a	902	4	22,20	176	38,30	52	75,10
9sym	1092	1	32,30	378	34,60	9	56,30
bw	1256	19	25,30	399	33,50	18	57,30
C880a	1552	25	48,50	317	43,20	43	69,90
alu2_a	1712	5	21,40	387	42,00	19	66,80
C499a	2038	32	63,10	406	37,90	135	75,90
x4_a	2160	66	13,60	450	39,10	7	60,40
c1355	2244	32	81,40	346	37,00	200	70,30
C1908a	2690	25	71,70	525	43,20	119	88,60

A Tabela 6.3 é o resultado dos leiautes gerados de circuitos cujas redes possuem *fanouts* iguais a 0, 1 e 2, com seus respectivos comprimentos médios de interconexões.

O comprimento médio das redes com *fanout* igual a zero, corresponde à saída do circuito e depende da borda, ou da maneira como são definidas as interfaces na geração do leiaute (norte, sul, leste, oeste). Colocando estes dados na Figura 6.2, observa-se que os valores do comprimento médio das interconexões variam de 20 a 40 μm para circuitos que possuem até 1700 transistores e de 20 a 80 μm para circuitos que possuem de 1700 a 2700 transistores.

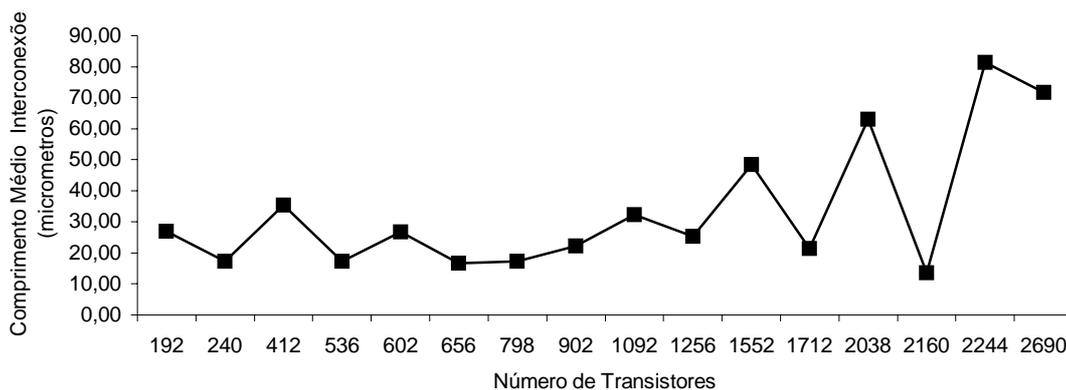


FIGURA 6.2 – Comp. médio das redes versus núm. de trans. para *fanout* igual a ZERO (tecnologia AMS 0,25 μm)

O comprimento médio das redes com *fanout* igual a 1, representados na Figura 6.3, abrangem todos os tipos de circuitos analisados para coleta de dados, sendo seus valores variáveis de 30 μ m a 40 μ m, independente do número de transistores.

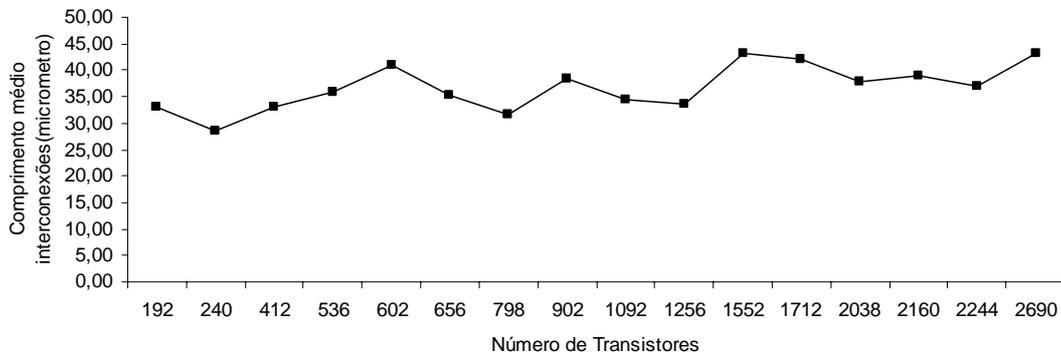


FIGURA 6.3 – Comp. médio das redes versus núm. de trans. para *fanout* igual a UM (tecnologia AMS 0,25 μ m)

Pela Figura 6.4, observa-se que as redes cujo *fanout* são iguais a 2, possuem uma maior preponderância no comprimento médio das interconexões. Neste caso verifica-se que os valores de comprimento médio das interconexões para estas redes variam de 40 μ m a 80 μ m, dependendo do número total de transistores do circuito. Até 1700 transistores o comprimento médio é variável entre 40 μ m e 60 μ m. De 1701 a 2300 transistores esta variação fica entre 60 μ m e 80 μ m.

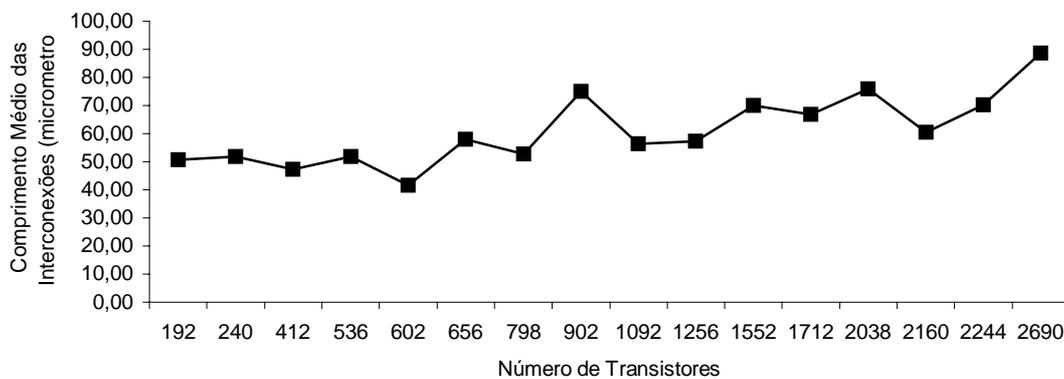


FIGURA 6.4 – Comp. médio das redes versus núm. de trans. para *fanout* igual a DOIS (tecnologia AMS 0,25 μ m)

A Tabela 6.4 apresenta circuitos que possuem redes com *fanouts* iguais a 3, 4 e 5. De uma maneira geral, observa-se que para redes com *fanouts* maiores que 3, o comprimento das interconexões aumenta consideravelmente.

TABELA 6.4 – Comprimento médio (μm) das redes para *fanouts* iguais a 3, 4 e 5

Nome do Circuito	Número de Trans.	Redes c/ Fanout "3"	Comp. Médio (micra)	Redes c/ Fanout "4"	Comp. Médio (micra)	Redes c/ Fanout "5"	Comp. Médio (micra)
cmb_a	192	0		0		0	
parity_a	240	0		0		0	
pcler8_a	412	6	65,30	0		0	
frg1_a	536	0		0		0	
count_a	602	14	58,70	0		0	
comp_a	656	4	54,40	4	81,00	0	
5xp1	798	9	86,90	3	148,00	3	151,40
C432a	902	2	87,20	2	140,00	1	245,80
9sym	1092	6	123,40	4	122,00	6	160,80
bw	1256	8	72,70	11	146,00	6	249,00
C880a	1552	19	128,50	10	108,00	6	147,60
alu2_a	1712	23	101,50	14	154,00	5	141,70
C499a	2038	5	69,90	13	123,00	2	148,60
x4_a	2160	2	72,80	3	160,00	2	143,10
c1355	2244	0		18	117,00	0	
C1908a	2690	15	110,20	23	122,00	1	122,50

A Figura 6.5, mostra a situação das redes que possuem *fanout* igual a 3. Observa-se uma preponderância maior deste tipo de rede sobre o comprimento médio das redes, que variam de $60\mu\text{m}$ a $120\mu\text{m}$.

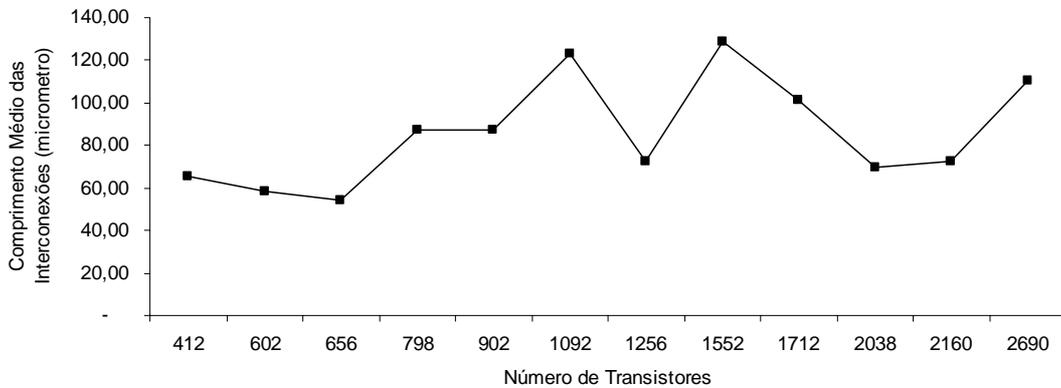


FIGURA 6.5 – Comp. médio das redes versus núm. de trans. para *fanout* igual a TRÊS (tecnologia AMS $0,25\mu\text{m}$)

A Figura 6.6 mostra redes com *fanouts* iguais a 4 para os circuitos analisados. Estes possuem um comprimento médio das interconexões variáveis entre $80\mu\text{m}$ e $160\mu\text{m}$, independente do número de transistores.

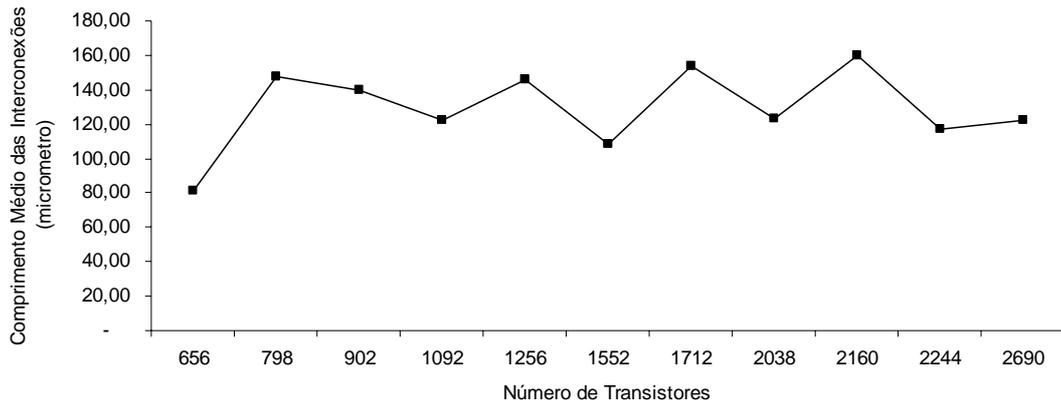


FIGURA 6.6 – Comp. médio das redes versus núm. de trans. para *fanout* igual a QUATRO (tecnologia AMS 0,25µm)

Para circuitos cujas redes possuem *fanout* igual a 5, os comprimentos médios variam entre 150µm e 250µm conforme observa-se na Figura 6.7.

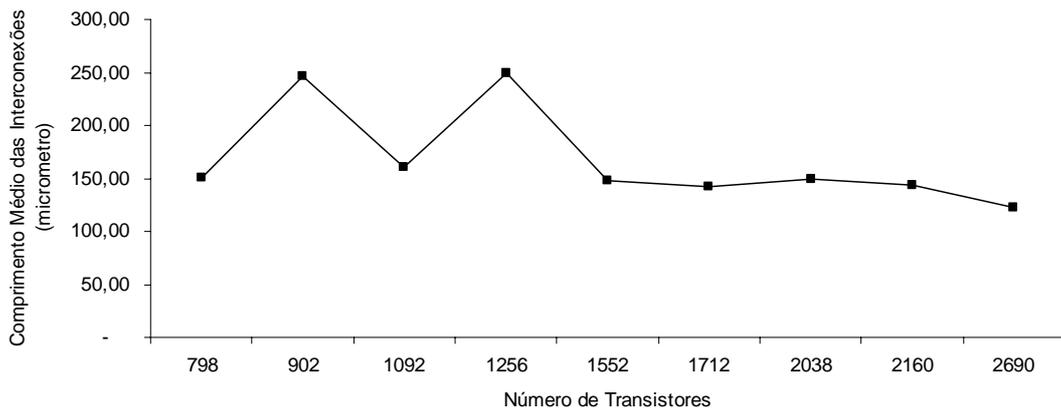


FIGURA 6.7 – Comp. médio das redes versus núm. de trans. para *fanout* igual a CINCO (tecnologia AMS 0,25µm)

A Tabela 6.5 apresenta circuitos que possuem redes com *fanouts* iguais a 6 e maiores. Para este caso, o comprimento médio assume valores consideráveis e não seguem a uma regra definida. Normalmente, os circuitos possuem muito poucas redes com *fanouts* maiores que 6, conforme comprovado neste circuitos gerados. As ferramentas de CAD que fazem roteamento são otimizadas para reduzir o comprimento das interconexões.

TABELA 6.5 – Comprimento médio (μm) das redes para *fanouts* iguais a 6 e maiores

Nome do Circuito	Número de Trans.	Redes c/ Fanout "6"	Comp. Médio (micra)	Redes c/ Fanout "maior que 6"	Comp. Médio (micra)
cmb_a	192	0	0	0	0
parity_a	240	0	0	0	0
pcler8_a	412	0	0	2	200,1
frg1_a	536	0	0	0	0
count_a	602	0	0	0	0
comp_a	656	0	0	0	0
5xp1	798	2	205,8	0	0
C432a	902	0	0	3	589,3
9sym	1092	2	131,2	3	227,2
bw	1256	2	362,5	6	306,3
C880a	1552	0	0	3	312,4
alu2_a	1712	5	235,1	11	373,5
C499a	2038	4	210,2	8	275,6
x4_a	2160	3	131,7	13	415,2
c1355	2244	0	0	8	492,8
C1908a	2690	2	215,3	18	360,8

O comprimento médio das redes que possuem *fanouts* iguais a 6 ou maiores, são difíceis de se estimar, e por consequência aumentam consideravelmente o comprimento médio das interconexões dos circuitos. Seus valores para os circuitos analisados variam entre $150\mu\text{m}$ e $600\mu\text{m}$, conforme observa-se nas Figuras 6.8 e 6.9.

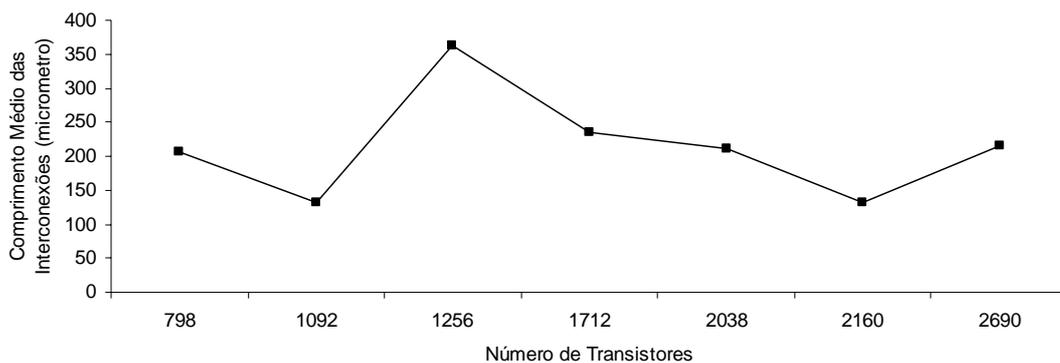


FIGURA 6.8 – Comp. médio das redes versus núm. de trans. para *fanout* igual a SEIS (tecnologia AMS $0,25\mu\text{m}$)

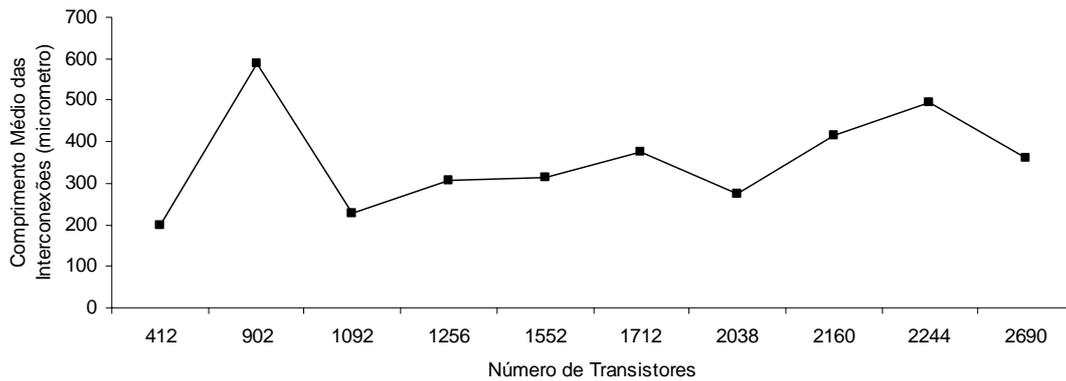


FIGURA 6.9 – Comp. médio das redes versus núm. de trans. para *fanout* maior do que SEIS (tecnologia AMS 0,25µm)

Outra análise que deve ser feita é o percentual de redes de diferentes *fanouts* (0, 1, 2, 3, 4, 5, 6 e maiores) sobre o total de redes em cada circuito. Estes dados são apresentados na Tabela 6.6. Observa-se que em todos os circuitos, há uma maior quantidade de redes com *fanout* igual a 1, enquanto que existem poucas redes com *fanout* maior que 4.

TABELA 6.6 – Percentual de redes para diferentes *fanouts* por circuito

Circuitos	Comp. Totl das Redes	% REDES Fanout=0	% REDES Fanout=1	% REDES Fanout=2	% REDES Fanout=3	% REDES Fanout=4	% REDES Fanout=5	% REDES Fanout=6	% REDES Fanout >6
cmb_a	48	8,33	87,50	4,17					
parity_a	75	1,33	80,00	18,67					
pcler8_a	127	13,39	79,53	0,79	4,72				1,57
frgl_a	151	1,99	95,36	2,65					
count_a	174	9,20	73,56	9,20	8,05				
comp_a	198	0,51	79,80	15,66	2,02	2,02			
5xp1	299	0,33	90,64	3,34	3,01	1,00	1,00	0,67	
C432a	240	1,67	73,33	21,67	0,83	0,83	0,42		1,25
9sym	409	0,24	92,42	2,20	1,47	0,98	1,47	0,49	0,73
9sym_3x3	339	0,29	74,93	4,72	5,60	5,60	8,85		
bw	469	4,05	85,07	3,84	1,71	2,35	1,28	0,43	1,28
C880a	423	5,91	74,94	10,17	4,49	2,36	1,42		0,71
alu2_a	469	1,07	82,52	4,05	4,90	2,99	1,07	1,07	2,35
C499a	605	5,29	67,11	22,31	0,83	2,15	0,33	0,66	1,32
x4_a	546	12,09	82,42	1,28	0,37	0,55	0,37	0,55	2,38
c1355	604	5,30	57,28	33,11		2,98			1,32
C1908a	728	3,43	72,12	16,35	2,06	3,16	0,14	0,27	2,47

Com estas informações é possível elaborar uma tabela de referência (Tabela 6.7), para estimativa do comprimento médio das interconexões de diferentes redes com diferentes *fanouts* e variável com o número de transistores do circuito.

TABELA 6.7 – Tabela de referência para estimativa do comprimento médio das redes para tecnologia AMS 0,25 μ m

Número de Transistores	fanout=0 Comprimento Médio	fanout=1 Comprimento Médio	fanout=2 Comprimento Médio	fanout=3 Comprimento Médio	fanout=4 Comprimento Médio	fanout=5 Comprimento Médio	fanout=6 Comprimento Médio	fanout > 6 Comprimento Médio
0-800	20-40	30-40	40-60	60-80	80	150	200	100
801-1700	20-40	35-45	60-80	80-120	120-160	150-200	200	200
1701-2500	20-80	40	60-80	80-120	120-160	150-200	200	300-500
2501-3000	70-80	40	80	80-120	120-160	150-200	200	300-500

Com relação às **entradas primárias** dos circuitos, a estimativa de comprimento médio destas interconexões é estabelecida pela Tabela 6.8, que relaciona o *fanin* para cada entrada do circuito.

TABELA 6.8 – Tabela de referência para estimativa do comprimento médio das entradas para tecnologia AMS 0,25 μ m

	Fanin "0"	Fanin "1"	Fanin "2"	Fanin "3"	Fanin "4"	Fanin "5"	Fanin "6"	Fanin 6 à 10	Fanin 11 à 14	Fanin 15 à 21	Fanin maior 21
Comp. Médio (microm)	30	35	50	70	95	150	200	300	500	600	850

Observe pela Tabela 6.8, que à medida que o *fanin* das entradas aumenta, o comprimento médio das redes aumenta, no entanto não existe uma linearidade entre estas variáveis. Estes valores de comprimento médio foram obtidos dos leiautes dos próprios circuitos analisados estabelecendo uma relação com as demais redes do circuito.

6.3.2 Tecnologia AMS 0,35 μ m

Os mesmos procedimentos usados na tecnologia 0,25 μ m na obtenção dos dados são adotados para a tecnologia 0,35 μ m.

A Tabela 6.9 mostra os circuitos usados na geração do leiaute com suas características como número de bandas, largura e comprimento dos blocos gerados para a tecnologia AMS 0,35 μ m. A Tabela 6.10 apresenta os comprimentos médios dos circuitos cujas redes possuem *fanouts* com valores iguais a 0, 1 e 2.

TABELA 6.9 – Dados usados na geração dos leiautes dos circuitos para tecnologia AMS 0,35 μ m

Nome do Circuito	Número de Bandas	Largura (microm)	Comp. (microm)
cmb_a	4	87,45	78,8
parity_a	5	84,25	95,7
pcler8_a	6	116,25	120,6
frg1_a	6	148,8	130,2
count_a	7	143,45	148,7
comp_a	7	161,29	147,1
5xp1	8	177,6	170,4
C432a	8	185,05	183,2
9sym	9	214,9	198,5
bw	10	223,45	233
C880a	10	257,6	247,4
alu2_a	10	284,25	271,4
C499a	12	283,2	279,6
x4_a	12	294,4	305,2
c1355	12	298,65	282,8
C1908a	13	340,25	333,3

TABELA 6.10 – Comprimento médio (μ m) das redes para *fanouts* iguais a 0, 1 e 2 (tecnologia 0,35 μ m)

Nome do Circuito	Número de Trans.	Redes c/ Fanout "0"	Comp. Médio (micra)	Redes c/ Fanout "1"	Comp. Médio (micra)	Redes c/ Fanout "2"	Comp. Médio (micra)
cmb_a	192	4	41	42	51,34	2	76,3
parity_a	240	1	24,3	60	44,2	14	76,37
pcler8_a	412	17	59,53	101	50,26	1	66,3
frg1_a	536	3	32,36	144	54,04	4	100,82
count_a	602	16	43,28	128	60,95	16	64,63
comp_a	656	1	24,8	158	57,36	31	93,94
5xp1	798	1	26,2	271	47,13	10	94,01
C432a	902	4	46,25	175	60,63	52	110,4
9sym	1092	1	71,3	378	50,91	9	110,8
bw	1256	19	33,93	398	50,47	18	92,14
C880a	1552	25	95,4	318	69,57	43	102,66
alu2_a	1712	5	23,68	387	72,84	19	112,58
C499a	2038	32	90,12	407	63,25	135	116,64
x4_a	2160	66	19,55	449	63,62	7	73,97
c1355	2244	32	128,96	34	61,89	200	106,53
C1908a	2690	25	94,99	525	72,2	119	143,86

As Figuras 6.10, 6.11 e 6.12 mapeiam os dados da Tabela 6.10 para curvas que relacionam comprimento médio versus numero de transistores.

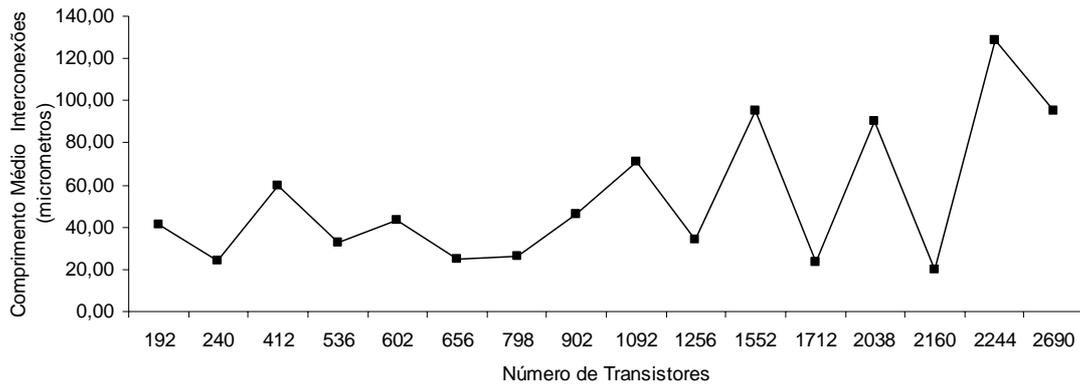


FIGURA 6.10 – Comp. médio das redes versus núm. de trans. para *fanout* igual a ZERO (tecnologia AMS 0,35 μ m)

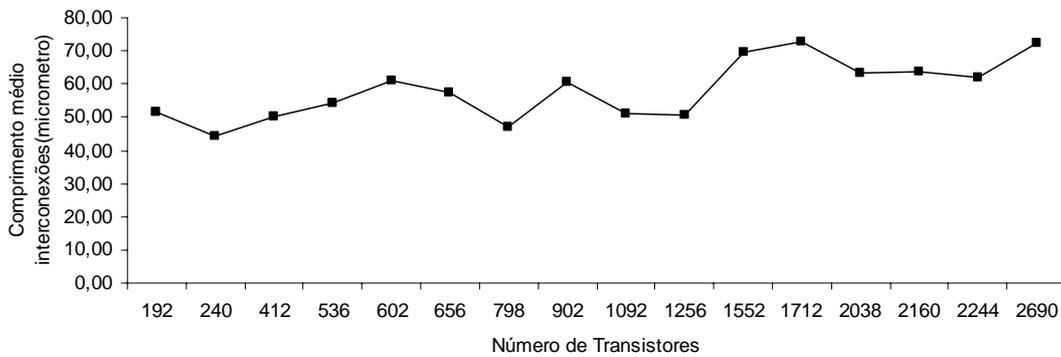


FIGURA 6.11 – Comp. médio das redes versus núm. de trans. para *fanout* igual a UM (tecnologia AMS 0,35 μ m)

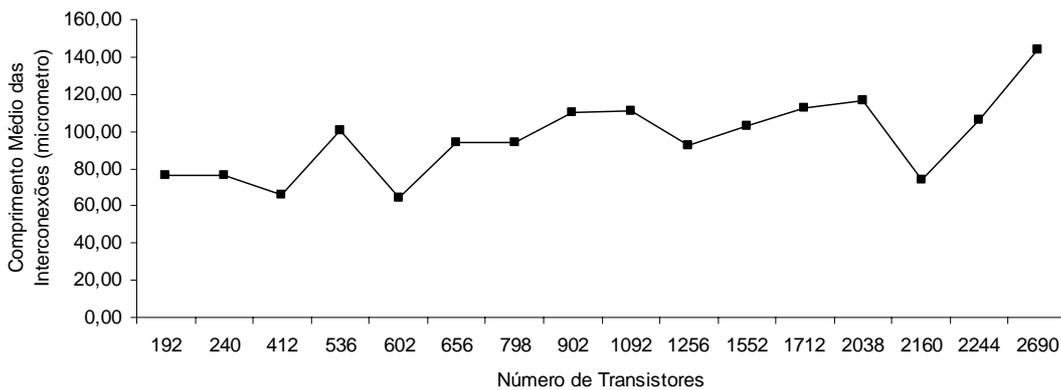


FIGURA 6.12 – Comp. médio das redes versus núm. de trans. para *fanout* igual a DOIS (tecnologia AMS 0,35 μ m)

A Tabela 6.11 apresenta os comprimentos médios dos circuitos cujas redes possuem *fanouts* com valores iguais a 3, 4 e 5.

TABELA 6.11 – Comprimento médio (μm) das redes para *fanouts* iguais a 3, 4 e 5 (tecnologia 0,35 μm)

Nome do Circuito	Número de Trans.	Redes c/ Fanout "3"	Comp. Médio (micra)	Redes c/ Fanout "4"	Comp. Médio (micra)	Redes c/ Fanout "5"	Comp. Médio (micra)
cmb_a	192	0		0		0	
parity_a	240	0		0		0	
pcler8_a	412	6	101,26	0		0	
frg1_a	536	0		0		0	
count_a	602	14	110,05	0		0	
comp_a	656	4	96,32	4	137,42	0	
5xp1	798	9	126	3	223,73	3	192,4
C432a	902	2	263,35	2	262,05	1	307,1
9sym	1092	6	136,05	4	189,2	6	291,6
bw	1256	8	142,31	11	283,1	6	413,25
C880a	1552	18	191,72	10	199,85	6	231,3
alu2_a	1712	23	159,18	14	233,64	5	234,7
C499a	2038	5	103,88	13	175,95	2	223,7
x4_a	2160	2	108,3	3	286,1	2	278,1
c1355	2244	0		18	186,2	0	
C1908a	2690	15	202,47	23	194,53	1	188,6

As Figuras 6.13, 6.14 e 6.15 mapeiam os dados da Tabela 6.11 para curvas que relacionam comprimento médio versus número de transistores.

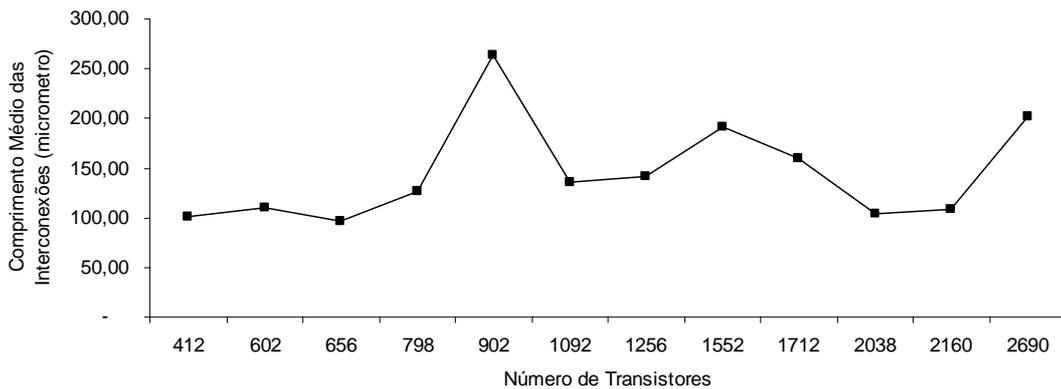


FIGURA 6.13 – Comp. médio das redes versus núm. de trans. para *fanout* igual a TRÊS (tecnologia AMS 0,35 μm)



FIGURA 6.14 – Comp. médio das redes versus núm. de trans. para *fanout* igual a QUATRO (tecnologia AMS 0,35µm)

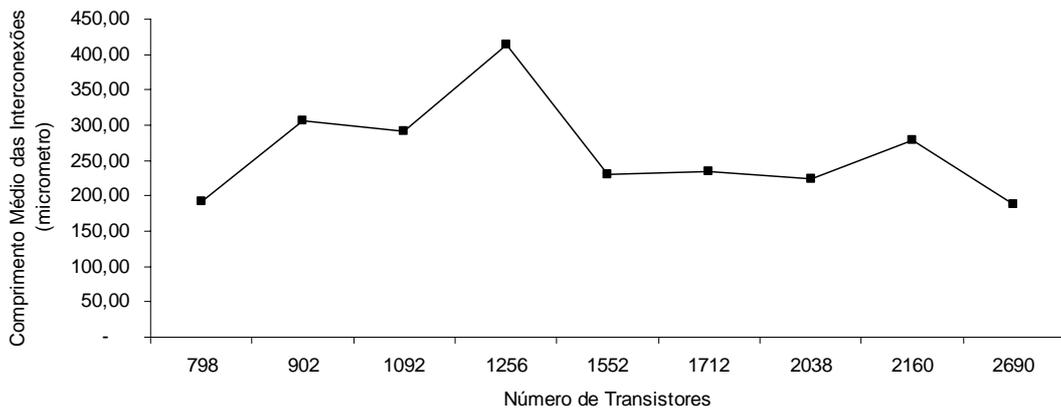


FIGURA 6.15 – Comp. médio das redes versus núm. de trans. para *fanout* igual a CINCO (tecnologia AMS 0,35µm)

A Tabela 6.12 apresenta os comprimentos médios dos circuitos cujas redes possuem *fanouts* com valores iguais a 6 e maiores.

TABELA 6.12 – Comprimento médio (μm) das redes para *fanouts* iguais a 6 e maiores (tecnologia 0,35 μm)

Nome do Circuito	Número de Trans.	Redes c/ Fanout "6"	Comp. Médio (micra)	Redes c/ Fanout "maior que 6"	Comp. Médio (micra)
cmb_a	192	0		0	
parity_a	240	0		0	
pcler8_a	412	0		2	321,75
frg1_a	536				
count_a	602				
comp_a	656				
5xp1	798	2	347,6		
C432a	902	0		3	874,2
9sym	1092	2	248,8	3	381,7
bw	1256	2	452,55	6	552,88
C880a	1552	0	0	3	540,16
alu2_a	1712	5	413,04	11	672,6
C499a	2038	3	347,23	8	479,75
x4_a	2160	2	232,7	14	690,59
c1355	2244	0		8	713,15
C1908a	2690	2	327,25	18	559,84

As Figuras 6.16 e 6.17 mapeiam os dados da Tabela 6.12 para curvas que relacionam comprimento médio versus número de transistores.

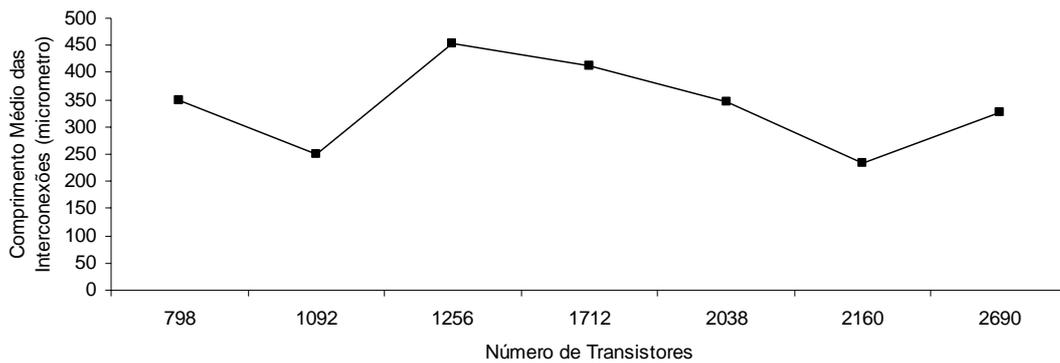


FIGURA 6.16 – Comp. médio das redes versus núm. de trans. para *fanout* igual a SEIS (tecnologia AMS 0,35 μm)

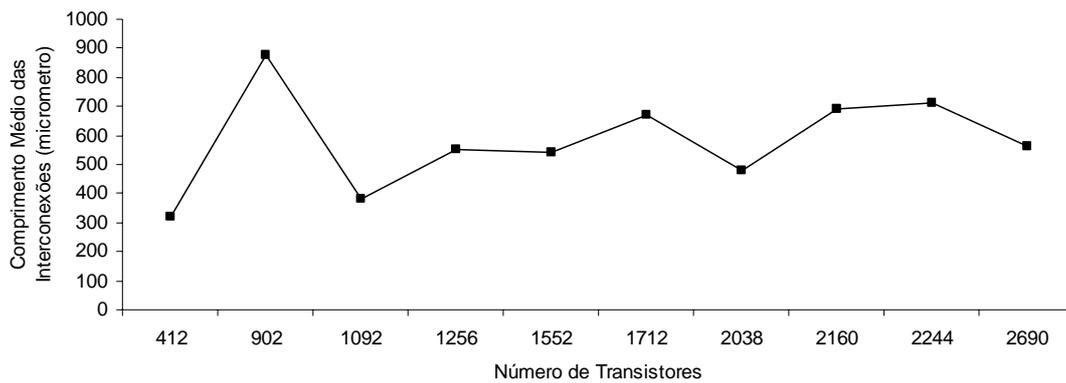


FIGURA 6.17 – Comp. médio das redes versus núm. de trans. para *fanout* maior do que SEIS (tecnologia AMS 0,35 μ m)

Como resultado, obtém-se a Tabela 6.13, que será usada no dimensionamento do comprimento das redes de circuitos CMOS para a tecnologia AMS 0,35 μ m, e por conseguinte, estimativa das capacitâncias e potência associada as interconexões referenciadas ao terra.

TABELA 6.13 – Tabela de referência para estimativa do comprimento médio das redes para tecnologia AMS 0,35 μ m

Número de Transistores	fanout=0 Comprimento Médio	fanout=1 Comprimento Médio	fanout=2 Comprimento Médio	fanout=3 Comprimento Médio	fanout=4 Comprimento Médio	fanout=5 Comprimento Médio	fanout=6 Comprimento Médio	fanout > 6 Comprimento Médio
0-800	20-60	40-60	60-100	100-150	150-250	200-400	250-400	300-800
801-1700	40-80	50-70	80-100	150-250	150-250	200-400	250-400	300-800
1701-2500	50-100	50-70	80-100	150-250	150-250	200-400	250-400	300-800
2501-3000	100-120	50-70	140	150-250	150-250	200-400	250-400	300-800

Analisando as duas tecnologias AMS 0,25 μ m e AMS 0,35 μ m, observa-se uma linearidade nos dados obtidos nas Tabelas 6.7 e 6.13, onde as variações ocorrem devido apenas a tecnologia considerada. Ou seja, para tecnologias maiores, maiores serão os comprimentos médios das redes. Isto nos garante que o método apresentado é aceitável.

6.4 Avaliação da metodologia para obtenção do comprimento médio para a tecnologia 0,25 μ m

O método proposto na seção anterior necessita ser validado. Os procedimentos adotados para esta validação consistem na estimativa do comprimento médio das interconexões de vários circuitos, comparando estes resultados com os obtidos pela geração do leiaute. Num primeiro momento a comparação foi feita com os circuitos usados para levantamento dos dados e num segundo momento com outros circuitos obtidos de *benchmarks*.

Os resultados desta verificação são apresentados nas Tabelas 6.14 e 6.15.

Pela Tabela 6.14, constata-se que o maior erro ocorre para o circuito x4_a, 25,99% enquanto que no circuito comp_a este erro é quase desprezível.

TABELA 6.14 – Comprimento médio de redes para circuitos *benchmarks* obtidos da extração elétrica versus estimativa lógica (tecnologia 0,25 μ m)

Nome do Circuito	Número de Trans.	Comp. Médio Ext (micrometro)	Comp. Médio Estimado (micrometro)	Erro (%)
cmb_a	192	33,23	35,32	6,30
parity_a	240	32,72	37,73	15,32
pcler8_a	412	37,68	38,14	1,20
frg1_a	536	36,05	35,48	-1,50
count_a	602	41,23	40,8	-1,03
comp_a	656	40,13	40,42	0,70
5xp1	798	37,37	41,38	10,70
C432a	902	55,01	50,56	-8,09
9sym	1092	40,97	45,13	10,10
9sym_3x3	1156	72,44	58,11	-19,70
bw	1256	45,04	49,09	8,90
C880a	1552	54,98	52,32	-4,80
alu2_a	1712	59,94	68,51	14,29
C499a	2038	54,45	61,1	12,20
x4_a	2160	46,92	59,12	25,99
c1355	2244	58,8	60,15	2,20
C1908a	2690	63,9	73,61	15,10

TABELA 6.15 – Análise do método para outros circuitos *benchmarks* (tecn. 0,25 μ m)

Nome do Circuito	Número de Trans.	Comp. Médio Ext (micrometro)	Comp. Médio Estimado (micrometro)	Erro (%)
cm85_a	175	36,69	39,69	7,56
rd53	188	33,10	39,35	15,88
pm1_a	196	36,72	42,50	13,60
z4ml	294	36,65	41,97	12,68
misex1	394	33,78	40,56	16,72
cordic_a	430	38,12	37,82	-0,79
f51m	656	37,00	39,49	6,31
duke2	2638	48,03	54,22	11,42
x3	3776	56,46	64,79	12,86
alu4	3410	74,89	69,95	-7,06
apex6	3228	70,57	66,21	-6,59
apex7	1086	49,21	59,5	17,29
cm163a	204	33,41	43,88	23,86
cu	258	36,58	43,52	15,95
i2	712	35,95	37,14	3,20
i5	1510	48,9	46,31	-5,59
rot	2814	65,28	61,92	-5,43
term1	1984	46,08	51,15	9,91
x1	1574	49,30	47,82	-3,09

Para outros circuitos *benchmarks*, conforme apresentado na Tabela 6.15, o maior erro ocorre no circuito cm163a que é de 23,86%, quando comparado ao valor obtido pela geração do leiaute.

Desta forma comprova-se plenamente a validade do método, embora saiba-se que como trata-se de comprimentos médios sempre haverá valores extremos que podem apresentar divergências consideráveis entre estimativa e extração. Um fato

importante a considerar é o objetivo aqui é estimar rapidamente os comprimentos médios sem haver a necessidade da geração do leiaute e extração das capacitâncias parasitas.

6.5 Capacitâncias devido às Interconexões

O comprimento médio das interconexões que pode ser obtido por estimativa pelo uso das Tabelas 6.7 e 6.8 considerando a tecnologia AMS 0,25 μm , possibilitam juntamente com dados de tecnologia fornecido pela *foundry*, estimar as capacitâncias decorrentes das interconexões.

A expressão que representa as capacitâncias de interconexões é definida como:

$$C_{inter} = CaME * LME * wirelength + 2 * (CpME * wirelength) \quad (101)$$

onde:

C_{inter} = Capacitância da interconexão

$CaME$ = Capacitância por Unidade de Área

LME = Largura do metal

$wirelength$ = Comprimento da fiação

$CpME$ = Capacitância por Unidade Linear

Os valores de $CaME$, $CpME$ e LME são fornecidos pela tecnologia de fabricação, enquanto que o comprimento da fiação em cada nodo (rede) do circuito é obtido das Tabelas 6.7 e 6.8.

6.6 Potência devido às Interconexões

Com os dados de capacitância das interconexões, atividade de comutação em cada nodo, tensão de alimentação e frequência de operação do circuito é possível estimar a potência dissipada pelo circuito devido às interconexões. A potência dinâmica em um circuito CMOS foi definida no Capítulo 2 pela equação 3. A adaptação desta equação para as interconexões é a seguinte:

$$P_{interc} = 0.5 f_c V_{dd}^2 \sum_{i=1}^n C_{inter} \alpha_i(x_i) \quad (102)$$

onde:

P_{interc} é igual à estimativa da potência média devido às interconexões

f_c é a frequência de operação

V_{dd} é a tensão de alimentação do circuito

C_{inter} é a estimativa de capacitância devido às interconexões

α_i é a atividade de comutação associada ao nodo (ou rede) “i”

x_i é o nodo (ou rede) do circuito no nível de portas lógicas

6.7 Avaliação da Estimativa de Capacitâncias de Interconexões

O método proposto de estimativa de capacitâncias das interconexões é validado nesta seção. Os circuitos *benchmarks* usados para este propósito são apresentados na Tabela 6.16, dos quais foram gerados os leiautes e extraídas as capacitâncias de interconexões. Neste caso, considerou-se como parâmetro de comparação o somatório das capacitâncias de interconexões obtidas da estimativa proposta pelo método versus o somatório das capacitâncias extraídas do leiaute. A análise feita considera a tecnologia da AMS 0,25 μ m. As Tabelas 6.7 e 6.8 são usadas para esta análise.

A Tabela 6.16 apresenta na quarta coluna, o resultado das comparações entre estes somatórios e na quinta coluna, a potência média consumida pelo circuito devido a estas interconexões. Com exceção do circuito *parity_a* que apresentou um erro de 25,9% os erros variaram de 2,2% (*comp_a*) a 9,5% (*p1cer8_a*), sendo perfeitamente aceitável pelo nível de abstração em que se está trabalhando.

A potência estimada no nível lógico não é comparada com a simulação elétrica devido ao grande número de entradas, ou seja, o número de vetores de entradas a ser analisado é proporcional a 2^{2n} , onde n é o número de entradas do circuito. Por exemplo, um circuito com 10 entradas seriam necessários 1048576 combinações, o que é extremamente difícil de simular.

TABELA 6.16 – Comparação das capacitâncias estimadas ao nível lógico com capacitâncias extraídas do leiaute

Nome Circuito	Somatório das Capac. Rot. Ext. Elét. (fF)	Somatório das Capac. Rot. Est. Lógica(fF)	Erro (%)	Potência das Interconexões (microW)
<i>comp_a</i>	907,82	888,21	2,2	93,16
<i>cmb_a</i>	248,35	240,3	3,3	24,70
<i>parity_a</i>	288,64	363,52	25,9	41,41
<i>frg1_a</i>	831,29	775,6	7,1	83,33
<i>count_a</i>	911,15	848,63	7,3	64,59
<i>C432a</i>	1412,21	1354,14	4,2	102,06
<i>alu2_a</i>	3087,27	2992,26	3,1	237,02
<i>p1cer8_a</i>	599,81	612,48	2,1	51,79
<i>C1908a</i>	4750,43	4427,2	7,3	388,61

A Tabela 6.17, demonstra a velocidade de processamento que um estimador de potência trabalha neste nível. Observe a altíssima velocidade de processamento quando se compara o método proposto de obtenção das capacitâncias com a geração e extração usando uma ferramenta de CAD. A razão de tempo pode chegar a 30 vezes (circuito *bw* com 1256 transistores). Esta velocidade de processamento dos dados, possibilita ao projetista uma rápida tomada de decisão, objetivo maior do método proposto

TABELA 6.17 – Tempos de Processamento: Extração das Capacitâncias *versus* Estimativa Lógica

Nome do Circuito	TROPIC3			Estimador Lógico tempo(ms)
	Geração (ms)	Extração (ms)	Total (ms)	
cmb_a	987	2933	3920	440
parity_a	859	1460	2319	440
pcler8_a	1319	1945	3264	330
frg1_a	1583	2177	3760	330
count_a	1744	2326	4070	380
comp_a	1859	2459	4318	380
5xp1	2353	3036	5389	380
C432a	2560	3057	5617	390
9sym	3342	3895	7237	440
bw	3896	4413	8309	270
alu2_a	5589	5169	10758	490
c1355	6291	6706	12997	770
C1908a	8155	7807	15962	880

6.8 Estimativa da Potência Total

Esta seção trata de estabelecer um método de estimativa de consumo médio de um circuito CMOS por completo. A estimativa da potência média dissipada usa as estimativas de **capacitâncias de porta** e de **capacitâncias de interconexões**, modeladas nos capítulos 5 e 6, respectivamente. A Tabela 6.18 apresenta esta composição.

Nesta simulação usou-se tensão de alimentação igual a 5V, frequência de operação igual a 20MHz e atividades de comutação das portas com atraso zero.

Na estimativa das capacitâncias de porta, usou-se dimensões de transistores padrões obtidos pela média dos valores gerados pelo leiaute. A tecnologia de fabricação 0,25 μ m da AMS foi usada na geração do leiaute.

O leiaute foi gerado pela ferramenta TROPIC3 e as capacitâncias extraídas pelo EXT [MOR 93].

TABELA 6.18 – Estimativa da potência total de circuitos *benchmarks* ao nível lógico

Nome Circuito	Potência de Porta (microW)	Potência das Interconexões (microW)	Potência Total (microW)
comp_a	183,18	93,16	276,34
cmb_a	37,27	24,70	61,97
parity_a	100,79	41,41	142,20
frg1_a	128,05	83,33	211,38
count_a	124,79	64,59	189,38
C432a	173,30	102,06	275,36
alu2_a	257,43	237,02	494,45
pcler8_a	86,09	51,79	137,88
C1908a	440,27	388,61	828,88

Analisando a Tabela 6.18, observa-se que a potência de porta em todos os casos analisados foi maior do que as interconexões, ou seja os fios que interligam as várias portas do circuito estão otimizados com relação ao posicionamento entre células. Lembrando sempre que o nosso estudo não envolve estimativa de capacitâncias de *crosstalk*.

6.9 Estimativa do Consumo de Potência de Portas Complexas

Esta seção faz uma breve análise sobre a estimativa em portas complexas sem no entanto entrar na modelagem matemática, propriamente dita.

Com **inversores**, **nand's** e **nor's** podemos implementar qualquer equação lógica, ou qualquer circuito digital. Porém para obtermos um circuito otimizado através da redução no número de transistores podemos utilizar portas complexas que implementam a função desejada sem uso das convencionais citadas acima.

Da mesma forma que foram modeladas portas convencionais para estimar capacitâncias e potência de circuitos digitais CMOS, pode-se fazer o mesmo para portas complexas. O fator limitante nesta modelagem está demonstrada na Tabela 6.19 [REI2000], abaixo. Esta tabela mostra o número possível de células que podem ser geradas em função da altura de uma célula. A altura de uma célula é definida como o número máximo de transistores em série, seja no subcircuito *pull-up* ou no subcircuito *pull-down*. Na célula da Figura 6.18, a altura é 3, pois existem 3 transistores em série no subcircuito *pull-up*.

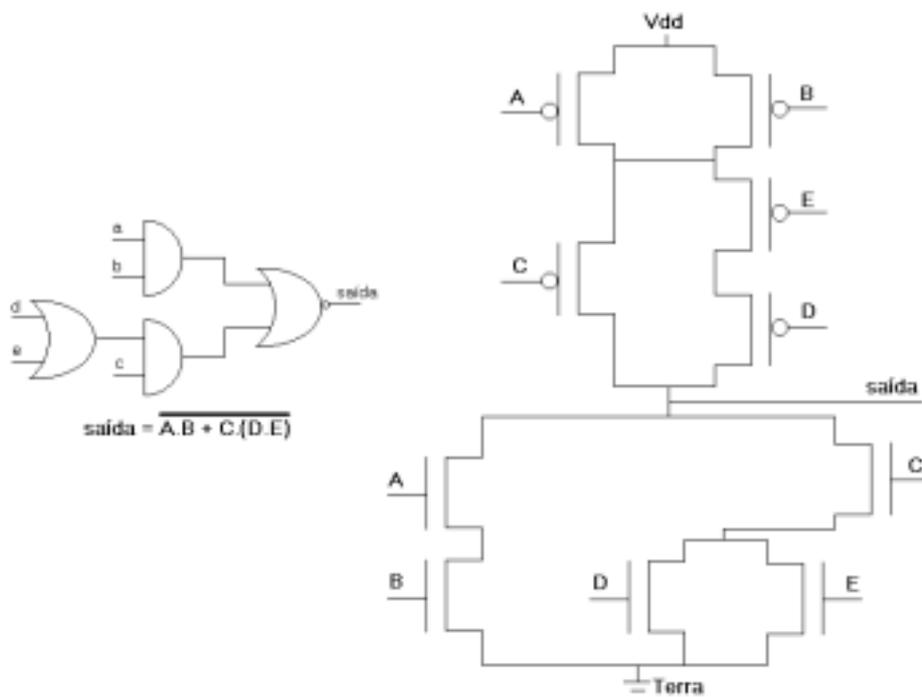


FIGURA 6.18 – Esquemas lógico e elétrico de uma porta complexa

TABELA 6.19 – Número de células possíveis em função da altura da célula

Altura da Célula	Número de Células
1	1
2	7
3	87
4	3503
5	425803
6	154793519

Observe pela Tabela 6.19 a grande quantidade de combinações que podem ser obtidas à medida que aumenta a altura das células. Cada célula originada deverá sofrer modelagem em termos de capacitâncias e combinações de entradas, conforme deduzida no Capítulo 5. Por esta razão, a estimativa de potência em portas complexas não foi tratada nesta Tese.

7 Conclusões

Minimizar o consumo de potência em circuitos CMOS VLSI deve ocorrer em todos os níveis hierárquicos, desde o nível de processo, circuitos lógicos, arquitetura, algoritmo até o sistema. Em cada um destes níveis existem ações que visam justamente reduzir esta dissipação de potência.

Entretanto, para se propor o uso de técnicas de redução de potência é necessário e imprescindível a estimativa do consumo de potência do circuito em estudo, de maneira confiável, rápida e precisa. A potência dinâmica é responsável por 90% do consumo total de circuitos combinacionais CMOS e depende da tensão aplicada, frequência de operação, capacitâncias parasitas e atividades de comutação das portas.

Esta Tese apresentou um método de estimativa de potência de porta e de interconexões no nível lógico. Para o nível de porta, foi desenvolvido um modelo de capacitância de porta que leva em consideração as combinações dos vetores de entrada. Para o nível de interconexões, as capacitâncias foram estimadas a partir da estimativa do comprimento médio das interconexões também no nível lógico em função do fanout e fanin de cada porta lógica.

As considerações e os modelos desenvolvidos basearam-se em considerações de que a tensão de alimentação, frequência de operação e atividades de comutação são valores conhecidos e definidos. Além de que, o método é usado para circuitos combinacionais, com portas lógicas CMOS convencionais, do tipo **inversor**, **nand's** e **nor's**.

Os resultados apresentados através da simulação no nível de transistores validam os métodos, tanto na estimativa da capacitância de porta quanto das capacitâncias das interconexões. Os erros ocorridos são devidos a (ao):

- aproximações feitas na modelagem das capacitâncias intrínsecas;
- método de estimativa de capacitância de interconexão ser obtido a partir do comprimento médio das interconexões;
- Ao modelo de transistor adotado pelo SPICE;
- Estimativa da potência com atraso zero.

O método proposto para potência das interconexões foi validado pela simulação de vários circuitos (Tabelas 6.14 e 6.15). Os erros encontrados na maioria dos circuitos foram aceitáveis para o nível lógico, pois neste caso não temos informações precisas da tecnologia, como capacitâncias extraídas do leiaute e mapeamento de transistores. O tempo de processamento observado na Tabela 6.17 justifica de sobremaneira o método, pois facilita enormemente a decisão do projetista com relação a potência dissipada, na confecção de um sistema, antes mesmo da geração do seu leiaute.

7.1 Trabalhos Futuros

Dando continuidade a esta Tese, alguns trabalhos futuros podem ser desenvolvidos como:

- generalizar o método de estimativa das capacitâncias de interconexões à outras ferramentas de leiaute automático;

- considerar atrasos unitários e genéricos nas portas lógicas do circuito;
- estender o método de estimativa das **capacitâncias de porta** para circuitos sequenciais;
- desenvolver estimativa de **capacitâncias de porta** para portas lógicas complexas;
- implementar o método do semi-perímetro na estimativa do comprimento das interconexões e das capacitâncias de interconexões;
- desenvolver um ambiente de CAD, visando a integração de várias ferramentas nos mais variados níveis, desde a análise de capacitâncias ao nível de leiaute até o nível lógico;
- estender o método de estimativa de capacitâncias incluindo a estimativa de capacitâncias de *crosstalk* .

Anexo 1

Artigo publicado no International Conference on Chip Design Automation
(ICDA 2000), 21-25 August, 2000, Beijing, China.

Capacitance and Power Modeling at Logic-Level

João Baptista Martins^{*}, Ricardo Reis

{batista,reis}@inf.ufrgs.br

II, UFRGS, Porto Alegre, Brazil

José Monteiro

jcm@inesc.pt

IST / INESC, Lisboa, Portugal

Abstract

Accurate and fast power estimation of CMOS circuits during the design phase is required to guide power optimization techniques employed to meet stringent power specifications. Logic-level power estimation tools, such as those available in the SIS and POSE frameworks are able to accurately calculate the switching activity under a given delay model. However, capacitance and delay modeling is crude. The objective of the work described in this paper is to investigate how close can logic-level power estimates get to estimates obtained with circuit-level simulators such as SPICE. We propose new models for the input and output capacitance's of logic gates, taking into account the gate's internal capacitances and the interconnect capacitances extracted from layout. The results we present show an agreement of the logic-level estimates and SPICE, with less than 10% error.

Keywords

Power consumption, capacitance modeling, CAD, IC Design

Introduction

Power consumption is becoming one of the most important design challenges in the design of VLSI circuits. Optimization techniques for low power are being employed at all design levels of abstraction. In order to guide designers and optimization tools, there is a pressing need for accurate and fast power estimation tools.

During normal operation, the power dissipation of a CMOS circuit is directly related to the switching activity. For a well designed circuit, the total average power can be approximated by the switched-capacitance power [5]. This is an underlying assumption of almost all of the available power estimation tools at the logic *gate* and higher levels of abstraction. The average dynamic power consumption of a CMOS circuit is then given by:

$$Power = \frac{1}{2} \cdot f \cdot V_{dd}^2 \cdot \sum_{i=1}^n C_i \cdot \alpha_i \quad (1)$$

Where f is the clock frequency, V_{dd} is the supply voltage, C_i and α_i are capacitance load and average switching of the logic *gate* i , respectively.

Significant amount of work has been carried out in developing efficient techniques to estimate the switching activity of a CMOS circuit [9]. These techniques can be divided into two classes: statistical techniques (also known as dynamic techniques), e.g. [1]; probabilistic (or static) techniques, e.g. [2].

Statistical techniques simulate the circuit repeatedly until the power values converge to an average power, based on some statistical measures. Probabilistic techniques propagate input statistics through the circuit to obtain the switching probability for each *gate* in the circuit. Probabilistic techniques are employed in the power estimation tools inside SIS [3] and POSE [4].

In this paper, we focus on the problem of capacitance modeling. This is typically done very simplistically at the logic level. We propose to build a more accurate model. At the logic-level, only the input and output nodes of the *gates* are available. We present an equivalent load capacitance model for each external node, computed from the internal transistor capacitances and the interconnection capacitance.

The interconnection capacitance is currently being extracted from layout information. We have performed several experiments on (necessarily) simple circuits where power estimates were obtained both using SPICE, and SIS using the capacitance values obtained through our model. The results are very promising, with power differences of less than 10%.

This paper is organized as follows. In Section 2, we briefly present the capacitance model used for a transistor MOSFET. In Section 3, we describe the methodology for calculating the equivalent

^{*} This researcher was supported by UFSM and CAPES/Brazil.

capacitances associated the input and output nodes of logic *gates*. In Section 4, we describe the experimental setup and the tools used. Experimental results are presented in Section 5. In Section 6, we give some conclusions and discuss future research.

MOSFET Capacitance Model

The MOSFET transistors exhibit a number of parasitic capacitance [6] (Figure 1), which must be accounted for in circuit design: *gate-to-source* capacitance (C_{GS}), *gate-to-drain* capacitance (C_{GD}), *gate-to-bulk* capacitance (C_{GB}), *source-to-bulk* capacitance (C_{SB}) and *drain to bulk* capacitance (C_{DB}). In this work, our interest will be the switching region of the transistors and all the capacitances will be considered, with the exception of C_{GB} .

Although these capacitances are a nonlinear function of the voltage, the general approach is to assume them as a linear, time-invariant element. Given some technology parameters and the size the transistor (W/L), a value for each of these parasitic capacitances can be computed [6].

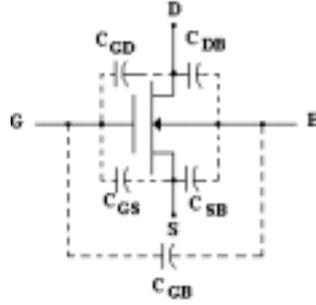


Figure 1- Parasitic Capacitances of the MOSFET

For definition [6]:

$$C_G = C_{ox}LW + 2C_o \quad (2)$$

$$C_{GD} = C_{GS} = \frac{1}{2}C_{ox}(WL) + C_o \quad (3)$$

$$C_{DB,bot} = K(V_l)C_{j0}A_D \quad (4)$$

$$C_{DB,sw} = K_{1/3}(V_l)C_{jsw}l_D \quad (5)$$

$$C_{DB} = C_{DB,bot} + C_{DB,sw} \quad (6)$$

$$C_{SB,bot} = K(V_l)C_{j0}A_S \quad (7)$$

$$C_{SB,sw} = K_{1/3}(V_l)C_{jsw}l_S \quad (8)$$

$$C_{SB} = C_{SB,bot} + C_{SB,sw} \quad (9)$$

Where:

L = channel length

W = channel width

C_o = overlap capacitance

C_{j0} = zero bias capacitance per unit area

C_{jsw} = zero-bias sidewall capacitance per unit perimeter

A_D = Area of drain

l_D = perimeter of drain

A_S = Area of source

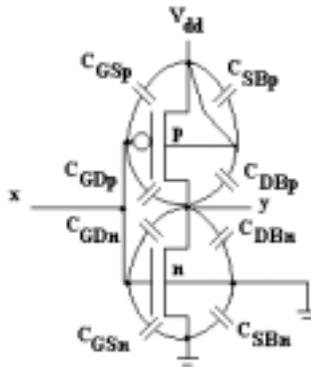
L_S = perimeter of source

Modeling Gate Input and Output Capacitances

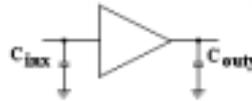
For circuits described at the logic level there is only access to the logic *gates* input and output nodes. Therefore, all capacitances internal to the *gate* have to be taken into account in concentrated capacitance models at these external nodes. In this section we describe how we compute the equivalent capacitance load for any logic *gate* from its transistor level description.

A. CMOS Inverter

The CMOS inverter (Figure 2) presents two external nodes. In this case, equivalent capacitances at the input x , $C_{in\ x}$, and output y , $C_{out\ y}$, nodes can be simply computed using:



(a)



(b)

Figure 2–Parasitic capacitances of CMOS inverter

- (a) Transistor-level representation
- (b) Logic-level representation

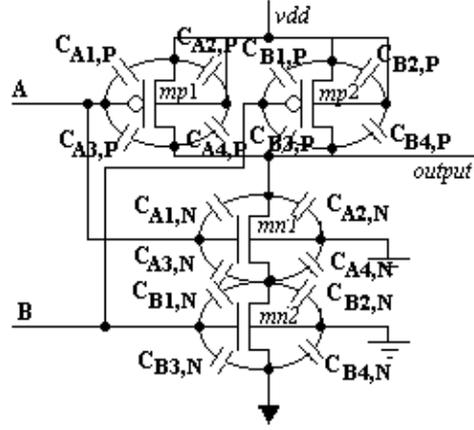
$$C_{inx} = C_{GSp} + C_{GDp} + C_{GSn} + C_{GDn} \quad (10)$$

$$C_{outy} = C_{DBp} + C_{GDp} + C_{GDn} + C_{DBn} \quad (11)$$

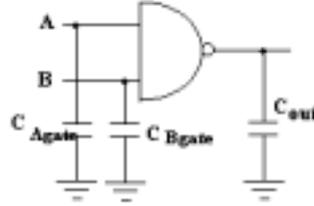
B. 2 -Input CMOS NAND Gate

In the case of a 2-input CMOS NAND *gate* (figure 3), there are two inputs nodes with concentrate capacitance and one output node. The additional difficulty is that now we need to do an internal capacitance modeling. We analyze the equivalent capacitance from superposition of the signals. For example, when calculate the equivalent capacitance to a input node, the inputs and output will be in the ground level.

The capacitance of internal nodes depends on the logic value of the others inputs. To model this effect we are assuming a probability of 0.5 that all inputs are set to 1. That is, the *NMOS* transistors will be ON at half the time.



(a)



(b)

Figure 3– Parasite Capacitances of 2-input NAND

(a) Transistor-level representation

(b) Logic-level representation

In the NAND2 case, the possible input combination is 00, 01, 10 and 11.

Input “a”:

For input combinations 00 and 10, the “mn2” transistor is OFF and the equivalent capacitance node (C_{a1}) is equal:

$$C_{a1} = 0.5 \left(C_{A1,N} + \frac{C_{A3,N} * C_{T,N}}{C_{A3,N} + C_{T,N}} \right) \quad (12)$$

$$C_{T,N} = C_{A4,N} + C_{B1,N} + C_{B2,N} \quad (13)$$

For input combinations 01 and 11, the mn2 transistor is ON and the equivalent capacitance (C_{a2}) is equal to:

$$C_{a2} = 0.5(C_{A1,N} + C_{A3,N}) \quad (14)$$

Then,

$$C_{A,N} = C_{a1} + C_{a2} \quad C_{A,N} = 0.5 \left(C_{A1,N} + \frac{C_{A3,N} * C_{T,N}}{C_{A3,N} + C_{T,N}} \right) + 0.5(C_{A1,N} + C_{A3,N}) \quad (15)$$

In terms of probability:

$$C_{A,N} = p_0 \left(C_{A1,N} + \frac{C_{A3,N} * C_{T,N}}{C_{A3,N} + C_{T,N}} \right) + p_1 (C_{A1,N} + C_{A3,N}) \quad (16)$$

and C_{Agate} is:

$$C_{A_{gate}} = p_0 \left(C_{A1,N} + \frac{C_{A3,N} * C_{T1,N}}{C_{A3,N} + C_{T1,N}} \right) + p_1 (C_{A1,N} + C_{A3,N}) + C_{A1,P} + C_{A3,P} \quad (17)$$

Where p_0 is the probability of second input is equal ZERO and p_1 is the probability of second input is equal ONE ($p_0 + p_1 = 1$).

Input “b”

In the same way, we can analyse the second input.

For input combinations 00 and 01, the *mn1* transistor is OFF and C_{b1} is equal to:

$$C_{b1,N} = 0.5 \left(C_{B3,N} + \frac{C_{B1,N} * C_{T1,N}}{C_{B1,N} + C_{T1,N}} \right) \quad (18)$$

For input combinations 10 and 11, the *mn1* transistor is ON and C_{B2} is equal to:

$$C_{b2,N} = 0.5 (C_{B1,N} + C_{B3,N}) \quad (19)$$

Then,

$$C_{B,N} = C_{b1,N} + C_{b2,N} \quad (20)$$

$$C_{B,N} = 0.5 \left(C_{B3,N} + \frac{C_{B1,N} * C_{T1,N}}{C_{B1,N} + C_{T1,N}} \right) + 0.5 (C_{B1,N} + C_{B3,N}) \quad (21)$$

$$C_{T1,N} = C_{A4,N} + C_{A3,N} + C_{B2,N} \quad (22)$$

In terms of probability:

$$C_{B,N} = p_0 \left(C_{B3,N} + \frac{C_{B1,N} * C_{T1,N}}{C_{B3,N} + C_{T1,N}} \right) + p_1 (C_{B1,N} + C_{B3,N}) \quad (23)$$

and $C_{B_{gate}}$ is:

$$C_{B_{gate}} = p_0 \left(C_{B3,N} + \frac{C_{B1,N} * C_{T1,N}}{C_{B1,N} + C_{T1,N}} \right) + p_1 (C_{B1,N} + C_{B3,N}) + C_{B1,P} + C_{B3,P} \quad (24)$$

Where:

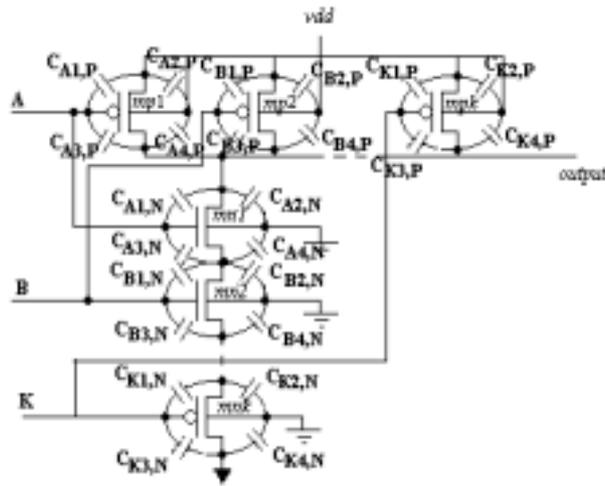
$C_{A_{gate}}$ and $C_{B_{gate}}$ are equivalent capacitances for input A and input B, respectively.

And the output is:

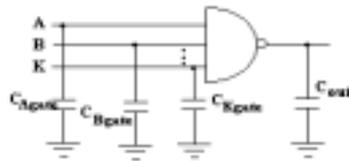
$$C_{out} = C_{A1,N} + C_{A2,N} + C_{A3,P} + C_{A4,P} + C_{B3,P} + C_{B4,P} \quad (25)$$

C. NAND with n -inputs

These considerations are also valid for CMOS NAND's with n inputs, where n is an integer number and probability = 0.5. In this case, one input is switching and the others inputs assume 0 or 1 values. All combinations of input vectors will be analyzed.



(a)



(b)

Figure 4–Parasitic Capacitances of a k-input NAND

(a) Transistor-level representation

(b) Logic-level representation

Initial Considerations:

$$C_{GD(0)} = C_{A1,N}$$

$$C_{DB(0)} = C_{A2,N}$$

$$C_{GS(0)} = C_{A3,N}$$

$$C_{SB(0)} = C_{A4,N}$$

For $k = B$ to K

$$C_{GD(1)} = C_{B1,N} \dots\dots\dots C_{GD(k)} = C_{K1,N}$$

$$C_{DB(1)} = C_{B2,N} \dots\dots\dots C_{DB(k)} = C_{K2,N}$$

$$C_{GS(1)} = C_{B3,N} \dots\dots\dots C_{GS(k)} = C_{K3,N}$$

$$C_{SB(1)} = C_{B4,N} \dots\dots\dots C_{SB(k)} = C_{K4,N}$$

a) Capacitance of the first input of the NMOS net (transistor connected to the output $i = 0$):

$$C(0) = C_{GD(0)} + \frac{2}{2^n} C_{GS(0)} + \frac{1}{2^n} \sum_{q=1}^{n-1} (2^{n-q} \frac{C_{GS(0)} * C_{Ti(q-1)}}{C_{GS(0)} + C_{Ti(q-1)}}) \tag{26}$$

$$C_{Ti(0)} = C_{SB(0)} + C_{GD(1)} + C_{DB(1)} \tag{27}$$

For $q=1$ to $n-2$

$$C_{Ti(q)} = C_{Ti(0)} + \sum_{i=1}^q (C_{GS(i)} + C_{SB(i)} + C_{GD(i+1)} + C_{DB(i+1)}) \tag{28}$$

The equivalent gate capacitance ($C_{A\text{gate}}$) is:

$$C_{A\text{gate}} = C_{(0)} + C_{A1,P} + C_{A3,P} \quad (29)$$

b) Capacitance of intermediate inputs ($0 < i < n-1$):

$$C(i) = \frac{2^{n-i}}{2^n} C_{GD(i)} + \frac{1}{2^n} \sum_{q=1}^i (2^{n-q} \frac{C_{GD(i)} * C_{T_GD(q-1)}}{C_{GD(i)} + C_{T_GD(q-1)}}) + \frac{2^{i+1}}{2^n} C_{GS(i)} + \frac{1}{2^n} \sum_{q=n-1}^{i+1} (2^q \frac{C_{GS(i)} * C_{T_GS(n-1-q)}}{C_{GS(i)} + C_{T_GS(n-1-q)}}) \quad (30)$$

$$C_{T_GD(0)} = C_{DB(i)} + C_{GS(i-1)} + C_{SB(i-1)} \quad (31)$$

For $q=1$ to $i=1$

$$C_{T_GD(q)} = C_{T_GD(0)} + \sum_i^{i-q-1} (C_{GS(i-2)} + C_{SB(i-2)} + C_{GD(i-1)} + C_{DB(i-1)}) \quad (32)$$

$$C_{T_GS(0)} = C_{GD(i+1)} + C_{SB(i)} + C_{DB(i+1)} \quad (33)$$

For $q=1$ to $i-1$

$$C_{T_GS(q)} = C_{T_GS(0)} + \sum_i^q (C_{GS(i+1)} + C_{SB(i+1)} + C_{GD(i+2)} + C_{DB(i+2)}) \quad (34)$$

The equivalent gate capacitance ($C_{I\text{gate}}$) is:

$$C_{I\text{gate}} = C_{(i)} + C_{i1,P} + C_{i3,P} \quad (35)$$

For $i = A, B, \dots (K-1)$

c) Capacitance of the last input (transistor connected to ground - $i = n-1$):

$$C(n-1) = \frac{2}{2^n} C_{GD(n-1)} + \frac{1}{2^n} \sum_{q=1}^{n-1} (2^{n-q} \frac{C_{GD(n-1)} * C_{Tf(q-1)}}{C_{GD(n-1)} + C_{Tf(q-1)}}) + C_{GS(n-1)} \quad (36)$$

$$C_{Tf(0)} = C_{GS(n-2)} + C_{SB(n-2)} + C_{DB(n-1)} \quad (37)$$

For $q=1$ to $n-2$

$$C_{Tf(q)} = C_{Tf(0)} + \sum_{i=1}^q (C_{GD(n-1-i)} + C_{DB(n-1-i)} + C_{GS(n-2-i)} + C_{SB(n-2-i)}) \quad (38)$$

The equivalent gate capacitance ($C_{K\text{gate}}$) is:

$$C_{K\text{gate}} = C_{(K-1)} + C_{K1,P} + C_{K3,P} \quad (39)$$

$n = K$

Where

C_{GD} = Capacitance Gate-Drain

C_{GS} = Capacitance Gate-Source

C_T = Equivalent capacitance of analyzed node

i = node

n = number of inputs of the logic gate

D. CMOS NOR's

The NOR logic *gates* are dual NAND logic *gates*. In this case the internal capacitance will be present in the *pmos* transistor network.

E. Interconnection Capacitances

The interconnection line capacitance is calculated and it is included to the output node of the considered *gate*. The next stage presents an equivalent capacitance, whose value depends on the type of the logic *gates* and connected load.

Then, the total node capacitance is the sum of the capacitance contributions to the node. The total capacitance of the output node of a *gate g* connected to *n* others *gates* is given by:

$$C_{TOTAL} = C_{line} + \sum_{i=1}^n C_{igate} \quad (40)$$

Where:

C_{line} = interconnection capacitance connected to the node.

C_{igate} = equivalent *gate* capacitance.

n = total number of logic *gates* connected to the node.

F. Experimental Setup

The block diagram in Figure 5 gives an overall view of the proposed method for validation of the proposed model. The method consists in extraction of capacitances and transistors from the layout of the circuit. We use the following tools in the validation process:

TROPIC2 [7] – This tool automatically generates the layout of a circuit with a “*linear-matrix multi-row*” layout style. It makes the partitioning, placement and routing of a circuit, generating a symbolic output file. Symbolic file is a textual description of a circuit without taking account of the design rules. Wire, contacts and transistors are indicating with a unitary dimension. The input file uses the SPICE format.

CADENCE – The Cadence system was used for layout compaction and capacitance extraction.

POWER ESTIMATE – This program is embedded into SIS. It calculates the circuit switching activity using various delays model: zero delay, unit delay and general delay.

HSPICE – This well know program does circuit analysis at the level transistor. It is very accurate and secure, however its use is limited to circuits with a small number of inputs.

The used validation flow is the following:

Read circuit in the format *blif* file;

Converter *blif* file to *sim* file. The *sim* file format is a SPICE logic *gate* level file;

the *sim* file is read by TROPIC2, that generates a symbolic layout file;

Circuit compaction is performed using the CADENCE tool. The interconnection capacitances and transistors are extracted. The result of this process step is a SPICE *netlist*;

With the *netlist* and *blif* files, it is possible to calculate the concentrated capacitances at the nodes of each logic *gate*.

The capacitances file and logic *gate* library are read into SIS where we can run the *Power Estimate*, that calculates the average power of the circuit;

The HSPICE is the simulation tool used. The analysis was done in two ways:

Distributed Capacitances: In this case, the interconnection capacitances are distributed over all the circuit;

Concentrated Capacitances: Here, the interconnection capacitances are concentrated in the nodes of the circuit. The nodes correspond to the inputs and output of logic *gates*;

The results were analyzed and compared.

G. Experimental Results

In this section, we present some power estimation results using the methods described in the previous sections. In Table I, we present some results for a set of standard logic *gates*: an inverter, AND *gates*, NAND and NOR with 2 to 4 inputs. The layout symbolic files were generated with TROPIC2 [7]. The interconnection capacitances and transistors were extracted using the CADENCE tools.

In order to confirm the accuracy of the approximations made, we compare the results obtained at logic level with the ones obtained with the HSPICE tool. We evaluated the average power using a sequence of input vectors covering all possible combinations of input transitions. For example, for a circuit with one input there is four possible transitions and we use the sequence: 0, 0, 1, 1, and 0. For a circuit with two inputs, there are sixteen possible transitions: 00, 00, 01, 00, 10, 00, 11, 01, 01, 10, 01, 11, 10, 10, 11, 11, and 00. In general, the total number input transition of a circuit with n inputs is 2^{2n} .

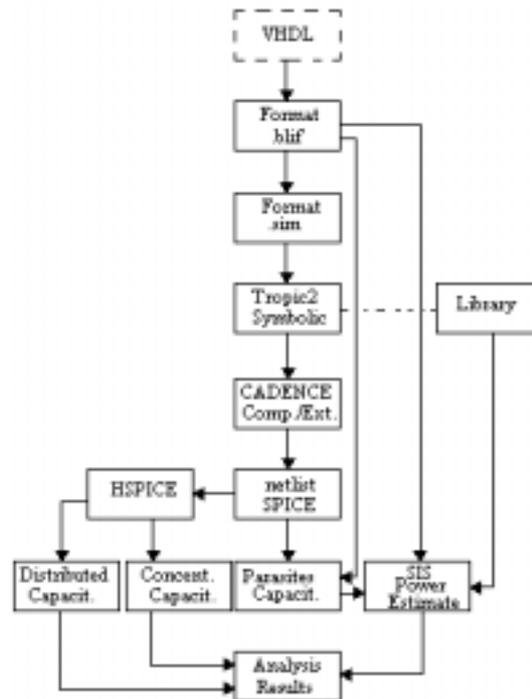


FIGURE 5 - Fluxogram of the Experimental Setup

It was done the comparison between the proposed evaluation method and HSPICE results. The simulations were made with transistors using a 0.8 technology. The average power values are in microWatts, assuming a 5V-supply voltage and 20MHz-clock frequency. As example of extracted capacitance file, we have the layout netlist file for a NAND2 generated with TROPIC2 and CADENCE.

* net 0 = gnd!

* net 1 = /b

* net 2 = /a input

* net 3 = /vdd!

c0 2 1 2.15401e-16

c1 3 1 8.81069e-16

c2 3 2 5.53879e-15

c3 0 1 2.54454e-15

c4 0 2 2.0584e-15

c5 0 3 3.60421e-14

.model model2 pmos level=2 vto=-0.7 gamma=0.4 kp=1.5e-05 lambda=0.03 tox=6e-07

m6 1 2 3 3 model2 w=8.01u l=0.745342u ad=1.8e-11 as=1.76e-11 pd=1.5173e-05 ps=1.504e-05

.model model3 nmos level=2 vto=0.7 gamma=0.2 kp=3e-05 lambda=0.02 tox=6e-07
 m7 1 2 0 0 model3 w=8.01u l=0.745342u ad=2.08e-11 as=1.72e-11 pd=1.48e-05 ps=1.392e-05

The estimated power is presented below:

Node a Cap.=9.749183e-015 sw=0.500000 Power=1.218648e-006

Node b Cap.=8.842158e-015 sw=0.500000 Power=1.105270e-006

Node c{[5069]} Cap.=5.149386e-015 sw=0.3800 Power=4.891917e007

Total Power: 2.813109e-006

The general results of these comparisons are presented in table I and III.

In the second and third column of table I, we show the estimations obtained with our approach and with the logic-level estimator of the Power Estimate tool, respectively, without taking into account the interconnection capacitance. Columns 5 and 6 present the estimations made considering the interconnection capacitance. The 4 and 7 columns of Table I show the percentage error between the proposed method and HSPICE. In the Table II we present the characteristics of five circuits MCNC '91 benchmark: C17, cm138a, cm42a, decod and majority. In the Table III, we show the average power estimation and error. The circuits analyzed are limited to two conditions: number of inputs and number of logic levels. The first condition has to do with the number of input combinations that we have to simulate in HSPICE. The second is related to the fact that we are still not computing the delays of the logic gates with our method and therefore we can not compute an accurate glitching power.

The small errors of the evaluations presented in the last column encourage us to continue the investigation for larger circuits.

Name	Without Interc.			With Interc.		
	HSP	Prop. Meth.	Error (%)	HSP	Prop. Meth.	Error (%)
Inv	0,139	0,153	0,0	0,680	0,585	5,6
And2	0,254	0,282	1,0	2,880	2,813	2,3
And3	0,368	0,367	0,2	4,758	4,608	3,1
And4	0,419	0,410	2,1	6,160	6,096	1,0
or2	0,275	0,298	8,3	3,220	3,205	0,4
or3	0,378	0,365	-3,4	4,388	4,223	-3,7
or4	0,441	0,418	5,2	5,874	5,686	3,2

Table I – Power consumption of conventional logic gates.

Name	# Gates	#Trans	#nodes	# interc. Capac.
C17	6	24	19	60
Cm138a	13	96	56	231
Cm42a	16	100	56	241
Decod	22	146	80	288
Majority	7	46	30	124

Table II – Characteristics of the Circuits Benchmarks Analyzed

Average Power (μW)				
Name	Level Logic	HSPICE Appr.(1)	Prop. Meth. (2)	Error (%)
C17	3	32.54	30.21	-7,16
Cm138a	3	99.40	96,94	-2,47
Cm42a	3	120.57	114.72	-4,85
Decod	3	234.24	221,77	-5,32
Majorit	4	64.91	58.49	-9,89

Table III – Evaluations of Power Consumption for the MCNC'91 Benchmarks Circuits

Conclusion and Future Work

We presented an accurate method for capacitance modeling by computing equivalent concentrated capacitances at the external nodes of logic. The model is able to take into account the capacitance of internal nodes and the interconnect capacitance extracted from the layout. The results of experimental applications to standard logic *gates* validated the proposed procedure. It was done a comparison between circuits with a number of logic levels less than three using MCNC'91 benchmark circuits. For logic with more than three levels, delay must be considered in a power consumption evaluation. The next step of this research will consist in taking account of the delay of each logic *gate* and switching activity with a variable probability.

References

1. R. Burch, F. Najm, P. Yang and T. N. Trick. (1993) A Monte Carlo Approach for Power Estimation. *IEEE Transactions on Very Large Scale Integration*, vol. I, no. 1, march, pp. 63 – 71.
2. A. Ghosh, S. Devadas, K. Keutzer and J. White (1992) Estimation of Average Switching Activity in Combinational and Sequential Circuits, in *Proceedings: 29th ACM/IEEE Design Automation Conference*, June, pp. 253-259.
3. J. Monteiro e S. Devadas. (1997) *Computer-Aided Design Techniques for Low Power Sequential Logic Circuits*, Ed. Kluwer Academic Publishers, USA.
4. S. Iman and M. Pedram (1995) Logic extraction and Factorization for Low Power. In: *32nd Design Automation Conference. Proceedings: IEEE Computer Society Press*, June, p. 248-253.
5. L. Glasser and D. Dobberpuhl (1985) *The Design and Analysis of VLSI Circuits*. Reading, MA: Addison-Wesley.
6. J. P. Uyemura (1992) *Circuit Design for CMOS VLSI*, Ed. Kluwer Academic Publishers, USA.
7. F. Moraes, N. Azemard, M. Robert and D. Auvergne (1993) Flexible Macrocell Layout Generator, *4th ACM/SIGDA Physical Design Workshop*, Los Angeles (USA), pp. 105-116.
8. F. Najm, Transition Density: A New Measure of Activity in Digital Circuits, (1993) *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 12, No. 2, Feb., pp. 310-323.
9. F. Najm (1994) A Survey of Power Estimation Techniques in VLSI Circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 2, Dec., pp. 446-455.

Anexo 2

Artigo publicado XV International Conference on Microelectronics and Packaging, SBMicro 2000, 18-24 September, Manaus, Amazonas, Brasil, 2000, p. 226- 231.

Power Estimation at Logic-Level Considering Interconnection Capacitances

Martins, J.B., Klein, F., Reis, R. and Moraes, F.

{batista,fklein,reis}@inf.ufrgs.br

moraes@inf.pucrs.br

II/UFRGS, Porto Alegre, Brazil

II, PUC, Porto Alegre, Brazil

Abstract

Accurate and fast power consumption estimation of CMOS circuits during the design phase is required to guide power optimization techniques employed to meet stringent power specifications. Logic-level power estimation tools, such as those available in the SIS frameworks (POSE and Power Estimate) are able to accurately calculate the switching activity under a given delay model. However, capacitance modeling is crude. The goal of the work described in this paper is to estimate the average power of a circuit with interconnection capacitances and transistors extracted from the layout. To validate the method we use the Diva extractor (Cadence Design Systems) and LASCA [8] extractor. We propose new models for the input and output capacitances of logic gates, taking into account the internal capacitances of the gates. The results we present show a comparison of logic-level power estimation with capacitances extracted with DIVA and LASCA, with an error of less than 10%.

Keywords – Power estimation, Logic Level, capacitances.

1. Introduction

Power consumption is becoming one of the most important VLSI design issues. Optimization techniques for low power are being employed at all design levels of abstraction. In order to guide designers and optimization tools, there is a pressing need for accurate and fast power consumption estimation tools.

During normal operation, the power dissipation of a CMOS circuit is directly related to the switching activity. For a well designed circuit, the total average power can be approximated by the switched-capacitance power [1]. This is an underlying assumption of almost all available power estimation tools at the logic and higher levels of abstraction. The average dynamic power consumption of a CMOS circuit is then given by:

$$Power = \frac{1}{2} \cdot f \cdot V_{dd}^2 \cdot \sum_{i=1}^n C_i \cdot \alpha_i \quad (1)$$

Where f is the clock frequency, V_{dd} is the supply voltage, C_i and α_i are capacitance load and average switching of the logic *gate* i , respectively.

Significant amount of work has been carried out in developing efficient techniques to estimate the switching activity of a CMOS circuit [2]. These techniques can be divided into two classes: statistical techniques (also known as dynamic techniques), e.g. [3]; probabilistic (or static) techniques, e.g. [4].

Probabilistic techniques propagate input statistics through the circuit to obtain the switching probability for each *gate* in the circuit. Probabilistic techniques are employed in the power estimation tools of SIS [5] and POSE [6].

In this paper, we focus on the problem of capacitance modeling. This is typically done very simplistically at the logic level. We propose to build a more accurate model. At the logic-level, only the input and output nodes of the *gates* are available. We present an equivalent load capacitance model for each external node, computed from the internal transistor capacitances and the interconnection capacitance.

The interconnection capacitance is currently being extracted from layout information. We have performed several experiments on (necessarily) simple circuits where power estimation was made using interconnection capacitances extracted of the LASCA and DIVA system. The results are very promising, with power estimation differences of less than 10%.

This paper is organized as follows. In Section 2, we briefly present the capacitance *gate* model used for a transistor MOSFET. In Section 3, we describe the methodology to extract the coupling and ground capacitances of interconnections. In the section 4 is presented the modeling of *gate* capacitance at logic level. In Section 5, we describe the experimental setup for to validate the propose method. Experimental results are presented in Section 6. In Section 7, we give some conclusions and discuss future research.

2. Capacitance Gate Model

The MOSFET transistors exhibit a number of parasitic capacitance [7] (Figure 1), which must be accounted in circuit design: *gate-to-source* capacitance (C_{GS}), *gate-to-drain* capacitance (C_{GD}), *gate-to-bulk* capacitance (C_{GB}), *source-to-bulk* capacitance (C_{SB}) and *drain to bulk* capacitance (C_{DB}). In this work, our interest is the switching region of the transistors and all the capacitances will be considered, with the exception of C_{GB} .

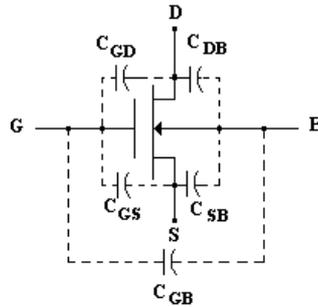


Figure 1- Parasitic Capacitances of the MOSFET

Although these capacitances are a nonlinear function of the voltage, the general approach is to assume them as a linear, time-invariant element. Given some technology parameters and the size the transistor (W/L), a value for each of these parasitic capacitances can be computed [7].

3. Interconnection Capacitances

The interconnection capacitances are extracted directly from a flat layout of a circuit. We developed a fast capacitance extraction tool (LASCA wire extractor [8]) using a Bin-Based algorithm [9] to extract the connectivities, empirical formulation [10] to compute the capacitances, and a simple 2½D methodology described in [11]. The input of the extractor is a flat layout description and a technology file. The layout description is scanned and during this process a net list describing all parasitic capacitances is generated.

The 2½D methodology described in [11] was based on experiments with a 3D field solver over 0.50µm, 0.35µm and 0.18µm process. Briefly, for each connection in the layer i , we should analyze the immediate neighbor in the same layer, all crossunders in the layer $i-1$ and all crossovers in the layer $i+1$, treating the layers $i±2$ like ground planes.

In Figure 2 is presented the model to calculate the interconnect capacitance in each node of the circuit. It consists of two conduction layers over the substrate, considered as a reference plane (ground plane). There are three capacitance components at any node [12]:

Overlap capacitance (C_{over}) - due the overlap between two conductors in different planes. They are C_{21a} and C_{23a} in the Figure 2.

Lateral capacitance (C_{lat}) - is the capacitance between two conductors in the same plane. In the Figure 2 is C_{22lat} .

Fringing capacitance (C_{fr}) - due the coupling between two conductors of different planes. They are C_{21fr} and C_{23fr} in the Figure 2.

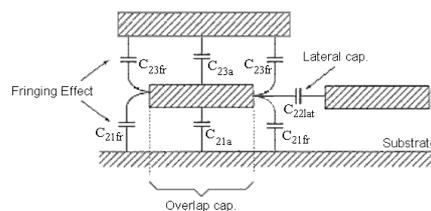


Figure 2 – Capacitance Model

The capacitances C_{ij} ($i, j=1 \dots n$, where n is the number of conductors) are calculated by an empirical formulation.

3.1 Intrinsic Capacitance

The intrinsic capacitance is the capacitance between one conductor layer and the ground plane. It has two components: overlap and fringing capacitances.

Two parallel plates model the overlap capacitance. The fringing capacitance is due to the edge of one conductor and the surface of the other one (in this case, the ground plane). They are calculated using the traditional formulation based in the overlap area and perimeter. Thus, the intrinsic capacitance is the sum of these two components:

Where C_{area} is capacitance per unit area (fF/ μm^2), C_{length} is the capacitance per unit length (fF/ μm), W is the wire width and L is the wire length.

3.2 Coupling Capacitance

The coupling capacitances are extracted, because, in deep-submicron processes, the increase in the ratio between wire thickness and space between wires has substantially increased the lateral capacitance.

The lateral coupling capacitance is a function, basically, of two factors: wire thickness and the space between wires. The wire thickness is a function of the technology and the layer considered. However, the space between wires is the main factor, depending of structure analyzed. As the space between two wires can have a large variation, we can not use the capacitance informed by technology rules, because it is restricted to the minimum distance of separation.

In deep-submicron process, the foundries set the minimum area usage for each metal layer. This value is normally fixed to 30%, for uniformity of etches rate or CMP planarization. The maximum area usage is around 50%, when the distance between two wires is equal to the minimum distance of the process. Thus, the distance between two wires has a large variation. If we use the capacitance from the technology rules, we have a large error.

To calculate the coupling capacitance it was used the empirical formulation described in [10], and two cases can happen: just one ground plane or two ground planes, where the second plane is the $i+2$ plane. These formulations need some parameters from layout, like the space between two wires and some others from the technology like thickness of metal line, width of metal line and thickness of dielectric layer between metal layer and ground plane.

3.3 Crossover Capacitance

The crossover capacitance is only calculated if there is a common area between two wires in two different planes. This common area is extracted by the AND logic operator, between the two wires. As the intrinsic capacitance, the crossover capacitance (C_{cross}) is modeled considering two parallel plates. It is calculated using the following equation:

$$C_{\text{cross}} = C_{\text{area}} \cdot W_1 \cdot W_2 + 2 \cdot C_{\text{length}} \cdot (W_1 + W_2) \quad (3)$$

4. Modeling Gate Capacitances in the Logic Level

For circuits described at the logic level there is only access to the logic *gates* input and output nodes. Therefore, all internal *gate* capacitances have to be taken into account in a concentrated capacitance model of external nodes. In this section we describe how we compute the equivalent capacitance load for any logic *gate* from its transistor level description.

A. Inverter

The CMOS inverter (Figure 3) presents two external nodes. In this case, equivalent capacitances at the input x , $C_{\text{in } x}$, and output y , $C_{\text{out } y}$, nodes can be simply computed using:

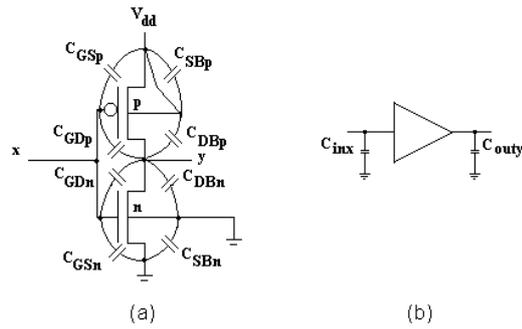


Figure 3 – Parasitic capacitances of CMOS inverter

- a) transistor-level representation
- b) logic-level representation

$$C_{inx} = C_{GSp} + C_{GDp} + C_{GSn} + C_{GDn} \tag{4}$$

$$C_{outy} = C_{GCDp} + C_{GDn} + C_{DBp} + C_{DBn} \tag{5}$$

B.NAND2

In the case of a CMOS 2-input NAND *gate* (Figure 4), there are two input nodes with a concentrate capacitance and one output node. The additional difficulty is that now it is necessary to model the internal capacitance. We analyze the equivalent capacitance by a superposition of signals. For example, when we calculate the equivalent capacitance of an input node, the inputs and output will be in the ground level.

The capacitance of internal nodes depends on the logic value of the others inputs. To model this effect we are assuming a probability of 0.5 that all inputs are set to 1. That is, the *NMOS* transistors will be ON at half the time.

In the case NAND2, the input combination possible is 00, 01, 10 and 11.

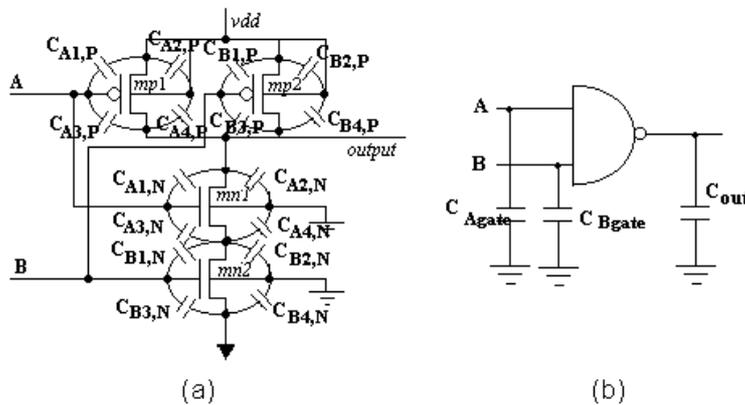


Figure 4–Parasitic Capacitances of 2-input NAND

- (a) Transistor-level representation
- (b) Logic-level representation

Input “a”:

For input combinations 00 and 10, the “mn2” transistor is OFF and equivalent capacitance node (Ca1) is equal to:

$$C_{a1} = 0.5(C_{A1,N} + \frac{C_{A3,N} * C_{T,N}}{C_{A3,N} + C_{T,N}}) \quad (6)$$

$$C_{T,N} = C_{A4,N} + C_{B1,N} + C_{B2,N} \quad (7)$$

For input combinations 01 and 11, the mn2 transistor is ON and the equivalent capacitance (Ca2) is equal to:

$$C_{a2} = 0.5(C_{A1,N} + C_{A3,N}) \quad (8)$$

Then

$$C_{A,N} = C_{a1} + C_{a2}$$

$$C_{A,N} = 0.5(C_{A1,N} + \frac{C_{A3,N} * C_{T,N}}{C_{A3,N} + C_{T,N}}) + 0.5(C_{A1,N} + C_{A3,N}) \quad (9)$$

In terms of probability:

$$C_{A,N} = p_0(C_{A1,N} + \frac{C_{A3,N} * C_{T,N}}{C_{A3,N} + C_{T,N}}) + p_1(C_{A1,N} + C_{A3,N}) \quad (10)$$

And C_{Agate} is:

$$C_{Agate} = p_0(C_{A1,N} + \frac{C_{A3,N} * C_{T,N}}{C_{A3,N} + C_{T,N}}) + p_1(C_{A1,N} + C_{A3,N}) + C_{A1,P} + C_{A3,P} \quad (11)$$

Where p_0 is the probability that the second input is equal the ZERO and p_1 is the probability that the second input is equal the ONE ($p_0 + p_1 = 1$).

Input “b”

In the same way, we can analyze the second input.

For input combinations 00 and 01, the mn1 transistor is OFF and C_{b1} is equal:

$$C_{b1,N} = 0.5(C_{B3,N} + \frac{C_{B1,N} * C_{T1,N}}{C_{B1,N} + C_{T1,N}}) \quad (12)$$

For input combinations 10 and 11, the mn1 transistor is ON and C_{b2} is equal:

$$C_{b2,N} = 0.5(C_{B1,N} + C_{B3,N}) \quad (13)$$

Then

$$C_{B,N} = C_{b1,N} + C_{b2,N} \quad (14)$$

$$C_{B,N} = 0.5(C_{B3,N} + \frac{C_{B1,N} * C_{T1,N}}{C_{B1,N} + C_{T1,N}}) + 0.5(C_{B1,N} + C_{B3,N}) \quad (15)$$

$$C_{T1,N} = C_{A4,N} + C_{A3,N} + C_{B2,N} \quad (16)$$

In terms of probability :

$$C_{B,N} = p_0(C_{B3,N} + \frac{C_{B1,N} * C_{T1,N}}{C_{B3,N} + C_{T1,N}}) + p_1(C_{B1,N} + C_{B3,N}) \quad (17)$$

and C_{Bgate} is:

$$C_{Bgate} = p_0(C_{B3,N} + \frac{C_{B1,N} * C_{T1,N}}{C_{B1,N} + C_{T1,N}}) + p_1(C_{B1,N} + C_{B3,N}) + C_{B1,P} + C_{B3,P} \quad (18)$$

Where C_{Agate} and C_{Bgate} are equivalent capacitances for input A and input B, respectively.

And output is:

$$C_{out} = C_{A1,N} + C_{A2,N} + C_{A3,P} + C_{A4,P} + C_{B3,P} + C_{B4,P} \quad (19)$$

CMOS NOR's

The NOR logic *gates* are dual NAND logic *gate*. In this case the internal capacitance will be present in the *pmos* transistor network.

5. Experimental Setup

The block diagram in Figure 5 gives an overview of the proposed method for validation of the proposed model. The method consists in extraction of capacitances and transistors from the layout of the circuit. We use the following tools in the validation process:

TROPIC3 [13] – This tool automatically generates the layout of a circuit with a “linear-matrix multi-row” layout style. It makes the partitioning, placement and routing of a circuit, generating the layout output file. The input file uses the SPICE format.

LASCA [8] – This tool is responsible by the capacitance extraction of the circuit from the layout description. The layout is generated by TROPIC3 tool.

Diva – The Cadence system was used for capacitance extraction.

Power Estimate – This program is embedded into SIS. It calculates the switching activity of the circuit using various delays model: zero delay, unit delay and general delay.

The validation flow used is the following:

Read circuit in the format *.cif* file (layout);

The *.cif* file is read by LASCA, that extracts the capacitances and transistors;

The *gate* capacitances are estimated using the extracted transistors file;

The interconnection capacitances and the *gate* capacitance are concentrated in the nodes (logic level);

The average power is estimated by Power Estimate tool using as input the extracted capacitance file, as well the switching activity and logic description of the circuit;

In proposed method of power estimation, the capacitances are concentrated in the external nodes of the logic *gates*;

The validation of the values of the interconnection capacitances used in the proposed method of power estimate is made by the comparison between results using DIVA and LASCA extractors.

The results were analyzed and compared.

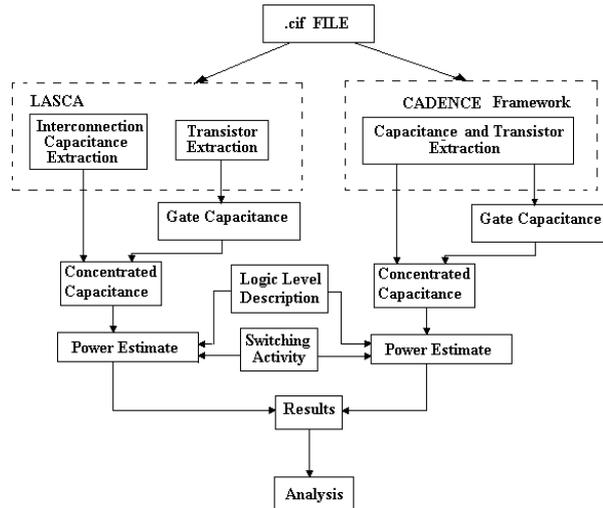


Figure 5 - Fluxogram of the Experimental Setup

6. Experimental Results

In this section, we present some power estimation results using the methods described in the previous sections. In Table 1 and 2, we present some benchmarks circuits: C17m, cm42a, decod and majority and power estimation results, respectively. The layout files were generated with TROPIC [13]. The interconnection capacitances and transistors were extracted using the LASCA tool [8].

In order to confirm the accuracy of the approximations made, we compare the results obtained at logic level with the ones obtained with capacitances extracted with DIVA. We evaluated the average power using Power Estimate. It was done the comparison between interconnection capacitances using LASCA and DIVA. The simulations were made with transistors using a 0,25 μ m technology. The average power values are in microWatts, assuming a 2.5V-supply voltage and 20MHz-clock frequency. The general results of these comparisons are presented in Table 1 and 2.

The small errors of the evaluations presented in the last column encourage us to continue the investigation for larger circuits.

Name	# <i>gates</i>	# Transistor	#nodes
C17m	6	24	19
Cm138a	13	96	56
Cm42a	16	100	56
Decod	22	146	80
Majority	7	46	30

Table 1 – Characteristics of the Circuits from MCNC'91 Benchmark

Name	Average Power(μ W)		
	LASCA	DIVA	%
C17m	9.60	8.69	9,47
Cm42a	8.49	7.88	7,1
Decod	16.22	15.56	4,06
Majority	8.10	8.66	5,43
Cm138a	10.47	9.68	9,47

Table 2 – Evaluations of Power Consumption for the MCNC'91 Benchmark Circuits

7. Conclusion and Future Work

We presented a method for power consumption estimate using a capacitance modeling by computing equivalent concentrated capacitances at the external nodes of a logic *gate*. The model is able to take into account the capacitance of internal nodes and the interconnect capacitance extracted from layout. The interconnection capacitances are extracted with LASCA [8] extractor and DIVA extractor. The results of

experimental application to standard logic *gates* validated the proposed procedure. It was done a comparison between circuits from MCNC'91 benchmark, with less than three logic levels. For logic with more than three levels, delay must be considered in a power consumption evaluation. The next step of this research will consist in taking account of each logic *gate* delay and a variable probability of switching activity.

References

- [1] L. Glasser and D. Dobberpuhl, *The Design and Analysis of VLSI Circuits*. Reading, MA: Addison-Wesley, 1985.
- [2] F. Najm, A Survey of Power Estimation Techniques in VLSI Circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 2, Dec 1994, pp. 446-455.
- [3] R. Burch, F. Najm, P. Yang and T. N. Trick. A Monte Carlo Approach for Power Estimation. *IEEE Transactions on Very Large Scale Integration*, vol. I, no. 1, march 1993, pp. 63 – 71.
- [4] A. Ghosh, S. Devadas, K. Keutzer and J. White. Estimation of Average Switching Activity in Combinational and Sequential Circuits, in *Proceedings: 29th ACM/IEEE Design Automation Conference*, June 1992, pp. 253-259.
- [5] J. Monteiro e S. Devadas. *Computer-Aided Design Techniques for Low Power Sequential Logic Circuits*, Ed. Kluwer Academic Publishers, USA, 1997.
- [6] S. Iman and M. Pedram. Logic extraction and Factorization for Low Power. In: *32nd Design Automation Conference. Proceedings: IEEE Computer Society Press*, June 1995, p. 248-253.
- [7] J. P. Uyemura *Circuit Design for CMOS VLSI*, Ed. Kluwer Academic Publishers, USA, 1992.
- [8] F. Ferreira, F. Moraes and R. Reis. *Extração de Elementos Parasitas em Circuitos CMOS Submicrônicos*. Porto Alegre: PPGC da UFRGS, 2000. Dissertação de Mestrado.
- [9] N. Sherwani. *Algorithms for VLSI Physical Design Automation*. Kluwer Academic Publisher, 1993.
- [10] J.H. Chern, J. Huang, L. Arledge, P. C. Li and P. Yang. Multilevel Metal Capacitances Models for CAD Design Synthesis Systems. *IEEE Electron Devices Letters*, 13(1):32-34, Feb. 1992.
- [11] J. Cong, A.B. Kahng, D. Noice, N. Shiralli and S.H. Yen. Analysis and Justification of a Simple, Practical 2 1/2D Capacitance Extraction Methodology. *UCLA Computer Science Technical Report 970013*, 1996.
- [12] N.D. Arora, K.V. Raol, R. Schumann and L.M. Richardson. Modeling and Extraction of Interconnect Capacitances for Multilayer VLSI Circuits. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 15(1):58-66, Jan. 1996.
- [13] F. Moraes, M. Robert and D. Auvergne. A Virtual CMOS Library Approach for Fast Layout Synthesis. In: *VLSI*, 1999.

Referências

- [ARO 96] ARORA, N.D. et al. Modeling and Extraction of Interconnect Capacitances for Multilayer VLSI Circuits. **IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems**, New York, v.15,n.1, p.58-66, Jan. 1996.
- [BAK 90] BAKOGLU, H. B. **Circuits, Interconnections and Packaging for VLSI**. Menlo Park, Addison-Wesley, 1990.
- [BEA 95] BEARDEN, D. et al. A 133 MHz 64b Four-Issue CMOS Microprocessor. In: INTERNATIONAL SOLID-STATE CIRCUITS CONFERENCE, 1995. **Proceedings...**[S. l.: s.n], 1995. p. 174-175.
- [BEC 94] BECHADE, R. et al. A 32b 66 MHz Microprocessor. In: INTERNATIONAL SOLID-STATE CIRCUITS CONFERENCE, 1994. **Proceedings...** [S. l.: s.n], 1994. p. 208-209.
- [BEL 95] BELLAOUAR, A.; ELMASRY, M. I. **Low-Power Digital VLSI Design Circuits and Systems**. Boston: Kluwer Academic, 1995.
- [BEN 94] BENINI, L. et al. Analysis of hazard contribution to power dissipation for CMOS IC's. In: ACM/IEEE INTERNATIONAL WORKSHOP ON LOW POWER DESIGN, 1994. **Proceedings ...** [S. l.: s.n], 1994. p.27-32.
- [BOW 95] BOWHILL, W. J. et al. A 300 MHz 64b Quad-Issue CMOS RISC Microprocessor. In: IEEE INTERNATIONAL SOLID-STATE CIRCUITS CONFERENCE, 1995. **Proceedings ...** [S. l.: s.n], 1995. p. 182-183.
- [BUR 93] BURCH, R. et al. A Monte Carlo Approach to Power Estimation. **IEEE Transactions on Very Large Scale Integration Systems**, New York, v. 1, n. 1, p. 63-71, Mar. 1993.
- [CHA 95] CHANDRAKASAN, A. P et al. Optimizing Power Using Transformations. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, v. 14, n. 1, p. 12 – 31, Jan. 1995.
- [CHA 96] CHANG, J.M. ; PEDRAM, M. Module Assignment for Low Power. In: IEEE EUROPEAN DESIGN AUTOMATION CONFERENCE, 1996. **Proceedings ...** Geneva (Switzerland): [s.n.], 1996. p. 376-381.
- [CHE 88] CHEN, X. ; BUSHNELL, M. L. A Module Area Estimator for VLSI Layout In: ACM/IEEE DESIGN AUTOMATION CONFERENCE, 25. , 1988. **Proceedings...** Anaheim (USA): IEEE Computer Society Press, 1988. p. 54-59.
- [CHE 90] CHENG, K. T. ; AGRAWAL, V. An Entropy Measure for the Complexity of Multi-output Boolean Functions. In: ACM/IEEE DESIGN

- AUTOMATION CONFERENCE, 27., 1990. **Proceedings ...** Orlando(USA): IEEE Computer Society Press, 1990. p. 302-305.
- [CHE 92] CHERN J. H. et al. Yang. Multilevel Metal Capacitances Models for CAD Design Synthesis Systems. **IEEE Electron Devices Letters**, [S.l.], v.13, n.1, p.32-34, Feb. 1992.
- [CHR2000] CHRISTIE, P. ; STROOBANDT, D. The Interpretation and Application of Rent's Rule. **IEEE Transactions on Very Large Scale Integration Systems**, Special Issue on System-Level Interconnect Prediction, New York , Sept. 2000.
- [CIR 87] CIRIT, M. Estimating Dynamic Power Consumption of CMOS Circuits. In: **IEEE INTERNATIONAL CONFERENCE COMPUTER AIDED DESIGN, 1987. Proceedings...** Santa Clara (USA): IEEE Computer Society Press, 1987. p.534-537.
- [COV 91] COVER, T.M. ; THOMAS, J. A. **Elements of Information Theory**. New York: Wiley, 1991.
- [DEV 94] DEVADAS, S.; GHOSH, A.; KEUTZER, K. **Logic Synthesis**. USA: McGraw-Hill, 1994.
- [DIN 98] DING, C. S.; TSUI, C.Y. ; PEDRAM, M. *Gate-Level Power Estimation Using Tagged Probabilistic Simulation*. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, v. 17, n. 11, p. 1099 – 1107, Nov. 1998.
- [DOB 92] DOBBERPUHL, D. W. et al. A 200 MHz Dual-Issue CMOS Microprocessor. **IEEE Journal Solid-State Circuits**, New York, v. 27, n. 11, p. 1555-1567, Nov. 1992.
- [DON 79] DONATH, W. E. Placement and Average Interconnection Lengths of Computer Logic, **IEEE Transactions Circuits Systems**, New York, v. CAS-26, p. 272-277, Apr. 1979.
- [FER2000] FERREIRA, F. et al. **Extração de Capacitâncias e Resistências Parasitas em Circuitos CMOS Submicrônicos**. 2000. 90f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- [FEV 82] FEVER, M. Connectivity of Random Logic. **IEEE Transactions Computer**, New York, v. C-31, p. 29-33, Jan. 1982.
- [GAM 81] GAMAL, A. A. El. Two-dimensional Stochastic Model for Interconnections in Master Slice Integrated Circuits. **IEEE Transaction Circuits Systems**, New York, v. CAS-28, p. 127-138, Feb. 1981.
- [GER 94] GEROSA, G. et al. A 2.2 W 80 MHz Superscalar RISC Microprocessor. **IEEE Journal of Solid-State Circuits**, New York, v. 29, n. 12, p. 1440-1454, Dec. 1994.

- [GHO 92] GHOSH, A. et al. Estimation of Average Switching Activity in Combinational and Sequential Circuits. In: ACM/IEEE DESIGN AUTOMATION CONFERENCE, 29., 1992. **Proceedings ...** Anaheim (USA): IEEE Computer Society Press, 1992. p. 253-259.
- [HEB 96] HEBGEN, W. ; ZIMMERMANN, G. Hierarchical Netlength Estimation for Timing Prediction. In: ACM/SIGDA PHYSICAL DESIGN WORKSHOP, 1996. **Proceedings...** [S.l. : s.n.], 1996. p. 118-125.
- [HEL 77] HELLER, W. R. et al. Prediction of Wiring Space Requirements for LSI. In: ACM/IEEE DESIGN AUTOMATION CONFERENCE, 14., 1997. **Proceedings...** Anaheim(USA): IEEE Computer Society Press, 1997. p. 32-42.
- [HIS 98] HSIEH, C. T. ; PEDRAM, M. Microprocessor Power Estimation Using Profile-Driven Program Synthesis. **IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems**, New York, v. 17, n. 11, p. 1080-1089, Nov. 1998.
- [KUR 89] KURDAHI, F. J. ; PARKER, A. C. Techniques for Area Estimation of VLSI Layouts. **IEEE Transactions Computer Aided Design**, New York, v. 8, p. 81-92, Jan.1989.
- [LAN 71] LANDMAN, B. S. ; RUSSO, R. L. On a Pin Versus Block Relationship for Partitions of Logic Graphs. **IEEE Transactions Computer**, New York, v. C-20, p. 1469-1479, Dec. 1971.
- [LAN 95] LANDMAN, P.; RABAEY, J. Activity-sensitive Architectural Power Analysis for the Control Path. In: In: ACM/IEEE INTERNATIONAL SYMPOSIUM LOW POWER DESIGN, 1995. **Proceedings ...** [S. l.: s.n], 1995. p. 93-98.
- [LIN 94] LIN, J. Y. ; LIU, T. C. ; SHEN, W. Z. A Cell-Based Power Estimation in CMOS Combinational Circuits. In: ACM/IEEE DESIGN AUTOMATION CONFERENCE, 31. , 1994. **Proceedings ...** Los Alamitos (USA): IEEE Computer Society Press, 1994. p. 304-309.
- [MAC 98] MACII, E. ; PEDRAM, M ; SOMENZI, F. High-Level Power Modeling Estimation and Optimization. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, v. 17, n. 11, p. 1061-1079, Nov. 1998.
- [MAR 96] MARCULESCU, D.; MASCOLESCU, R.; PEDRAM, M. Information Theoretic Measures for Power Analysis. **IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems**, New York, v. 15, n. 6, p. 599-609, June 1996.
- [MAR2000a] MARTINS, J.B. et al. Power Estimation at Logic-Level Considering Interconnection Capacitance. In: INTERNATIONAL CONFERENCE ON MICROELECTRONICS AND PACKAGING, SBMICRO, 15., 2000, Manaus. **Chip in the Jungle: proceedings.** [Manaus: SBMICRO], 2000. p. 226-231.

- [MAR2000b] MARTINS, J.B. et al. Capacitance and Power Modeling at Logic Level. In: INTERNATIONAL CONFERENCE ON CHIP DESIGN AUTOMATION. **Proceedings ...** Beijing (China): PHEI, 2000.
- [MIC 94] MICHELI, G. D. **Synthesis and Optimization of Digital Circuits**. USA: McGraw-Hill, 1994.
- [MON 97] MONTEIRO, J. ; DEVADAS, S. **Computer-Aided Design Techniques for Low Power Sequential Logic Circuits**. Boston: Kluwer Academic, 1997.
- [MOR 93] MORAES, F. et al. Flexible Macro Cell Layout Generator. In: ACM/SIGDA PHYSICAL DESIGN WORKSHOP, 4., 1993. **Proceedings ...** Los Angeles: [s.n.], 1993. p. 105-116.
- [MOR 99] MORAES, F. et al. A Virtual CMOS Library Approach for Fast Layout Synthesis. In: IFIP TC10 WG10.5 INTERNATIONAL CONFERENCE ON VERY LARGE SCALE INTEGRATION, VLSI, 10. , 1999, Lisboa. **Proceedings ...** Boston: Kluwer Academic, 2000. p. 415-426.
- [NAG 75] NAGEL, W. **SPICE2, A Computer Program to Simulate Semiconductor Circuits**. Berkeley, California: University of California, Department of Electrical Engineering and Computer Sciences, 1975. 63p. (UCB/ERL M75/520).
- [NAJ 93] NAJM, F. Transition Density: A New Measure of Activity in Digital Circuits. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, v. 12, n. 2, p. 310-323, Feb. 1993.
- [NAJ 94] NAJM, F. A Survey of Power Estimation Techniques in VLSI Circuits. **IEEE Transactions on Very Large Scale Integration Systems**, New York, v. 2, n. 4, p. 446-455, Dec. 1994.
- [NEM 96] NEMANI, M. ; NAJM, F. Towards a High-Level Power Estimation Capability. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, v. 15, n. 6, p. 588-598, June 1996.
- [NEM 99] NEMANI, M and NAJM, F. High-Level Area and Power Estimation for VLSI Circuits. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, v. 18, n. 6 , p. 697-713, June 1996.
- [PED 89] PEDRAM, M. ; PREAS, B. Interconnection Length Estimation for Optimized Standard Cell Layouts. In: IEEE INTERNATIONAL CONFERENCE COMPUTER-AIDED DESIGN, 1989. **Proceedings ...** Santa Clara (USA): IEEE Computer Society Press, 1989. p. 390-393.
- [PED 99] PEDRAM, M. ; PREAS, B. T. Interconnection Analysis for Standard Cell Layouts. **IEEE Transactions on Computer Aided Design of**

- Integrated Circuits and Systems**, New York, v. 18, n. 10, p. 1512-1519, Oct. 1999.
- [PIP 77] PIPPENGER, N. **Information Theory and Complexity of Boolean Functions**: Mathematical Systems Theory. New York: Springer-Verlag, 1977.
- [RAB 96] RABAEY, J. M. **Digital Integrated Circuits: A design Perspective**. New Jersey: Prentice-Hall, 1996.
- [REI2000] REIS, R. A. L. (Organizador). **Sistemas Digitales: Síntese Física de Circuitos Integrados**. Bogotá: CYTED, 2000. p. 65-66.
- [ROY 98] ROY, S. **Low-Power-Driven Synthesis Algorithms for Sequential and Combinational Circuits**. 1998. Thesis. University of Illinois at Urbana-Champaign.
- [SAS 86] SASTRY, S. ; PARKER, A. C. Stochastic Models for Wirability Analysis of *Gate* Arrays. **IEEE Transactions Computer Aided Design**, New York, v. CAD-5, p. 52-65, Jan. 1986.
- [SEC 87] SECHEM, C. Average Interconnection Length Estimation for Random and Optimized Placements. In: IEEE INTERNATIONAL CONFERENCE COMPUTER AIDED DESIGN, 1987. **Proceedings ...** Santa Clara (USA): IEEE, Computer Society Press, 1987. p. 190-193.
- [SEN 92] SENTOVICH, E. M. et al. **SIS: A System for Sequential Circuit Synthesis**. Berkeley: Electronics Research Lab. University of California, 1992. (Tech. Rep. UCB/ERL M92/41).
- [SHE 93] SHERWANI, N. **Algorithms for VLSI Physical Design Automation**. USA: Kluwer Academic, 1993.
- [SPE 93] THE NEW CONTENDERS. **IEEE Spectrum**, New York, p. 20-25, Dec. 1993.
- [UYE 92] UYEMURA, J. P. **Circuit Design for CMOS VLSI**. USA : Kluwer Academic, 1992.
- [VEE 84] VEENDRICK, H. J. M. Short Circuit Dissipation of Static CMOS Circuitry and Its Impact on the Design of Buffer Circuits. **IEEE Journal Solid State Circuits**, New York, v. 19, n. 4, p. 468-473, Aug. 1984.
- [YEU 94] YEUNG, N. K. et al. The Design of a 55SPECint92 RISC Processor under 2W. In: IEEE INTERNATIONAL SOLID-STATE CIRCUITS CONFERENCE, 1994. **Proceedings ...** USA: [s.n.], 1994. p. 206-207.
- [ZAR 98] ZARKESH-HA, P. et al. On a Pin Versus *Gate* Relationship for Heterogeneous Systems: Heterogeneous Rent's Rule. In: IEEE CUSTOM INTEGRATED CIRCUIT CONFERENCE. **Proceedings...** USA: [s.n.], 1998. p. 98-96.